DTU Library

A separation between RLSLPs and LZ77

Bille, Philip; Gagie, Travis; Gørtz, Inge Li; Prezza, Nicola

Published in:
Journal of Discrete Algorithms

Link to article, DOI:
10.1016/j.jda.2018.09.002

Publication date:
2018

Document Version
Peer reviewed version

Citation (APA):
Bille, P., Gagie, T., Gørtz, I. L., & Prezza, N. (2018). A separation between RLSLPs and LZ77. *Journal of Discrete Algorithms*, *50*, 36-39. https://doi.org/10.1016/j.jda.2018.09.002

A Separation Between RLSLPs and LZ77

Philip Bille, Travis Gagie, Inge Li Gørtz, Nicola Prezza

PII: S1570-8667(18)30138-2
DOI: https://doi.org/10.1016/j.jda.2018.09.002
Reference: JDA 704

To appear in: *Journal of Discrete Algorithms*

Received date: 11 December 2017
Revised date: 25 August 2018
Accepted date: 6 September 2018

Please cite this article in press as: P. Bille et al., A Separation Between RLSLPs and LZ77, *J. Discret. Algorithms* (2018), https://doi.org/10.1016/j.jda.2018.09.002

A Separation Between RLSLPs and LZ77☆

Philip Bille[a], Travis Gagie[b], Inge Li Gørtz[a,*], Nicola Prezza[a]

[a]DTU Compute, Technical University of Denmark, Lyngby, Denmark
[b]EIT, Universidad Diego Portales, Santiago, Chile

Abstract

In their ground-breaking paper on grammar-based compression, Charikar et al. (2005) gave a separation between straight-line programs (SLPs) and Lempel-Ziv '77 (LZ77): they described an infinite family of strings such that the size of the smallest SLP generating a string of length n in that family, is an $\Omega(\log n/\log\log n)$-factor larger than the size of the LZ77 parse of that string. However, the strings in that family have run-length SLPs (RLSLPs) — i.e., SLPs in which we can indicate many consecutive copies of a symbol by only one copy with an exponent — as small as their LZ77 parses. In this paper we modify Charikar et al.'s proof to obtain the same $\Omega(\log n/\log\log n)$-factor separation between RLSLPs and LZ77.

Keywords: grammar-based compression; run-length compression; SLP; RLSLP; LZ77; Thue-Morse sequence

1. Introduction

Storing and processing massive datasets has become a fundamental task of modern computer science and inspired a renaissance in data compression. Most datasets are highly repetitive and so dictionary- and grammar-based compression algorithms often achieve dramatic results, with the added benefit that some

☆PB, ILG and NP are funded by DFF grant 4005-00267. TG is funded by FONDECYT grant 1171058.

*Corresponding author

Email addresses: phbi@dtu.dk (Philip Bille), travis.gagie@gmail.com (Travis Gagie), inge@dtu.dk (Inge Li Gørtz), nicola.prezza@gmail.com (Nicola Prezza)

are "computation-friendly" in the sense that we can perform many calculations faster on compressed datasets than on uncompressed ones. However, traditional techniques for analyzing compression — developed mainly for statistical compressors and based on empirical entropy or the expected compressibility of strings generated by Markov sources — do not accurately predict how well we can compress repetitive datasets.

Charikar et al. [1] changed the course of this line of research by proving upper and lower bounds relating the sizes of several compressed representations, to the size of the input's Lempel-Ziv '77 (LZ77) parse [2] and to the size of its smallest straight-line program (SLP). The LZ77 parse of a text is a greedy left-to-right parse into 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 SLPs, which are context-free grammars that generate only the text as output, and other popular compression schemes.

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. [1] and Rytter [3] 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, they showed how to build an SLP of size $O(z_{no} \log(n/z_{no}))$ from the LZ77 parse, so g^* has at most that size. Finally, Charikar et al. 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. 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 SLPs are often more computation-friendly 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 and Navarro [5, 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. RLSLPs were formally introduced by Nishimoto et al. [6], who used them in a compressed data structure for computing longest common extensions, although Jeż [7] used a similar idea earlier in his paper on approximating the smallest SLP via recompression. Interestingly, Jeż's construction gives a balanced RLSLP, which becomes unbalanced when it is converted into a standard SLP. Very recently, Gagie, Navarro and Prezza [8] used RLSLPs in a data structure supporting fast random access to compressed strings.

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. [8] proved that $z \leq 2g^*_{rl}$, implying we cannot hope to significantly beat LZ77 with overlaps using RLSLPs, but they did not give a separation between z and g^*_{rl}; for strings in the family Charikar et al. described, $g^*_{rl} = O(z_{no})$

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 [1], 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 be-*

tween the size of the smallest RLSLP and the length of the LZ77 parse is

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

We note that since Charikar et al.'s and Rytter's work, other researchers simplified the proof that $g^* \in O(z_{no}\log(n/z_{no}))$ and strengthened it to show $g^* \in O(z\log(n/z))$ [7], and described an infinite family of *binary* strings for which $g^*/z_{no} = \Omega(\log n/\log\log n)$ [9]. In collaboration, these authors and others [10] have recently independently proposed ideas similar to some of the ones we describe in this paper: specifically, they used the cube-freeness of the Thue-Morse sequence to show there is an infinite family of strings — prefixes of the Thue-Morse sequence — whose minimum RLSLPs are not asymptotically smaller than their minimum SLPs. This does not separate RLSLPs from LZ77, however, since those strings' LZ77 parses are not asymptotically smaller than their minimum SLPs, either.

Now that we know the Charikar et al.'s separation between SLPs and LZ77 can be made robust with respect either to alphabet size or to run-length encoding of symbols in rules, an obvious open problem is strengthening the results in this paper to hold for strings over small alphabets, ideally binary. In the longer term, we feel researchers should revisit several other natural generalizations of SLPs that Charikar et al. proposed in the conference version of their paper and claimed to show not to have significantly greater power, "suggest[ing] the robustness of grammar based string complexity"; the proofs seemed fragile and those sections were omitted from the final version.

2. Prelimaries

Charikar et al. [1] 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 [11, 12, 13, 14], and therefore also cubefree on two symbols [15]. We denote the infinite Thue-Morse sequence as t in the following.

3. Separation

Size of smallest RLSLP. Let $t(n)$ be the prefix of length n of the infinite 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 cubefree, there is no difference in the size of the smallest grammar and the smallest RLSLP for the string $\hat{s}$. To see why this holds, consider any RLSLP for $\hat{s}$. Since $\hat{s}$ is cubefree, for any rule of the form $X \to Y^\ell$ it must be the case that $\ell = 2$. Then, we can convert all rules of this kind to the form $X \to YY$ and obtain a SLP for $\hat{s}$ of the same size.

Let $s = x^{k_1}|_1 x^{k_2}|_2 \ldots |_{p-1} x^{k_p}$. Assume we have a grammar of size g for $\hat{s}$. Replacing all the terminals $(-1, 0, 1)$ by x gives us 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 exist integers $k_1, \ldots, k_q$, with $q \in \Theta(\log k_1)$ and k_1 being the largest integer in the sequence, 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 SLP (and thus RLSLP, for the above considerations) 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 $O(\log n)$ [17]. Now consider the string $\hat{s}$, and let z_1 be the LZ77 parse of $t(k_1)|_1$, of size $O(\log k_1)$. The LZ77 parse of $\hat{s}$ is then z_1 followed by $(1, k_2)|_2 \ldots |_{p-1}(1, k_p)$. 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

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

[1] M. Charikar, E. Lehman, D. Liu, R. Panigrahy, M. Prabhakaran, A. Sahai, a. shelat, The smallest grammar problem, IEEE Transactions on Information Theory 51 (7) (2005) 2554–2576.

[2] J. Ziv, A. Lempel, A universal algorithm for sequential data compression, IEEE Trans. Information Theory 23 (3) (1977) 337–343.

[3] W. Rytter, Application of Lempel–Ziv factorization to the approximation of grammar-based compression, Theoretical Computer Science 302 (1-3) (2003) 211–222.

[4] P. Bille, G. M. Landau, R. Raman, K. Sadakane, S. R. Satti, O. Weimann, Random access to grammar-compressed strings and trees, SIAM Journal on Computing 44 (3) (2015) 513–539.

[5] S. Kreft, G. Navarro, On compressing and indexing repetitive sequences, Theoretical Computer Science 483 (2013) 115–133.

[6] T. Nishimoto, T. I, S. Inenaga, H. Bannai, M. Takeda, Fully Dynamic Data Structure for LCE Queries in Compressed Space, in: P. Faliszewski, A. Muscholl, R. Niedermeier (Eds.), 41st International Symposium on Mathematical Foundations of Computer Science (MFCS 2016), Vol. 58 of LIPIcs, Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik, Dagstuhl, Germany, 2016, pp. 72:1–72:15.

[7] A. Jeż, Approximation of grammar-based compression via recompression, Theoretical Computer Science 592 (2015) 115–134.

[8] T. Gagie, G. Navarro, N. Prezza, Optimal-time text indexing in BWT-runs bounded space, in: Proceedings of the Twenty-Ninth Annual ACM-SIAM Symposium on Discrete Algorithms, Society for Industrial and Applied Mathematics, 2018, pp. 1459–1477.

[9] D. Hucke, M. Lohrey, C. P. Reh, The smallest grammar problem revisited, in: Proceedings of the 23rd Symposium on String Processing and Information Retrieval (SPIRE), 2016, pp. 35–49.

[10] H. Bannai, D. Hucke, A. Jeż, M. Lohrey, C. P. Reh, Personal communication.

[11] A. Thue, Über unendliche zeichenreihen, Norske vid. Selsk. Skr. Mat. Nat. Kl. 7 (1906) 1–22.

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

[13] J.-P. Allouche, J. Shallit, The ubiquitous Prouhet-Thue-Morse sequence (1999).

[14] J. Berstel, A. Lauve, C. Reutenauer, F. V. Saliola, Combinatorics on words. Christoffel words and repetitions in words., Providence, RI: American Mathematical Society (AMS), 2009.

[15] M. Morse, G. A. Hedlund, Unending chess, symbolic dynamics, and a problem in semigroups, Duke Math. J. 11 (1944) 1–7.

[16] J. Berstel, D. Perrin, The origins of combinatorics on words, European Journal of Combinatorics 28 (3) (2007) 996 – 1022.

[17] S. Constantinescu, L. Ilie, The lempel–ziv complexity of fixed points of morphisms, SIAM Journal on Discrete Mathematics 21 (2) (2007) 466–481.