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

On the Size of Partial Derivatives and the Word Membership Problem

Stavros Konstantinidis · António Machiavelo · Nelma Moreira · Rogério Reis

Received: date / Accepted: date

Abstract Partial derivatives are widely used to convert regular expressions to nondeterministic automata. For the word membership problem, it is not strictly necessary to build an automaton. In this paper we study the size of partial derivatives on the average case. For expressions in strong star normal form, we show that on average and asymptotically the largest partial derivative is at most half the size of the expression. The results are obtained in the framework of analytic combinatorics considering generating functions of parametrised combinatorial classes defined implicitly by algebraic curves. Our average case estimates suggest that a detailed word membership algorithm based directly on partial derivatives should be analysed both theoretically and experimentally.

Keywords Regular expressions · Partial derivatives · Word membership · Average case · Analytic combinatorics

1 Introduction

Membership testing for a word of size n in the language represented by a regular expression of size m can be solved in $O(nm)$ time and $O(m)$ space by the

The authors António Machiavelo, Nelma Moreira and Rogério Reis were partially supported by CMUP, which is financed by national funds through FCT – Fundação para a Ciência e a Tecnologia, I.P., under the project with reference UIDB/00144/2020. Stavros Konstantinidis was partially supported by NSERC, Canada.

Stavros Konstantinidis
Saint Mary's University, Halifax, Nova Scotia, Canada,
E-mail: s.konstantinidis@smu.ca

António Machiavelo · Nelma Moreira · Rogério Reis
CMUP & DM-DCC, Faculdade de Ciências da Universidade do Porto, Rua do Campo Alegre, 4169-007 Porto, Portugal,
E-mail: {antonio.machiavelo,nelma.moreira,rogerio.reis}@fc.up.pt

construction of the Thompson nondeterministic finite automaton (NFA) [27]. Several improvements on the running time by polylogarithmic factors are known [24,4], but recently it was shown by a conditional lower bound [3,5] that $\Omega(nm^{1-\epsilon})$ time, for every $\epsilon > 0$, cannot be achieved assuming the strong exponential time hypothesis (SETH)[1].

Among other conversions from regular expressions to NFAs, the ones based on partial derivatives have been widely studied (see [6]). The partial derivative automaton, introduced by Mirkin and Antimirov [23,2,15], can be computed in $O(m^2)$ time and space [13,20], but its number of states is smaller than the number of states of the Thompson NFA (and of other known automaton constructions [14,6]).

The average-case complexity of several conversions from regular expressions to automata has also been studied using the framework of analytic combinatorics. Let the size of an NFA be the sum of the number of states plus the number of transitions. In particular, asymptotically and as the alphabet size grows, the average size of the partial derivative automaton is $\frac{3m}{4}$, where m is the size of the regular expression [7,8]. In comparison, the asymptotic average size of the Thompson automaton is $\frac{13m}{4}$ [9,11]. This motivates the study of the word membership problem using partial derivatives in spite of its quadratic worst-case complexity. In particular, it is well known that a word belongs to the language of a regular expression if and only if the empty word belongs to the language of one of its partial derivatives by that word. In this context, one can decide directly word membership without the need of constructing an automaton. Derivatives (and partial derivatives) have also been used directly for context-free parsing [1]. Initially, the proposed algorithms were thought to have an exponential running time, but indeed their time complexity can be $O(Gn^3)$, where G is the size of the context-free grammar and n the size of the word. This performance matches the best known one for parsing with combinatorial algorithms and is also an upper bound for parsing regular languages with derivatives.

In this paper, we estimate upper bounds for the average size of the largest partial derivative, as well as for the number of new operators that can be created when partial derivatives with respect to a symbol are computed. In addition to ordinary regular expressions, we will consider regular expressions in strong star normal form (ssnf) which are normalised expressions for which efficient algorithms are known and transforming a regular expression into this normal form can be achieved in linear time [12,14,18]. Using the framework of analytic combinatorics, the average-case complexity of several measures and conversions from ssnf expressions to other models was studied by Broda et al. [10]. In that study it was not feasible to have explicit formulae for the families of generating functions indexed by the alphabet size that one need to deal with. Thus, one need to use generating functions implicitly defined by algebraic curves, and a new method had to be developed to extract the required

[1] This hypothesis states that for every positive $\delta < 1$, SAT cannot be solved in time $O^*(2^{\delta n})$—see [22].

information for the asymptotic estimates. That was achieved by combining the use of the existence of Puiseux expansions at singularities with Newton polygon technique. This method allows to find, for the combinatorial classes considered, the behaviour of the generating function without knowing beforehand the explicit value of its singularity. In this paper we apply the same method, but a new technique had to be used in order to obtain the adequate polynomials of which the generating functions are roots, i.e., algebraic curves of which the generating functions are branches.

It is interesting to note that, although expressions in ssnf are more compact than standard ones, the asymptotic average estimates obtained by Broda et al. [10] coincide with the ones for standard regular expressions. In particular, the transformation of a standard regular expression into ssnf gives, for large alphabetic sizes, an expression of essentially the same size. This is not the case for the problem addressed in this paper. The asymptotic estimates for ssnf expressions are significantly smaller than the ones for standard regular expressions.

This paper is organised as follows. In the next section we review some basics on regular expressions and partial derivatives. Then we argue that an algorithm for the word membership problem based directly on partial derivatives is feasible on the average case using the results presented in this paper. Section 4 gives a brief overview of how we use the framework of analytic combinatorics for obtaining asymptotic estimates. In Section 5 we study the average size of partial derivatives for standard regular expressions. Section 6 has the main contributions of this work. For expressions in strong star normal form, we obtain several average estimates. Section 7 presents some experimental results that corroborate some of the estimates presented in the previous section. In Section 8 we consider a compact representation for the set of partial derivatives for expressions in strong star normal form, and study the average number of new concatenations when computing partial derivatives. Conclusions are drawn in Section 9.

2 Preliminaries

Given an alphabet $\Sigma = \{\sigma_1, \sigma_2, \ldots, \sigma_k\}$ of size $k \geq 1$, the set $\mathcal{R}_k$ of (standard) *regular expressions* α over Σ consists of $\emptyset$ and the expressions defined by the following context-free grammar:

$$\alpha := \varepsilon \mid \sigma_1 \mid \cdots \mid \sigma_k \mid (\alpha + \alpha) \mid (\alpha \cdot \alpha) \mid (\alpha)^\star, \tag{1}$$

where the $\cdot$ is often omitted. The *language* associated with α is denoted by $\mathcal{L}(\alpha)$ and is defined as usual. We say that α is *nullable* if $\varepsilon \in \mathcal{L}(\alpha)$. For the *size* of a regular expression α, denoted by $\|\alpha\|$, we will consider Polish notation length, i.e., the number of symbols in α, not counting parentheses. The *alphabetic size* of α, denoted by $|\alpha|_\Sigma$, is the number of letters in α. For a regular expression $\alpha \in \mathcal{R}_k$ and a symbol $\sigma \in \Sigma$, the set of *partial derivatives*

of α w.r.t. σ is defined inductively as follows:

$$\begin{aligned} &\partial_\sigma(\emptyset) = \partial_\sigma(\varepsilon) = \emptyset, && \partial_\sigma(\alpha+\alpha') = \partial_\sigma(\alpha) \cup \partial_\sigma(\alpha'), \\ &\partial_\sigma(\sigma') = \begin{cases} \{\varepsilon\} & \text{if } \sigma' = \sigma, \\ \emptyset & \text{otherwise,} \end{cases} && \partial_\sigma(\alpha\alpha') = \begin{cases} \partial_\sigma(\alpha)\alpha' \cup \partial_\sigma(\alpha), & \text{if } \varepsilon \in \mathcal{L}(\alpha) \\ \partial_\sigma(\alpha)\alpha', & \text{otherwise,} \end{cases} \\ &\partial_\sigma(\alpha^\star) = \partial_\sigma(\alpha)\alpha^\star, \end{aligned} \tag{2}$$

where, for any $S \subseteq \mathcal{R}_k \setminus \{\emptyset\}$, we define $S\emptyset = \emptyset S = \emptyset$, $S\varepsilon = \{\varepsilon\}S = S$, and,

$$S\alpha' = \{\, \alpha\alpha' \mid \alpha \in S \wedge \alpha \neq \varepsilon \,\} \cup \{\, \alpha' \mid \exists \varepsilon \in S \,\}$$

if $\alpha' \neq \emptyset, \varepsilon$. The definition of partial derivatives can be extended in a natural way to sets of regular expressions, words, and languages. We have that $\mathcal{L}(\partial_w(\alpha)) = \{\, w' \mid ww' \in \mathcal{L}(\alpha) \,\}$, for $w \in \Sigma^*$. The set of all partial derivatives of α w.r.t. non-null words is denoted by $\partial^+(\alpha)$ and satisfies the following.

Proposition 1 ([23])

$$\begin{aligned} &\partial^+(\emptyset) = \emptyset, && \partial^+(\alpha+\beta) = \partial^+(\alpha) \cup \partial^+(\beta), \\ &\partial^+(\varepsilon) = \emptyset, && \partial^+(\alpha\beta) = \partial^+(\alpha)\beta \cup \partial^+(\beta), \\ &\partial^+(\sigma) = \{\varepsilon\}, && \partial^+(\alpha^\star) = \partial^+(\alpha)\alpha^\star. \end{aligned} \tag{3}$$

Proposition 2 ([2], Th. 3.4) *For any regular expression α, the following inequality holds: $|\partial^+(\alpha)| \leq |\alpha|_\Sigma$.*

Proposition 3 ([2], Th. 3.8) *Given a regular expression α, a partial derivative of α is either ε or a concatenation $\alpha_1 \cdot \alpha_2 \cdot \cdots \cdot \alpha_n$ such that α_i is a subexpression of α and n is no greater than the number of occurrences of concatenations and stars in α.*

Corollary 1 *For $\beta \in \partial^+(\alpha)$, the size $\|\beta\|$ is $O(\|\alpha\|^2)$.*

We are interested in the maximal size of the partial derivatives of a regular expression. Given a set of regular expressions S, let

$$m(S) = \max\{\, \|\alpha\| \mid \alpha \in S \,\}.$$

Thus, we are interested in $m(\partial^+(\alpha))$, for $\alpha \in \mathcal{R}_k$.

Lemma 1 *For all alphabetic sizes greater than 1 and all expressions α of size $n > 1$, the maximum value of $m(\partial^+(\alpha))$ is $\binom{n+1}{2} + n - 3 = \frac{n(n+3)}{2} - 3$.*

Proof The maximal size is obtained for the family of expressions s_n, with $s_2 = a^\star$, $s_3 = a^{\star^\star}$, $s_4 = a^{\star^{\star^\star}}$, ..., with $a \in \Sigma$. We have $\|s_n\| = n$ and $\partial^+(s_n) = \{s_2 \cdots s_n\}$, for $n > 1$. This is maximal because only the operator $\star$ leads to strictly large expressions (which can be proved by induction). Then,

$$m(\partial^+(s_n)) = \|s_2 \cdots s_n\| = \binom{n+1}{2} + n - 3 = \frac{n(n+3)}{2} - 3.$$

For instance, $\partial^+(s_4) = \{a^\star \cdot a^{\star^\star} \cdot a^{\star^{\star^\star}}\}$ and $m(\partial^+(s_4)) = 11$. [2] □

Concerning average-case estimates we recall the following results.

Proposition 4 ([25, 7]) *Asymptotically, and as the alphabet size grows, the average size of* $|\alpha|_\Sigma$ *is* $\frac{\|\alpha\|}{2}$.

Proposition 5 ([7]) *Asymptotically, and as the alphabet size grows, the average size of* $\partial^+(\alpha)$ *is* $\frac{\|\alpha\|}{4}$.

Proposition 6 ([11]) *For expressions* α *of size* n *and alphabet size* k*, asymptotically the average number of concatenations is* $\frac{n}{4}$ *and the number of stars is* $\frac{4}{-1+\sqrt{8+8k}}n$*, which tends to* 0 *as* k *grows.*

3 Motivation

The motivation for estimating sizes of partial derivatives (PDs), in this work, as well as the number of PDs, in the literature, is that these parameters affect the performance of algorithms for the *word membership problem:* given regular expression α and word w, decide whether $w \in \mathcal{L}(\alpha)$. We are particularly interested in such algorithms working directly on the PDs of the given α (without building the PD automaton). The general algorithmic method (shown below) iterates through each symbol σ of the input word and computes the next set of derivatives w.r.t. σ from the current set whose initial value is $\{\alpha\}$. The decision is YES iff the final set contains a partial derivative γ with $\varepsilon \in \mathcal{L}(\gamma)$ (i.e. if γ is *nullable*).

WordMembership algorithm: given $\alpha \in \mathcal{R}_k$ and word $w \in \Sigma^\star$

```
Curr := {α}
for each symbol σ of w
    Next := ∅
    for each γ ∈ Curr
        Next := Next ∪ ∂_σ(γ)
    Curr := Next
for each γ ∈ Curr              # Curr = ∂_w(α)
    if ε ∈ L(γ) return YES     # w ∈ L(α) iff ε ∈ ⋃_{γ∈∂_w(α)} L(γ)
return NO
```

The details and exact complexity of the above algorithm depend on the choice of data structures and require further analysis. In any case, we have that

(i) in each iteration of the inner loop, at most $|\alpha|_\Sigma$ PDs are computed—this follows from Proposition 2, and

[2] Note that $m(\partial^+(s_n))$ is sequence $A034856$ minus 2 in OEIS (`https://oeis.org/A034856`).

(ii) the algorithmic size[3] $s(\gamma)$ of each partial derivative γ affects the complexity of the algorithm.

Corollary 1 gives the upper bound $O(\|\alpha\|^2)$ for $s(\gamma)$. Instead of the worst case estimate for the largest size of a PD, we can consider average case estimates. The main results of this work imply that the average value of $s(\gamma)$ is $O(\|\alpha\|^{3/2})$—see Theorem 3—and if α is in strong star normal form then the average value of $s(\gamma)$ is only $O(\|\alpha\|)$—see Theorem 4. Moreover using (Section 8) the compact tree-like representation of the set of PDs we have that, asymptotically and on average, a constant number of new nodes, depending only on $|\Sigma|$, are added when computing the set $\partial_\sigma(\gamma)$ of a previously computed PD γ—see Theorem 5. For deciding whether γ is nullable, one can mark each node (= a subexpression) of the tree-like structure as nullable or not, while computing the PDs—the nullability of each node can be computed as a function of its subexpressions. This can be done in linear time.

4 Asymptotic Coefficients of Generating Functions Given by Algebraic Curves

Given some measure of the objects of a combinatorial class, $\mathcal{A}$, for each $n \in \mathbb{N}_0$, let a_n be the sum of the values of this measure for all objects of size n. Now, let $A(z) = \sum_n a_n z^n$ be the corresponding generating function (*cf.* [17]). The generating function $A(z)$ can be seen as a complex analytic function. When this function has a unique dominant singularity ρ, the study of the behaviour of $A(z)$ around it gives us access to the asymptotic form of its coefficients. In particular, if $A(z)$ is analytic in some indented disc neighbourhood of ρ, then one has the following [17, Corol. VI.1, p. 392]:

Theorem 1 *The coefficients of the series expansion of the complex function*

$$f(z) \underset{z \to \rho}{\sim} \lambda \left(1 - \frac{z}{\rho}\right)^{\nu},$$

where $\nu \in \mathbb{C} \setminus \mathbb{N}_0$, $\lambda \in \mathbb{C}$, have the following asymptotic approximation:

$$[z^n]f(z) = \frac{\lambda\, n^{-\nu-1}\rho^{-n}}{\Gamma(-\nu)} + o\left(n^{-\nu-1}\rho^{-n}\right).$$

Here Γ is, as usual, the Euler's gamma function.

Thus, to use this, one needs to have a way to obtain the singularity, ρ, as well as the constants ν and λ.

The combinatorial classes that we deal with give rise to generating functions implicitly defined by algebraic curves. Moreover, as we are interested in understanding how the average complexity of those classes also varies with

[3] This could be $\|\gamma\|$. We note, however, that representing a *set of PDs* with the compact method of Section 8 results into subtree sharing, so the total size of the set is less than the sum of the sizes of its elements.

the size of the alphabet, we were forced to understand how the singularities depend on that size, which precludes any sort of numerical approach.

In previous works [10,21,11], a method was developed to extract the required information for the asymptotic estimates. That was achieved by combining the use of the existence of Puiseux expansions at singularities, with the Newton polygon technique. This method allows to find, for the combinatorial classes considered, the behaviour of the generating function without knowing beforehand the explicit value of its singularity. This provides a very useful technique that circumvents some of the more cumbersome steps of the *Algebraic Coefficient Asymptotics* algorithm presented by Flajolet and Sedgewick [17, pages 504 − 505], and reduces to a minimum the use of inexact numerical methods.

Generically, from an unambiguous generating grammar, one obtains a set of polynomial equations involving the generating functions for the objects corresponding to the variables of the grammar, in particular the one whose coefficients we want to asymptotically estimate. Computing a Gröbner basis for the ideal generated by those polynomials, one gets an algebraic equation for that generating function $w = w(z)$, i.e., an equation of the form

$$G(z, w) = 0,$$

where $G(z, w)$ is a polynomial in $\mathbb{Z}[z][w]$ of which $w(z)$ is a root.

Since $w(z)$ is the generating function of a combinatorial class, thus a series with non-negative integer coefficients which is not a polynomial, it must have, by Pringsheim's Theorem (*cf.* [17], Thm IV.6), a real positive singularity, ρ, smaller than or equal to 1. In the case of the generating functions dealt with in this paper, it turns out that there is no other singularity with that norm. At this singularity, ρ, two cases may occur:

Case I: $\lim_{z\to\rho} w(z) = a$, where a is a positive real number.
Case II: $\lim_{z\to\rho} w(z) = +\infty$.

Analysing the form of the curve G, and using its partial derivatives, one can find an irreducible polynomial for the singularity ρ_k, and, in Case I, an irreducible polynomial for a. In Case II, the irreducible polynomial for ρ is a factor of the leading coefficient of $G(z, w)$ when seen as a polynomial in w (*cf.* [19], Th. 12.2.1). After making the change of variable $s = 1 - z/\rho$, one knows that $w = w(s)$ has a Puiseux series expansion at the singularity $s = 0$, i.e., there exists a slit neighbourhood of that point in which $w(s)$ has a representation as a power series with fractional powers (*cf.* [19], Chap. 12). Using the irreducible polynomial for ρ, and the one for a in Case I, while in Case II one changes variables in order to replace $+\infty$ with 0, one decides which partial derivatives of G are non-zero, and uses that information to draw a Newton polygon that yields the value of ν needed to apply Theorem 1. To obtain the asymptotic approximation for the coefficients of the original generating function one uses the following result from [11].

Theorem 2 *With the notations and in the conditions above described, one has*

$$[z^n]w(z) \underset{n\to\infty}{\sim} \begin{cases} \dfrac{-b_G}{\Gamma(-\nu)}\rho^{-n}n^{-\nu-1}, & \text{if } \lim_{z\to\rho} w(z) \in \mathbb{R}, \quad (4) \\ \dfrac{1}{c_G\,\Gamma(\nu)}\rho^{-n}n^{\nu-1}, & \text{if } \lim_{z\to\rho} w(z) = +\infty, \quad (5) \end{cases}$$

where ρ and ν are as above, and b_G, c_G can be computed using the Newton polygon technique, using the partial derivatives of G (see [11] for more details).

5 Average Maximal Size of Partial Derivatives for Standard Regular Expressions

We start by considering the problem of estimating the average size of the partial derivatives for standard regular expressions. Considering the grammar (1), the generating function for $\mathcal{R}_k$, $R_k = R_k(z)$, satisfies the equation

$$R_k = (k+1)z + 2zR_k^2 + zR_k. \quad (6)$$

To estimate the average value $m(\partial^+(\alpha))$ for a regular expression α of size n, we consider a cost function defined by induction in the structure of α. Whenever the expression α is of the form $\alpha' \circ \alpha''$ with $\circ \in \{+, \cdot\}$, the cost of α should be the maximum of the values of the operands. However, the standard symbolic method [17] cannot be applied with the maximum function. Therefore, as an upper bound we consider the sum of the values of the operands, as we know that for any sets of expressions S_1 and S_2, $m(S_1 \cup S_2) \leq m(S_1) + m(S_2)$. For example, for $\partial^+(\alpha + \alpha') = \partial^+(\alpha) \cup \partial^+(\alpha')$, we have $m(\partial^+(\alpha + \alpha')) \leq m(\partial^+(\alpha)) + m(\partial^+(\alpha'))$. With this in mind and using (3), we define the cost function c, for the maximal size of partial derivatives, satisfying the following

$$\begin{aligned} c(\varepsilon) = 0,\ c(\sigma) &= 1, \\ c(\alpha+\beta) &= c(\alpha) + c(\beta), \\ c(\alpha\cdot\beta) &= c(\alpha) + \|\beta\| + 1 + c(\beta), \\ c(\alpha^\star) &= c(\alpha) + \|\alpha\| + 2. \end{aligned}$$

The generating function $C_k = C_k(z) = \sum_\alpha c(\alpha) z^{\|\alpha\|}$ for the cost c satisfies the equation

$$\begin{aligned} C_k &= \sum_\sigma z^{\|\sigma\|} + \sum_{\alpha,\beta} c(\alpha+\beta) z^{\|\alpha+\beta\|} + \sum_{\alpha,\beta} c(\alpha\cdot\beta) z^{\|\alpha\cdot\beta\|} + \sum_\alpha c(\alpha^\star) z^{\|\alpha^\star\|}, \\ &= kz + z\sum_\alpha c(\alpha) z^{\|\alpha\|} \sum_\beta z^{\|\beta\|} + \sum_\beta c(\beta) z^{\|\beta\|} \sum_\alpha z^{\|\alpha\|} + \\ &\quad z\sum_{\alpha,\beta} (c(\beta) + c(\alpha) + \|\beta\| + 1) z^{\|\alpha\|} z^{\|\beta\|} + z\sum_\alpha (c(\alpha) + \|\alpha\| + 2) z^{\|\alpha\|}, \\ &= kz + 4zC_kR_k + zR_kS_k + zR_k^2 + zC_k + zS_k + 2zR_k, \\ &= z(k + 2R_k + S_k + C_k + 4C_kR_k + R_kS_k + R_k^2), \end{aligned}$$

where

$$S_k = \sum_{\alpha} \|\alpha\| z^{\|\alpha\|} = z\partial_z R_k = (k+1)z + 2zR_k^2 + 4zR_kS_k + zR_k + zS_k.$$

Using Gröbner basis on the polynomial equations for C_k, S_k, and R_k, one gets that $w = C_k(z)$ is root of

$$4m_k(z)^2\, w^2 + m_k(z)p_2(z)\, w - zp_3(z),$$

where

$$\begin{aligned} m_k(z) &= 1 - 2z - (7+8k)z^2, \\ p_2(z) &= 1 - 4(2+k)z - (21+24k)z^2, \\ p_3(z) &= k + 3(1-2k)z - (21+24k+18k^2)z^2 - (49+63k+15k^2+8k^3)z^3. \end{aligned}$$

Proceeding as explained in the previous section, we see that one is here dealing with case (5) of Theorem 2, and computing the respective constants, one gets: $\rho_k = \frac{-1+\sqrt{8+8k}}{7+8k}$, the positive root of $m_k(z)$; $\nu = 1$; and $c_{C_k} = \frac{16}{2+\sqrt{2+2k}}$. The value for the asymptotic behaviour of the coefficients of $C_k(z)$ is then:

$$[z^n]C_k(z) \underset{n\to\infty}{\sim} \frac{1}{c_{C_k}}\rho_k^{-n} = \frac{2+\sqrt{2+2k}}{16}\rho_k^{-n}, \tag{7}$$

which gives the total accumulative size of all upper bounds for the largest partial derivative for each of the regular expressions of size n.

Recalling that [8,11]

$$[z^n]R_k(z) \underset{n\to\infty}{\sim} \frac{\sqrt{2-2\rho_k}}{8\rho_k\sqrt{\pi}}\, n^{-\frac{3}{2}}\, \rho_k^{-n}, \tag{8}$$

one now gets

Theorem 3 *The average ratio of an upper bound for the maximum size of partial derivatives of an expression of size n over its original size is given by*

$$\frac{[z^n]C_k(z)}{n[z^n]R_k(z)} \underset{n\to\infty}{\sim} \frac{\sqrt{1+k}\,\sqrt{\pi}}{\sqrt{2}+4\sqrt{1+k}}\, n^{\frac{1}{2}} \underset{k\to\infty}{\sim} \frac{\sqrt{\pi}}{4}\, n^{\frac{1}{2}}. \tag{9}$$

This means that the largest partial derivative of a regular expression is, on average and asymptotically, a constant times the square root of the size of the expression. Although we can conclude that asymptotically on average $m(\partial^+(\alpha))$ is $O(\|\alpha\|^{\frac{3}{2}})$, instead of the worst-case $O(\|\alpha\|^2)$, it is not linear as one may except. Note that this can be due to our cautious upper bound. In fact if, whenever we have a union in the right-hand size of (3) we consider only one of the terms when computing c above then a linear estimate is obtained. However, these computations do not ensure that we obtain an estimate for the maximal value.

6 Strong Star Normal Form and Partial Derivatives

A regular expression α is in strong star normal form (ssnf) if for any subexpression of the form $\beta^\star$ or $\beta+\varepsilon$, β is not nullable. Introducing the operator *option* $^?$ with $\mathcal{L}(\beta^?) = \mathcal{L}(\beta)\cup\{\varepsilon\}$, one can define the set $\mathcal{S}_k$ of *regular expressions in* ssnf over some alphabet $\Sigma = \{\sigma_1, \ldots, \sigma_k\}$ by the following context-free grammar:

$$\begin{aligned} \alpha &:= \varepsilon \mid \emptyset \mid \beta_\varepsilon \mid \beta_{\overline{\varepsilon}}, \\ \beta_\varepsilon &:= \beta_\varepsilon \cdot \beta_\varepsilon \mid \beta_\varepsilon + \beta_{\overline{\varepsilon}} \mid \beta_{\overline{\varepsilon}} + \beta_\varepsilon \mid \beta_\varepsilon + \beta_\varepsilon \mid \beta_{\overline{\varepsilon}}^\star \mid \beta_{\overline{\varepsilon}}^?, \\ \beta_{\overline{\varepsilon}} &:= \sigma_1 \mid \cdots \mid \sigma_k \mid \beta_{\overline{\varepsilon}}\beta_{\overline{\varepsilon}} \mid \beta_{\overline{\varepsilon}}\beta_\varepsilon \mid \beta_\varepsilon\beta_{\overline{\varepsilon}} \mid \beta_{\overline{\varepsilon}} + \beta_{\overline{\varepsilon}}, \end{aligned} \tag{10}$$

where β_ε are regular expressions nullable, while for $\beta_{\overline{\varepsilon}}$, $\varepsilon \notin \mathcal{L}(\beta_{\overline{\varepsilon}})$. In the remaining of the paper we will use β to denote either of these expressions.

For a regular expression $\beta \in \mathcal{S}_k$ and a symbol $\sigma \in \Sigma$, the set of *partial derivatives* of β w.r.t. σ is defined inductively as in (2), except for the following cases:

$$\begin{aligned} \partial_\sigma(\beta_\varepsilon\beta) &= \partial_\sigma(\beta_\varepsilon)\beta \cup \partial_\sigma(\beta), & \partial_\sigma(\beta_{\overline{\varepsilon}}^\star) &= \partial_\sigma(\beta_{\overline{\varepsilon}})\beta_{\overline{\varepsilon}}^\star, \\ \partial_\sigma(\beta_{\overline{\varepsilon}}\beta) &= \partial_\sigma(\beta_{\overline{\varepsilon}})\beta, & \partial_\sigma(\beta_{\overline{\varepsilon}}^?) &= \partial_\sigma(\beta_{\overline{\varepsilon}}). \end{aligned} \tag{11}$$

And, the set of all partial derivatives of $\beta \in \mathcal{S}_k$ w.r.t. non-null words $\partial^+(\beta)$ satisfies the following.

Proposition 7

$$\begin{aligned} \partial^+(\emptyset) = \partial^+(\varepsilon) &= \emptyset, & \partial^+(\beta+\beta') &= \partial^+(\beta)\cup\partial^+(\beta'), \\ \partial^+(\sigma) &= \{\varepsilon\}, & \partial^+(\beta\beta') &= \partial^+(\beta)\beta' \cup \partial^+(\beta'), \\ \partial^+(\beta_{\overline{\varepsilon}}^\star) &= \partial^+(\beta_{\overline{\varepsilon}})\beta_{\overline{\varepsilon}}^\star, & \partial^+(\beta_{\overline{\varepsilon}}^?) &= \partial^+(\beta_{\overline{\varepsilon}}). \end{aligned} \tag{12}$$

Proof The proof follows from the one of Proposition 3. □

Proposition 8 *If $\beta \in \mathcal{S}_k$ then for all $\sigma \in \Sigma$, $\partial_\sigma(\beta) \subseteq \mathcal{S}_k$. Moreover, we have $\partial^+(\beta) \subseteq \mathcal{S}_k$.*

Proof The proof follows by induction on the structure of the expressions and on the size of the words. □

The following example shows that for ssnf expressions β the (maximal) size of partial derivatives can also be $\Theta(\|\beta\|^2)$.

Example 1 Consider $r_0 = a$ and $r_n = (r_{n-1}^\star \cdot a)$, for $n \geq 1$ over the unary alphabet $\{a\}$. For instance, we have $r_3 = ((a^\star a)^\star a)^\star a$. These expressions belong to $\mathcal{S}_1$ and the size of r_n is $3n+1$, for $n \geq 0$. For $n \geq 1$, the largest partial derivative of $\partial_a(r_n) = \{r_i \cdots r_n \mid i \in [1,n]\} \cup \{\varepsilon\}$ is $r_1 r_2 \cdots r_n$ whose size is

$$n - 1 + \sum_{i=1}^{n}(3i+1) = \frac{3n^2+7n-2}{2} = \Theta(n^2).$$

For instance, $\partial_a(r_3) = \{r_1r_2r_3,\, r_2r_3,\, r_3,\, \varepsilon\}$. Because $\partial^+(r_n) = \partial_a(r_n)$ the largest partial derivative has also size $\Theta(n^2)$. □

6.1 Average Ratio of β_ε and $\beta_{\overline{\varepsilon}}$

The generating functions for β_ε and $\beta_{\overline{\varepsilon}}$ regular expressions, $B_k = B_k(z)$ and $\overline{B}_k = \overline{B}_k(z)$, satisfy respectively,

$$B_k = 2zB_k^2 + 2zB_k\overline{B}_k + 2z\overline{B}_k, \tag{13}$$
$$\overline{B}_k = kz + 2zB_k\overline{B}_k + 2z\overline{B}_k^2. \tag{14}$$

We start by estimating the ratio of expressions β_ε over all expressions in $\mathcal{S}_k$, for large k. As done in [10], one sees that B_k and $\overline{B}_k$ have the same singularity, namely the only root, η_k, of the polynomial

$$\ell_k(z) = z^3 + \frac{9}{2k+27}\,z^2 - \frac{1}{4(2k+27)}\,z - \frac{1}{k(2k+27)}, \tag{15}$$

in the interval $]0,1[$ and one then gets that

$$[z^n]B_k(z) \underset{n\to\infty}{\sim} \frac{b_{B_k}}{2\sqrt{\pi}}\eta_k^{-n}n^{-\frac{3}{2}},$$
$$[z^n]\overline{B}_k(z) \underset{n\to\infty}{\sim} \frac{b_{\overline{B}_k}}{2\sqrt{\pi}}\eta_k^{-n}n^{-\frac{3}{2}},$$

where b_{B_k} and $b_{\overline{B}_k}$ satisfy

$$b_{B_k} \underset{k\to\infty}{\sim} \sqrt{8} \quad \text{and} \quad b_{\overline{B}_k} \underset{k\to\infty}{\sim} \sqrt{k}. \tag{16}$$

We note that it follows from this that

Proposition 9 *The ratio of the total number of expressions β_ε of size n to the total number of expressions in $\mathcal{S}_k$ of the same size is given by*

$$\frac{[z^n]B_k}{[z^n]B_k + [z^n]\overline{B}_k} \underset{k\to\infty}{\sim} \sqrt{\frac{8}{k}}.$$

We can conclude that as the alphabet size grows, the occurrence of expressions β_ε tends to zero. This result can suggest that for large alphabets one can study the average complexity considering only expressions $\beta_{\overline{\varepsilon}}$.

6.2 Average Maximal Size of Partial Derivatives for $\mathcal{S}_k$

From equations (2) and (11), the size of the largest partial derivative is bounded by the function s satisfying:

$$\begin{aligned}
s(\sigma) &= 1,\\
s(\beta^{\star}_{\overline{\varepsilon}}) &= s(\beta_{\overline{\varepsilon}}) + \|\beta_{\overline{\varepsilon}}\| + 2,\\
s(\beta^{?}_{\overline{\varepsilon}}) &= s(\beta_{\overline{\varepsilon}}),\\
s(\beta_{\varepsilon} + \beta_{\overline{\varepsilon}}) &= s(\beta_{\overline{\varepsilon}} + \beta_{\varepsilon}) = s(\beta_{\varepsilon}) + s(\beta_{\overline{\varepsilon}}),\\
s(\beta_{\overline{\varepsilon}} + \beta_{\overline{\varepsilon}}) &= 2s(\beta_{\overline{\varepsilon}}),\\
s(\beta_{\varepsilon} + \beta_{\varepsilon}) &= 2s(\beta_{\varepsilon}),\\
s(\beta_{\varepsilon}\beta_{\varepsilon}) &= 2s(\beta_{\varepsilon}) + \|\beta_{\varepsilon}\| + 1,\\
s(\beta_{\overline{\varepsilon}}\beta_{\overline{\varepsilon}}) &= s(\beta_{\overline{\varepsilon}}) + \|\beta_{\overline{\varepsilon}}\| + 1,\\
s(\beta_{\overline{\varepsilon}}\beta_{\varepsilon}) &= s(\beta_{\overline{\varepsilon}}) + \|\beta_{\varepsilon}\| + 1,\\
s(\beta_{\varepsilon}\beta_{\overline{\varepsilon}}) &= s(\beta_{\varepsilon}) + \|\beta_{\overline{\varepsilon}}\| + 1 + s(\beta_{\overline{\varepsilon}}).
\end{aligned} \tag{17}$$

Note again that this is a cautious upper bound, since we use as an upper bound for the maximum size of elements of the union of two sets, the sum of the maximum size of each set. Let $E_k = E_k(z)$ and $N_k = N_k(z)$ be the cost generating functions, for the measure s, associated with the expressions β_ε and $\beta_{\overline{\varepsilon}}$, respectively, and

$$F_k = E_k + N_k,$$

be the generating function for our upper bound for the sum of the sizes of the largest partial derivative of each regular expression in ssnf of a given size. To show how to obtain equations characterising these functions, one proceeds as follows. Let

$$E_k = \sum_{\beta_\varepsilon} s(\beta_\varepsilon) z^{\|\beta_\varepsilon\|}.$$

Using the grammar (10) and equations (17), one obtains

$$\begin{aligned}
E_k &= \sum_{\beta,\beta' \in \beta_\varepsilon} s(\beta\cdot\beta') z^{\|\beta\cdot\beta'\|} + \cdots\\
&= z \sum_{\beta,\beta' \in \beta_\varepsilon} (s(\beta) + s(\beta')) z^{\|\beta\|+\|\beta'\|} + z \sum_{\beta,\beta' \in \beta_\varepsilon} \|\beta'\| z^{\|\beta\|+\|\beta'\|}\\
&\qquad + z \sum_{\beta,\beta' \in \beta_\varepsilon} z^{\|\beta\|+\|\beta'\|} + \cdots\\
&= z \sum_{\beta \in \beta_\varepsilon} s(\beta) z^{\|\beta\|} \sum_{\beta' \in \beta_\varepsilon} z^{\|\beta'\|} + z \sum_{\beta \in \beta_\varepsilon} z^{\|\beta\|} \sum_{\beta' \in \beta_\varepsilon} s(\beta') z^{\|\beta'\|} +\\
&\qquad z \sum_{\beta \in \beta_\varepsilon} z^{\|\beta\|} \sum_{\beta' \in \beta_\varepsilon} \|\beta'\| z^{\|\beta'\|} + z \sum_{\beta \in \beta_\varepsilon} z^{\|\beta\|} \sum_{\beta' \in \beta_\varepsilon} z^{\|\beta'\|} + \cdots
\end{aligned}$$

Noting that $\sum_{\beta' \in \beta_\varepsilon} \|\beta'\| z^{\|\beta'\|} = z\partial_z(B_k)$, one obtains:

$$E_k = 2zE_kB_k + zB_k\, z\partial_z(B_k) + zB_k^2 + \cdots$$

Doing all the computations, one eventually gets:

$$E_k = 2z\left(E_k\overline{B}_k + 2E_kB_k + N_kB_k + N_k\right) + z\left(D_kB_k + B_k^2 + \overline{D}_k + 2\overline{B}_k\right), \tag{18}$$

$$N_k = z\left(k + 2N_k(B_k + \overline{B}_k) + D_k\overline{B}_k + \overline{D}_k(B_k + \overline{B}_k) + \overline{B}_k^2\right) + z\left(2B_k\overline{B}_k + E_k\overline{B}_k\right), \tag{19}$$

where

$$\begin{aligned} D_k &= z\partial_z(B_k) \\ &= 2z\left(B_k^2 + B_k\overline{B}_k + \overline{B}_k + 2B_kD_k + D_k\overline{B}_k + B_k\overline{D}_k + \overline{D}_k\right), \end{aligned} \tag{20}$$

$$\begin{aligned} \overline{D}_k &= z\partial_z(\overline{B}_k) \\ &= kz + 2z\left(B_k\overline{B}_k + \overline{B}_k^2 + D_k\overline{B}_k + B_k\overline{D}_k + 2\overline{B}_k\overline{D}_k\right). \end{aligned} \tag{21}$$

All the symbolic manipulators that we tried were unable to compute, from these equations, the algebraic curves containing E_k and N_k as branches that we need in order to apply the method described in [11]. We have, however, found a workaround to obtain those curves, which we now describe. Using Gröbner basis for (13) and (14), one gets

$$B_k = \frac{2}{k}\,\overline{B}_k^2, \tag{22}$$

$$0 = 2zB_k\overline{B}_k + kzB_k - \overline{B}_k + kz, \tag{23}$$

and from these, one sees that $\overline{B}_k$ is a root of the polynomial

$$f(X) = 4zX^3 + 2kzX^2 - kX + k^2z.$$

The equations for D_k, $\overline{D}_k$, E_k and N_k allow us to write these functions as rational functions of $\overline{B}_k$. To do that one can proceed as follows. Solve (18) for N_k, and do the same for (19). Equating these, one gets an expression that, when solved for E_k, yields E_k as a rational function in terms of $B_k, \overline{B}_k, D_k$, and $\overline{D}_k$. Using then (20) and (21), which when solved for D_k and $\overline{D}_k$, yields these two generating functions in terms of B_k and $\overline{B}_k$, one obtains E_k, then N_k, and hence F_k, as a rational function of B_k and $\overline{B}_k$, and finally (22) allow us to get F_k as a rational function in $\overline{B}_k$, $F_k = \frac{U_k}{V_k}$.

Now, one computes the inverse, $\bar{V}_k$, of V_k module the polynomial $f(X)$. Reducing $U_k\bar{V}_k$, again module $f(X)$, one obtains F_k as a polynomial in $\overline{B}_k$. Let $g(X)$ be that polynomial, i.e. $F_k(z) = g(\overline{B}_k(z))$. By reducing it modulo $f(X)$, one can get a polynomial $g(X)$ of degree at most 3. Finally, working on the extension $\mathbb{Q}(k,z)(\gamma)$, with $\gamma = \overline{B}_k$, which essentially means working modulo $f(X)$, and putting $\mu = g(\overline{B}_k)$, one obtains $a_{ij} \in \mathbb{Q}(k,z)$ such that

$$\begin{aligned} \mu &= a_{11} + a_{12}\gamma + a_{11}\gamma^2, \\ \mu\gamma &= a_{21} + a_{22}\gamma + a_{23}\gamma^2, \\ \mu\gamma^2 &= a_{31} + a_{32}\gamma + a_{33}\gamma^2. \end{aligned}$$

A polynomial equation of degree at most 3, with coefficients in $\mathbb{Q}(k,z)$, can now be obtained for $\overline{B}_k$ from the fact that the following determinant is null:

$$\begin{vmatrix} \mu - a_{11} & -a_{12} & -a_{13} \\ -a_{21} & \mu - a_{22} & -a_{23} \\ -a_{31} & -a_{32} & \mu - a_{33} \end{vmatrix} = 0.$$

Clearing denominators, this yields a polynomial in $\mathbb{Q}(k,z)[w]$ that has $w = F_k(z)$ as a root. This polynomial is too large to be explicitly given here, having the form

$$16z\,\ell_k(z)\,p_5(z)\,p_6(z)\,w^3 + 4z\,\ell_k(z)\,p_{12(z)}\,w^2 - p_{16}(z)\,w + kz\,p_{15}(z),$$

where $\ell_k(z)$ is the polynomial given in (15), and $p_i(z)$ is an irreducible polynomial in $\mathbb{Q}(k)[z]$ of degree i. The singularity of $F_k(z)$ is again η_k, the only positive root of the polynomial $\ell_k(z)$ in the interval $]0,1[$.

Proceeding as explained in [11], we see that one is here dealing with case (5) of Theorem 2, and computing the respective constants ρ, ν, and c, one gets the following value for the asymptotic behaviour of the coefficients of $F_k(z)$:

$$[z^n]F_k(z) \underset{n\,\to\,\infty}{\sim} \frac{1}{c_{F_k}\sqrt{\pi}}\eta_k^{-n}n^{-\frac{1}{2}}, \tag{24}$$

where c_{F_k} is a function of k with an expression too cumbersome to write here, but that satisfies

$$c_{F_k} \underset{k\,\to\,\infty}{\sim} \frac{4}{\sqrt{k}}. \tag{25}$$

From this one now gets

Theorem 4 *The average ratio of an upper bound for the maximum size of partial derivatives of an expression in $\mathcal{S}_k$ of size n over its original size is given by*

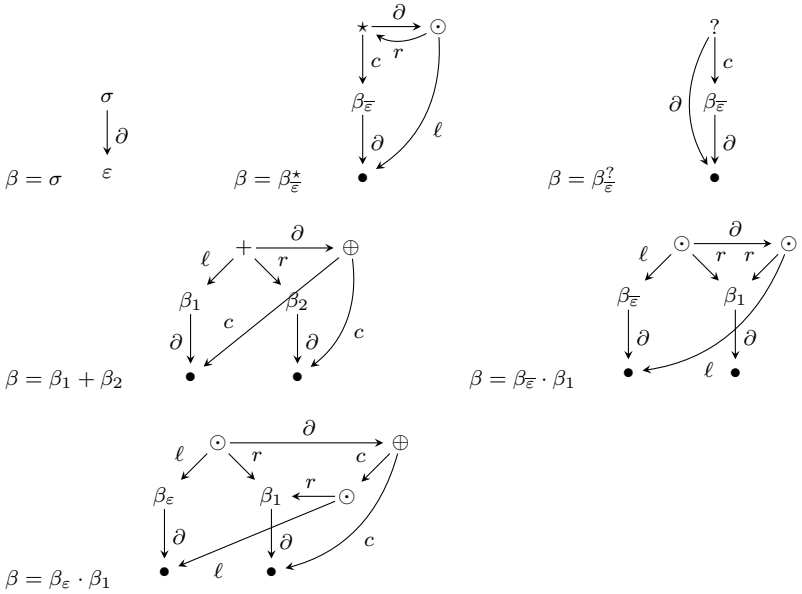
$$\frac{[z^n]F_k(z)}{n[z^n](B_k(z)+\overline{B}_k(z))} \underset{n\,\to\,\infty}{\sim} \frac{2}{c_{F_k}(b_{B_k}+b_{\overline{B}_k})} \underset{k\,\to\,\infty}{\sim} \frac{1}{2}. \tag{26}$$

This means that the largest partial derivative of an ssnf regular expression is, on average and asymptotically as the alphabet grows, half the size of the expression.

7 Experimental Results

We ran some experiments, using the FAdo package [26], to obtain average maximal sizes of partial derivatives. We consider both standard expressions (α) and strong star normal form expressions (ssnf). Table 1 summarises some of the results. For the results to be statistically significant, regular expressions were uniformly random generated using a version of the grammars for $\mathcal{R}_k$ and $\mathcal{S}_k$ in reverse polish notation. For each expression size $n \in \{100, 200, 300, 500, 1000, 2000\}$,

Fig. 1 Compact tree-like representation of the set of partial derivatives of β w.r.t. one symbol, σ.

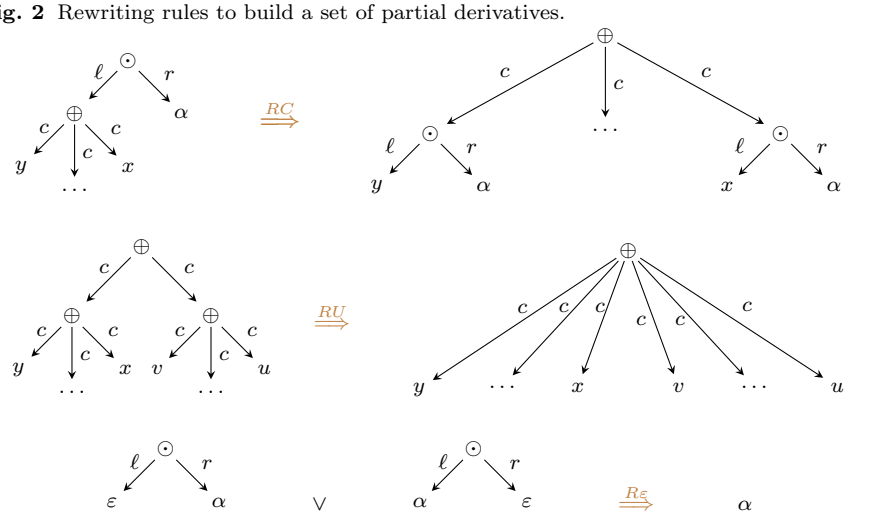


and alphabet size $k \in \{1, 2, 10, 50\}$, samples of 10000 expressions were generated (of each kind). This is sufficient to ensure a 95% confidence level within a 1% error margin [16, p. 75]. For each expression, the maximal size of its partial derivatives was computed. For each size n, alphabet size k and kind of expression we give the average value (first row) and the maximum value (second row) of the maximal size of partial derivatives. It is clear that the average sizes for expressions in $\mathcal{S}_k$ are significantly smaller than in $\mathcal{R}_k$, and the sizes decrease as the alphabet size grows. These results corroborate the analytic combinatorial analysis undertaken in the previous sections.

8 Tree-like Representation for $\mathcal{S}_k$

In this section, we consider regular expressions in $\mathcal{S}_k$ and we give a compact tree-like representation for their set of partial derivatives w.r.t. a word. We consider each regular expression as a tree such that binary operators ($\cdot$ and $+$, which is represented as $\odot$) have a left (ℓ) and a right (r) child, and the unary operators ($\star$ and ?) only one child (c). Let $\oplus$ be an operator (node) that represents a set and has an arbitrary number of children (c). For simplicity, suppose that the alphabet is unary, i.e., $k = 1$. Figure 1 shows the tree-like structures that represent the set of partial derivatives w.r.t. a symbol, $\partial_\sigma(\beta)$, for each type of expression $\beta \in \mathcal{S}_k$. For each edge labeled by ∂, its source node is an expression β' and its target the root of $\partial_\sigma(\beta')$. Nodes labeled by $\bullet$ are

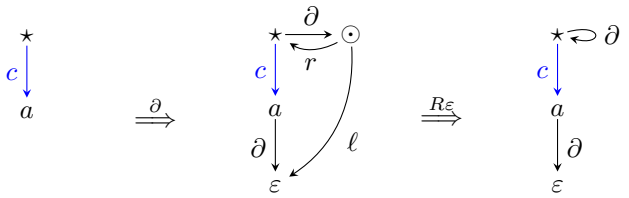
Fig. 2 Rewriting rules to build a set of partial derivatives.



the root nodes of the corresponding subexpression. Note that we can also use sharing of equal subexpressions (at least for the leaf nodes).

Moreover we need the rewriting rules presented in Figure 2 to deal with the union of two sets, the concatenation of an expression with a set, and the concatenation of an expression with the empty word. Subexpression sharing should also be used. The following examples illustrate the use of this representation.

Example 2 Consider the regular expression $a^\star$. In the below picture, the arrow $\overset{\partial}{\Longrightarrow}$ points to the partial derivative minus the rewriting rule $R\varepsilon$, i.e., $\partial_a(a^\star) = \partial_a(a) \odot a^\star$ Then, the arrow $\overset{R\varepsilon}{\Longrightarrow}$ points to the final result after applying the rewriting rule $R\varepsilon$, i.e., $\partial_a(a^\star) = a^\star$.

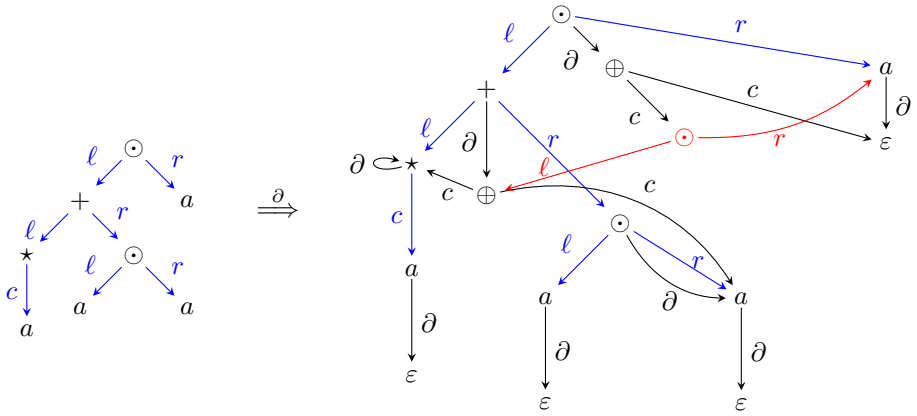


□

Example 3 Consider the regular expression $(a^\star + aa)a$. In the below picture, the arrow $\overset{\partial}{\Longrightarrow}$ points to its partial derivative having applied the rewriting rule $R\varepsilon$ but not rule RC. Using the notations of the tree-like structure and simplifying the partial derivative of subexpressions, that tree corresponds to

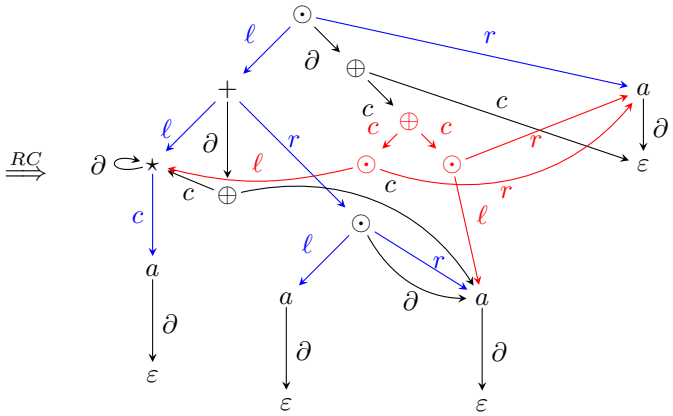
the following computation

$$\begin{aligned}\partial_a((a^\star + a \odot a) \odot a) &= (\partial_a(a^\star + a \odot a) \odot a) \oplus \partial_a(a)\\ &= ((\partial_a(a^\star) \oplus \partial_a(a \odot a)) \odot a) \oplus \partial_a(a)\\ &= ((a^\star \oplus (\partial_a(a) \odot a)) \odot a) \oplus \varepsilon\\ &= ((a^\star \oplus (\varepsilon \odot a)) \odot a) \oplus \varepsilon\\ &= ((a^\star \oplus a) \odot a) \oplus \varepsilon\end{aligned}$$



Then, below, the arrow $\overset{RC}{\Longrightarrow}$ points to the result of applying the rewriting rule RC, i.e.,

$$\begin{aligned}\partial_a((a^\star + a \odot a) \odot a) &= ((a^\star \oplus a) \odot a) \oplus \varepsilon\\ &= ((a^\star \odot a) \oplus (a \odot a)) \oplus \varepsilon\end{aligned}$$



We do not show the result of applying the rewriting rule RU. The result obtains if we replace each of the two branches $\odot \xrightarrow{\partial} \oplus \xrightarrow{c} \oplus \xrightarrow{c} \odot$ with $\odot \xrightarrow{\partial} \oplus \xrightarrow{c} \odot$, that is, remove the second $\oplus$ and connect the first $\oplus$ directly to the second $\odot$ with label c. □

Considering this tree-like representation, we see that when computing $\partial_\sigma(\beta)$, for $\sigma \in \Sigma$, the new nodes created are either nodes $\oplus$ or $\odot$. The number of $\oplus$ nodes corresponds to different partial derivatives whose number is bounded by the alphabetic size $|\beta|_\Sigma$ in the worst case; and the number of $\odot$ nodes corresponds to concatenations whose number is bounded as indicated in Proposition 3. In the next section, however, we will give an average estimate of an upper bound for the number of new concatenations when computing the partial derivatives w.r.t. a symbol.

8.1 Average Number of New Concatenations in Partial Derivatives w.r.t. a Symbol

In this section we estimate the average number of new concatenations when computing $\partial_\sigma(\beta)$ for $\sigma \in \Sigma$ and $\beta \in \mathcal{S}_k$. Using (11), let u be the cost function of the number of partial derivatives w.r.t. any symbol of Σ and g the number of concatenations in all computed partial derivatives. We have the following for $\beta \in \mathcal{S}_k$:

$$\begin{aligned}
g(\varepsilon) &= 0, & u(\varepsilon) &= 0,\\
g(\sigma) &= 0, & u(\sigma) &= 1,\\
g(\beta+\beta') &= g(\beta)+g(\beta'), & u(\beta+\beta') &= u(\beta)+u(\beta),\\
g(\beta_{\overline{\varepsilon}}\beta) &= g(\beta_{\overline{\varepsilon}})+u(\beta_{\overline{\varepsilon}}), & u(\beta_{\overline{\varepsilon}}\beta) &= u(\beta_{\overline{\varepsilon}}),\\
g(\beta_{\varepsilon}\beta) &= g(\beta_{\varepsilon})+u(\beta_{\varepsilon})+g(\beta), & u(\beta_{\varepsilon}\beta) &= u(\beta_{\varepsilon})+u(\beta),\\
g(\beta^{\star}_{\overline{\varepsilon}}) &= g(\beta_{\overline{\varepsilon}})+u(\beta_{\overline{\varepsilon}}), & u(\beta^{\star}_{\overline{\varepsilon}}) &= u(\beta_{\overline{\varepsilon}}),\\
g(\beta^{?}_{\overline{\varepsilon}}) &= g(\beta_{\overline{\varepsilon}}), & u(\beta^{?}_{\overline{\varepsilon}}) &= u(\beta_{\overline{\varepsilon}}).
\end{aligned}$$

And, for convenience, the special cases for expressions $\beta_{\overline{\varepsilon}}$:

$$\begin{aligned}
g(\sigma) &= 0, & u(\sigma) &= 1,\\
g(\beta_{\overline{\varepsilon}}+\beta'_{\overline{\varepsilon}}) &= g(\beta_{\overline{\varepsilon}})+g(\beta'_{\overline{\varepsilon}}), & u(\beta_{\overline{\varepsilon}}+\beta_{\overline{\varepsilon}}) &= u(\beta_{\overline{\varepsilon}})+u(\beta_{\overline{\varepsilon}}),\\
g(\beta_{\overline{\varepsilon}}\beta) &= g(\beta_{\overline{\varepsilon}})+u(\beta), & u(\beta_{\overline{\varepsilon}}\beta) &= u(\beta_{\overline{\varepsilon}}),\\
g(\beta_{\varepsilon}\beta_{\overline{\varepsilon}}) &= g(\beta_{\varepsilon})+u(\beta_{\varepsilon})+g(\beta_{\overline{\varepsilon}}), & u(\beta_{\varepsilon}\beta_{\overline{\varepsilon}}) &= u(\beta_{\varepsilon})+u(\beta_{\overline{\varepsilon}}).
\end{aligned}$$

Note that for g we take in consideration the fact that whenever we have a concatenation of a set S of expressions with an expression β, we concatenate β with all elements of S. For instance if $\partial_\sigma(\beta_{\overline{\varepsilon}}) = \{\beta_1, \ldots, \beta_2\}$ we have

$$\partial_\sigma(\beta_{\overline{\varepsilon}})\beta = \{\beta_1 \cdot \beta, \ldots, \beta_2 \cdot \beta\}.$$

Let $U_k = U_k(z)$ and $\overline{U}_k = \overline{U}_k(z)$ be the cost generating functions for the measure u, associated with the expressions β and $\beta_{\overline{\varepsilon}}$, respectively. Analogously, let $G_k = G_k(z)$ and $\overline{G}_k(z) = \overline{G}_k(z)$ be the cost generating functions for the

measure g. Setting $T_k = B_k + \overline{B}_k$, one has

$$\begin{aligned}
U_k &= kz + 2zU_kT_k + z\overline{U}_kT_k + z(U_k - \overline{U}_k)T_k + zU_kB_k + 2z\overline{U}_k,\\
\overline{U}_k &= kz + z\overline{U}_kT_k + z(U_k - \overline{U}_k)\overline{B}_k + z\overline{U}_kB_k + 2z\overline{U}_k\overline{B}_k,\\
G_k &= 2zG_kT_k + z\overline{G}_kT_k + z\overline{U}_kT_k + z(G_k - \overline{G}_k)T_k + z(U_k - \overline{U}_k)T_k\\
&\quad + zG_kB_k + 2z\overline{G}_k + z\overline{U}_k,\\
\overline{G}_k &= z\overline{G}_kT_k + zU_k\overline{B}_k + 2z\overline{G}_k\overline{B}_k + z(G_k - \overline{G}_k)\overline{B}_k + z(U_k - \overline{U}_k)\overline{B}_k\\
&\quad + z\overline{G}_kB_k.
\end{aligned}$$

Using the technique expounded in the Section 4, one obtains the following polynomial in $\mathbb{Q}(k,z)[w]$, of which $w = G_k(z)$ is a root:

$$z\ell_k(z)^2\, w^3 + kzp_6(z)\, w^2 + kp_7(z)\, w + k^2z^2p_5(z),$$

where $\ell_k(z)$ is, once more, the polynomial given in (15).

Proceeding as above, using [11], see that we are dealing with case (4) of Theorem 2, and computing the respective constants, ρ, ν, b, one gets the following value for the asymptotic behaviour of the coefficients of $G_k(z)$:

$$[z^n]G_k(z) \underset{n\,\to\,\infty}{\sim} \frac{b_{G_k}}{2\sqrt{\pi}}\eta_k^{-n}n^{-\frac{3}{2}}, \tag{27}$$

where b_{G_k} is a function of k with an expression too cumbersome to write here, but that satisfies

$$b_{G_k} \underset{k\,\to\,\infty}{\sim} 14\sqrt{k}. \tag{28}$$

From this one now gets

Theorem 5 *The average number of an upper bound of new concatenations on all the partial derivatives by a single symbol of a regular expression in $\mathcal{S}_k$ is given by*

$$\frac{[z^n]G_k(z)}{[z^n](B_k(z) + \overline{B}_k(z))} \underset{n\,\to\,\infty}{\sim} \frac{b_{G_k}}{b_{B_k} + b_{\overline{B}_k}} \underset{k\,\to\,\infty}{\sim} 14. \tag{29}$$

Doing the same analysis for U_k one gets an upper bound of the average number of partial derivatives of a regular expression in ssnf w.r.t. to all symbols, and we concluded that this number is, asymptotically, 6. This is exactly the same value that is known for standard regular expressions [7,25] and was already calculated in [10, p. 16].

9 Conclusions

We study the average size of partial derivatives. For regular expressions in strong star normal form, asymptotically and on average, the maximal size of partial derivatives of an expression is at most half the size of the expression. To obtain this result, a new technique had to be used in order to obtain the adequate algebraic curves of which the generating functions are branches.

This method can be useful in other situations. It was also used to show that, asymptotically and on average, the number of new concatenations on all partial derivatives w.r.t. a symbol is a function on the alphabet size and tends asymptotically to 14 (Theorem 5). The results in the paper indicate that, at least on average, a word membership algorithm based solely on partial derivatives could be of practical value. A detailed description of the algorithm as well as its theoretical analysis and practical performance have to be done in future work. A similar study for 2D-RE regular expressions following [21] is also planned for future research.

References

1. Adams, M.D., Hollenbeck, C., Might, M.: On the complexity and performance of parsing with derivatives. In: C. Krintz, E. Berger (eds.) Proc. 37th ACM SIGPLAN PLDI, pp. 224–236. ACM (2016). DOI 10.1145/2908080.2908128
2. Antimirov, V.M.: Partial derivatives of regular expressions and finite automaton constructions. Theoret. Comput. Sci. **155**(2), 291–319 (1996)
3. Backurs, A., Indyk, P.: Which regular expression patterns are hard to match? In: I. Dinur (ed.) Proc. 57th FOCS, pp. 457–466. IEEE Computer Society (2016). DOI 10.1109/FOCS.2016.56
4. Bille, P., Thorup, M.: Faster regular expression matching. In: S. Albers, A. Marchetti-Spaccamela, Y. Matias, S.E. Nikoletseas, W. Thomas (eds.) Proc. 36th ICALP, Part I, *LNCS*, vol. 5555, pp. 171–182. Springer (2009). DOI 10.1007/978-3-642-02927-1_16
5. Bringmann, K., Grønlund, A., Larsen, K.G.: A dichotomy for regular expression membership testing. In: C. Umans (ed.) Proc. 58th FOCS, pp. 307–318. IEEE Computer Society (2017). DOI 10.1109/FOCS.2017.36
6. Broda, S., Holzer, M., Maia, E., Moreira, N., Reis, R.: Mesh of automata. Information and Computation **265**, 94–111 (2019). DOI 10.1016/j.ic.2019.01.003
7. Broda, S., Machiavelo, A., Moreira, N., Reis, R.: On the average state complexity of partial derivative automata: an analytic combinatorics approach. Int. J. Found. Comput. Sci. **22**(7), 1593–1606 (2011). DOI 10.1142/S0129054111008908
8. Broda, S., Machiavelo, A., Moreira, N., Reis, R.: On the average size of Glushkov and partial derivative automata. Int. J. Found. Comput. Sci. **23**(5), 969–984 (2012). DOI 10.1142/S0129054112400400
9. Broda, S., Machiavelo, A., Moreira, N., Reis, R.: A hitchhiker's guide to descriptional complexity through analytic combinatorics. Theoret. Comput. Sci. **528**, 85–100 (2014)
10. Broda, S., Machiavelo, A., Moreira, N., Reis, R.: On average behaviour of regular expressions in strong star normal form. Int. J. Found. Comput. Sci. **30**(6-7), 899–920 (2019). DOI 10.1142/S0129054119400227
11. Broda, S., Machiavelo, A., Moreira, N., Reis, R.: Analytic combinatorics and descriptional complexity of regular languages on average. ACM SIGACT News **51**(1), 38–56 (2020). DOI 10.1145/3388392.3388400. SIGACT News Complexity Theory Column 104
12. Brüggemann-Klein, A.: Regular expressions into finite automata. Theoret. Comput. Sci. **48**, 197–213 (1993)
13. Champarnaud, J., Ziadi, D.: From c-continuations to new quadratic algorithms for automaton synthesis. Intern. Journ. of Alg. and Comp. **11**(6), 707–736 (2001). DOI 10.1142/S0218196701000772
14. Champarnaud, J.M., Ouardi, F., Ziadi, D.: Normalized expressions and finite automata. Intern. Journ. of Alg. and Comp. **17**(1), 141–154 (2007). DOI 10.1142/S021819670700355X
15. Champarnaud, J.M., Ziadi, D.: From Mirkin's prebases to Antimirov's word partial derivatives. Fundam. Inform. **45**(3), 195–205 (2001)
16. Cochran, W.G.: Sampling Techniques, third edn. John Wiley and Sons (1977)
17. Flajolet, P., R.Sedgewick: Analytic Combinatorics. CUP (2008)

18. Gulan, S.: On the relative descriptional complexity of regular expressions and finite automata. Ph.D. thesis, Universität Trier (2011)
19. Hille, E.: Analytic Function Theory, vol. 2. Blaisdell Publishing Company (1962)
20. Khorsi, A., Ouardi, F., Ziadi, D.: Fast equation automaton computation. J. Discrete Algorithms **6**(3), 433–448 (2008). DOI 10.1016/j.jda.2007.10.003
21. Konstantinidis, S., Machiavelo, A., Moreira, N., Reis, R.: On the average state complexity of partial derivative transducers. In: A. Chatzigeorgiou, R. Dondi, H. Herodotou, C.A. Kapoutsis, Y. Manolopoulos, G.A. Papadopoulos, F. Sikora (eds.) Proc. SOFSEM 2020, *LNCS*, vol. 12011, pp. 174–186. Springer (2020). DOI 10.1007/978-3-030-38919-2_15
22. Lokshtanov, D., Marx, D., Saurabh, S.: Lower bounds based on the exponential time hypothesis. Bull. EATCS **105**, 41–72 (2011)
23. Mirkin, B.G.: An algorithm for constructing a base in a language of regular expressions. Eng. Cybernetics **5**, 51—57 (1966)
24. Myers, E.W.: A four russians algorithm for regular expression pattern matching. J. ACM **39**(2), 430–448 (1992). DOI 10.1145/128749.128755
25. Nicaud, C.: On the average size of Glushkov's automata. In: A. Dediu, A.M. Ionescu, C.M. Vide (eds.) Proc. 3rd LATA, *LNCS*, vol. 5457, pp. 626–637. Springer (2009)
26. Project FAdo: tools for formal languages manipulation. `http://fado.dcc.fc.up.pt` (Access date:1.1.2021)
27. Thompson, K.: Regular expression search algorithm. CACM **11**(6), 410–422 (1968)

Table 1 Average maximal sizes of partial derivatives.

k	n	$\max \lvert\partial(\alpha)\rvert$	$\max \lvert\partial(\mathsf{ssnf})\rvert$	$\frac{\max \lvert\partial(\alpha)\rvert}{\lvert\alpha\rvert}$	$\frac{\max \lvert\partial(\mathsf{ssnf})\rvert}{\lvert\mathsf{ssnf}\rvert}$
1	100	250 1054	137 427	2.5	1.37
	200	666 2504	324 1298	3.33	1.62
	300	1186 5123	542 2289	3.95	1.81
	500	2471 9938	1051 4030	4.94	2.10
	1000	6739 22756	2683 9455	6.74	2.68
	2000	18762 65140	7007 26887	9.38	3.5
2	100	218 1010	130 479	2.18	1.30
	200	574 2293	307 1290	2.87	1.54
	300	1007 3741	512 2167	3.36	1.71
	500	2091 7343	985 3642	4.18	1.97
	1000	5724 19880	2492 9468	5.72	2.49
	2000	15763 50956	6446 23668	7.88	3.22
10	100	152 675	115 466	1.52	1.15
	200	377 1803	265 948	1.89	1.33
	300	636 2708	440 1871	2.12	1.47
	500	1272 6081	821 3790	2.54	1.64
	1000	3365 14410	2019 7957	3.37	2.02
	2000	8962 39553	5002 20208	4.48	2.5
50	500	802 2564	677 3534	1.60	1.35
	1000	1969 7895	1574 7091	1.97	1.57
	2000	4931 23911	3755 18424	2.47	1.88