A Separation Between Run-Length SLPs and LZ77

Philip Bille Travis Gagie Inge Li Gørtz Nicola Prezza

November 21, 2017

Abstract

In this paper we give an infinite family of strings for which the length of the Lempel-Ziv'77 parse is a factor $\Omega(\log n/\log\log n)$ smaller than the smallest run-length grammar.

1 Introduction

The Lempel-Ziv factorization of a text [9] (LZ77) is a greedy left-to-right parse in maximal factors such that each factor already occurred to the left. Despite its simplicity, LZ77 can be easily shown to be optimal among all unidirectional parses (i.e. that copy phrases from left-to-right), and dominates other popular compression schemes such as Straight Line Programs (SLPs), i.e., context-free grammars that generate only the text as output. Let z_{no} be the number of phrases of the Lempel-Ziv parse when overlaps are not allowed between phrases and their sources, and let g^* be the size of the smallest SLP. Charikar at al. [5] and Rytter [12] showed how to obtain a unidirectional parse of size at most g starting from a SLP of size g. It follows from the optimality of LZ77 that the relation $z_{no} \leq g^*$ holds. On the other hand, Charikar at al. [5] showed an infinite family of strings for which $g^*/z_{no} = \Omega(\log n/\log\log n)$, where n is the length of the string. Together, these results imply that LZ77 compression without overlaps is always at least as good as grammar compression, and strictly better in some cases.

Given that, in fields such as compressed computation, SLPs are often easier to treat than LZ77, one might wonder whether we could enhance SLPs so that they become as powerful as Lempel-Ziv compression. See, for example, Bille et al. [4, Thm 1.1] and Kreft ad Navarro [8, Thm 4.11] for classical solutions to the random access problem on grammar- and Lempel-Ziv-compressed texts, respectively. One possible extension of SLPs is to add so-called run-length rules, i.e. rules of the form $X \to Y^\ell$, for $\ell > 1$ (meaning that X expands to ℓ repetitions of Y). This extension takes the name *run-length SLP*, or RLSLP in what follows [11]. Let g^*_{rl} be the size of the smallest RLSLP. It is easy to show that $g^* = \Theta(\log n)$ and $g^*_{rl} = O(1)$ on unary strings of length n. This implies that RLSLPs are a strict improvement over SLPs. Since $z_{no} \in \Theta(\log n)$ on unary strings, we also have that $z_{no}/g^*_{rl} = \Theta(\log n)$ for an infinite class of strings: RLSLPs improve upon Lempel-Ziv compression in some cases, and therefore are good candidates for capturing it. However, a slight modification to the LZ77 compression scheme adds enough power to capture, again, grammar compression with run-length rules. Let z be the number of phrases of the Lempel-Ziv parse when overlaps are allowed between phrases and their sources. By adapting Rytter's proof, Gagie et al. in [7] proved that $z \leq g^*_{rl}$, which implies that we cannot hope to beat LZ77 with overlaps using RLSLPs.

The missing piece in the puzzle is the following: are RLSLPs always at least as good as Lempel-Ziv (with or without overlaps)? In this paper, we answer negatively to this question. By adapting Charikar at al.'s proof [5], we give an infinite family of strings for which $g^*_{rl}/z_{no} = \Omega(\log n/\log\log n)$. Since $z \leq z_{no}$ trivially holds, our result implies that Lempel-Ziv compression with overlaps is always at least as good as grammar-compression with run-length rules, and strictly better in some cases. Formally, we prove the following theorem.

Theorem 1 *There exists an infinite family of strings for which the ratio between the size of the*

smallest RLSLP and the length of the LZ77 parse is

$$\frac{g^*_{rl}}{z_{no}} = \Omega\left(\frac{\log n}{\log\log n}\right).$$

2 Preliminaries

Charikar et al. Charikar et al. [5] showed a separation between the smallest grammar and the size of the LZ77 parse of a string.

Lemma 2 (Charikar et al.) *There exists an infinite family of strings for which the ratio between size of the smallest grammar and the length of the LZ77 parse is*

$$\frac{g^*}{z_{no}} = \Omega\left(\frac{\log n}{\log\log n}\right).$$

The proof is based on the following lemma (implicit in the paper) that they proved using a link between grammars and addition chains.

Lemma 3 (Charikar et al.) *Let $k_1, \ldots, k_p$ be a set of distinct positive integers, and consider strings of the form $s = x^{k_1}|_1 x^{k_2}|_2 \ldots |_{p-1} x^{k_p}$, where k_1 is the largest of the k_i. Let $p = \Theta(\log k_1)$. There exists an infinite class of sequences of integers $k_1, \ldots, k_p$ such that the smallest grammar for s has size*

$$\Omega\left(\frac{\log^2 k_1}{\log\log k_1}\right).$$

Since the LZ77 parse for the string has size $O(p + \log k_1) = O(\log k_1)$ Lemma 2 follows.

Thue-Morse Sequence The Thue-Morse sequence can be generated by starting with 01 and keep appending the inverse binary negation of the sequence already generated:

$$01 \to 0110 \to 01101001 \to 0110100110010110 \to \ldots$$

The Thue-Morse sequence is overlapfree [1, 2, 13, 14], and therefore also cubefree on two symbols [10]. We can obtain a squarefree sequence on three symbols by taking the first difference of the Thue-Morse sequence: take the Thue-Morse sequence

0110100110010110100101100110100 1...

and form a new sequence in which each term is the difference of two consecutive terms in the Thue-Morse sequence

1 0 -1 1 -1 0 1 0 -1 0 1 -1 1 0 -1 1 -1 0 1 -1 1 0 -1 0 1 0 -1 1 -1 0 1...

This sequence is squarefree (see e.g. [1–3]). We call this sequence the Diff-Thue-Morse sequence.

3 Separation

Size of smallest RLSLP Let $t(n)$ be the prefix of length n of the infinite Diff-Thue-Morse sequence. Let $k_1, \ldots, k_p$ be a set of distinct positive integers, and consider strings of the form

$$\hat{s} = t(k_1)|_1 t(k_2)|_2 \ldots |_{p-1} t(k_p),$$

where k_1 is the largest of the k_i.

Since the sequences $t(k_i)$ are squarefree, there is no difference in the size of the smallest grammar and the smallest RLSLP for the string $\hat{s}$.

Let $s = x^{k_1}|_1 x^{k_2}|_2 \ldots |_{p-1} x^{k_p}$. Assume you have a grammar of size g for $\hat{s}$. Replacing all the terminals $(-1, 0, 1)$ by x gives you a grammar for s of size g. Thus the smallest grammar for $\hat{s}$ must be at least the size of the smallest grammar for s. From Lemma 3 we know that there exists exists integers $k_1, \ldots, k_q$ such that the smallest grammar for s has size $\Omega\left(\frac{\log^2 k_1}{\log\log k_1}\right)$. It follows that the smallest RLSLP for $\hat{s}$ has size at least

$$\Omega\left(\frac{\log^2 k_1}{\log\log k_1}\right).$$

Size of LZ77 parse The LZ77 parse for the Thue-Morse sequence of length n has size $O(\log n)$ [6]. The LZ77 parse for the Diff-Thue-Morse sequence $t(n)$ is at most 2 times larger than the LZ77 parse, z_t, for the corresponding Thue-Morse sequence $t_m(n+1)$: We can construct a parse $\hat{z}_t$ of size at most $2|z_t|$ such that each phrase f in z_t gives at most 2 phrases in $\hat{z}_t$. Consider a phrase f in z_t. Since phrase f exists earlier in $t_m(n+1)$, then the sequence of the differences between the terms in f exists previously in t and we construct a corresponding phrase in $\hat{z}$ of length $|f|-1$. The term denoting the difference between the first position in f and the last position in the previous phrase is in its own phrase. The parse $\hat{z}_t$ has size at most $2|z_t|$ and thus the LZ77 parse of $t(n)$ has size at most $2|z_t|$, since the LZ77 parse is optimal. It follows that the LZ parse of $t(n)$ has size $O(\log n)$.

Now consider the string $\hat{s}$. The LZ77 parse of $\hat{s}$ is then $z_1|_1(1, k_2)|_2 \ldots |_{p-1}(1, k_p)$. The size of the parse is $O(\log k_1 + p) = O(\log k_1)$. The ratio between the smallest RLSLP and the length of the LZ77 parse is therefore

$$\Omega\left(\frac{\log k_1}{\log\log k_1}\right) = \Omega\left(\frac{\log n}{\log\log n}\right).$$

References

[1] J.-P. Allouche and J. Shallit. The ubiquitous prouhet-thue-morse sequence, 1999.

[2] Jean Berstel, Aaron Lauve, Christophe Reutenauer, and Franco V. Saliola. *Combinatorics on words. Christoffel words and repetitions in words.* Providence, RI: American Mathematical Society (AMS), 2009.

[3] Jean Berstel and Dominique Perrin. The origins of combinatorics on words. *European Journal of Combinatorics*, 28(3):996 – 1022, 2007.

[4] Philip Bille, Gad M Landau, Rajeev Raman, Kunihiko Sadakane, Srinivasa Rao Satti, and Oren Weimann. Random access to grammar-compressed strings and trees. *SIAM Journal on Computing*, 44(3):513–539, 2015.

[5] Moses Charikar, Eric Lehman, Ding Liu, Rina Panigrahy, Manoj Prabhakaran, Amit Sahai, and Abhi Shelat. The smallest grammar problem. *IEEE Transactions on Information Theory*, 51(7):2554–2576, 2005.

[6] Sorin Constantinescu and Lucian Ilie. The lempel-ziv complexity of fixed points of morphisms. In *MFCS*, volume 4162 of *Lecture Notes in Computer Science*, pages 280–291. Springer, 2006.

[7] Travis Gagie, Gonzalo Navarro, and Nicola Prezza. Optimal-time text indexing in bwt-runs bounded space. *arXiv preprint arXiv:1705.10382*, 2017.

[8] Sebastian Kreft and Gonzalo Navarro. On compressing and indexing repetitive sequences. *Theoretical Computer Science*, 483:115–133, 2013.

[9] Abraham Lempel and Jacob Ziv. On the complexity of finite sequences. *IEEE Transactions on information theory*, 22(1):75–81, 1976.

[10] M.Morse and G. A. Hedlund. Unending chess, symbolic dynamics, and a problem in semigroups. 11:1–7, 1944.

[11] Takaaki Nishimoto, Shunsuke Inenaga, Hideo Bannai, Masayuki Takeda, et al. Fully dynamic data structure for lce queries in compressed space. *arXiv preprint arXiv:1605.01488*, 2016.

[12] Wojciech Rytter. Application of lempel–ziv factorization to the approximation of grammar-based compression. *Theoretical Computer Science*, 302(1-3):211–222, 2003.

[13] A. Thue. Über unendliche zeichenreihen. *Norske vid. Selsk. Skr. Mat. Nat. Kl.*, 7:1–22, 1906.

[14] A. Thue. Über die gegenseitige lage gleicher teile gewisser zeichenreihen. *Norske vid. Selsk. Skr. Mat. Nat. Kl.*, (1):1–67, 1912.