Open Problems on Prefix Normal Words

Zsuzsanna Lipták,
zsuzsanna.liptak@univr.it

last updated: 28 May 2021

Joint work with:
Péter Burcsi, Gabriele Fici, Rajeev Raman, Frank Ruskey, Joe Sawada

A binary word is called *prefix normal* if no substring has more 1s than the prefix of the same length. For example, 1101010110 is prefix normal but 1100110110 is not, because, for instance, the substring 1101 has too many 1s. These words are the sequences of the first differences of the function $F(w,k) = \max\{d(u) \mid u \text{ is a substring of } w \text{ of length } k\}$, where $d(u)$ denotes the number of 1s in the binary word u. For example, for the word $w = 1100110110$, we get:

k	0	1	2	3	4	5	6	7	8	9	10
$F(w,k)$	0	1	2	2	3	4	4	4	5	6	6

The sequence of the first differences is $w' = 1101100110 =: \text{PNF}(w)$. We call this (necessarily binary) word the *prefix normal form* of w. It is prefix normal by construction, and it is the only prefix normal word in the equivalence class of w w.r.t. the equivalence $w \equiv v$ iff $F(w,\cdot) = F(v,\cdot)$.

Prefix normal words, fist introduced in [10], are motivated by *Binary Jumbled Pattern Matching (BJPM)* [7]: Given a binary string T and a query (x,y), does T contain a substring with x zeros and y ones? BJPM can be solved by a linear scan of T in time $O(n)$. For the indexing variant (IBJPM), define $f(w,k)$ as the *minimum* number of 1s in a substring of w of length k. Then the answer to query (x,y) is YES if and only if $f(T,x+y) \le y \le F(T,x+y)$. Thus the problem can be solved by storing $F(w,\cdot)$ and $f(w,\cdot)$ in $O(n)$ space; this is a linear size index, with $O(1)$ query time. The current fastest computation of these functions F and f is given by Chan and Lewenstein, in $O(n^{1.864})$ time [6].

1. **Enumeration of prefix normal words (counting).**

   Let $pnw(n)$ denote the number of prefix normal words of length n. It is easy to see that pnw grows exponentially. It was conjectured in [5], and recently proved by Balister and Gerke [2] that $pnw(n) = 2^{n-\Theta(\log^2 n)}$.

   No closed form is known for $pnw(n)$; OEIS sequence number A194850 [13] lists $pnw(n)$ up to $n = 50$. Generating functions for some (few) subsets exist [5], but not for $pnw(n)$.

   The question can be rephrased as: How many different functions F can exist, where a necessary and sufficient condition for a 0-1 step function F to be the $F(w,\cdot)$ of some word w is that $F(w,0) = 0$ and for all $i < j, F(i+j) \le F(i) + F(j)$ [10, 5].

2. **Equivalence class sizes.**

   Some equivalence classes are singletons (e.g. $1^n, 0^n, 1001, \ldots$; this implies that the word is a palindrome, since $F(w,\cdot) = F(w^{\text{rev}},\cdot)$), some are much larger. Balister and Gerke proved that the size of the largest equivalence class is $2^{n-O(\sqrt{n \log n})}$ [2].

However, no closed form is known. The OEIS sequence A238110 [13] lists the size of the largest equivalence class for n up to 50. This question is the same as asking how many distinct words can have the same function F.

3. **Generation of equivalence classes.**

   Given a prefix normal word w, list all words v s.t. $\mathrm{PNF}(v) = w$. Some attempts in this direction were made recently in [11].

4. **Testing.**

   The best algorithm to decide whether a string w is prefix normal is: Compute $\mathrm{PNF}(w)$; w is prefix normal iff $\mathrm{PNF}(w) = w$. The fastest algorithm for doing this is given in [6]. However, it is not clear that *recognition* is as hard as computing the F-function.

5. **Which prefix normal forms w.r.t. 1 can be combined with which prefix normal forms w.r.t. 0?**

   Define $F_0(w, \cdot)$ and $\mathrm{PNF}_0(w)$ analogously to above, but w.r.t. 0 instead of 1. (For constructing $\mathrm{PNF}_0(w)$, we put a 0 when F_0 increases, and a 1 otherwise.) Then the two prefix normal forms of w encode the index for BJPM. These can be used to answer BJPM queries as follows:

   $$(x, y) \text{ is a YES-query} \Leftrightarrow \mathrm{rank}_1(\mathrm{PNF}_0(w), x + y) \leq y \leq \mathrm{rank}_1(\mathrm{PNF}_1(w), x + y).$$

   Prefix normal words w.r.t. 0 are defined analogously to prefix normal words w.r.t. 1. Given w, a prefix normal word w.r.t. 1, and w', a prefix normal word w.r.t. 0, we call w and w' *compatible* if there exists a binary word v s.t. $w = \mathrm{PNF}_1(v)$ and $w' = \mathrm{PNF}_0(v)$.

   The open problem is: Which prefix normal words w.r.t. 1 are compatible with which prefix normal words w.r.t. 0?

6. **How big are the Parikh-equivalence classes?**

   Another equivalence relation is given by: w Parikh-equivalent to v iff $\mathrm{PNF}_1(w) = \mathrm{PNF}_1(v)$ and $\mathrm{PNF}_0(w) = \mathrm{PNF}_0(v)$. Note that this holds iff the Parikh sets of w and v are the same, where the *Parikh set* of a string is the set of Parikh vectors of its substrings. How big are these equivalence classes? I.e. how many different strings can have the same Parikh set?

   Similar results about the multiset (not set) of Parikh vectors of substrings can be found in [1].

7. (**Expected critical prefix length of a random prefix normal word of length n. -** Solved)

   Let the *critical prefix* of a binary word be defined as the sum of the lengths of the first run of 1s (possibly empty) plus the first run of 0s, and $cr(w)$ be the length of w's critical prefix. E.g. $cr(1110001010) = 6$, $cr(0001100101) = 3$, $cr(1110000000) = 10$, $cr(1^n) = cr(0^n) = n$.

   The expected critical prefix length of a prefix normal word was conjectured to be $O(\log n)$ in [5], and recently proved to be $O(\log^2 n)$ [4]. We do not expect further advance here.

Further relevant literature includes:

- generating algorithms for prefix normal words: (a) in worst-case $O(n)$ time per word [8] (exploiting connections to lexicographic order); (b) in amortized $O(\log^2 n)$ time per word [4] (exploiting that prefix normal words form a bubble language);
- infinite prefix normal words and forms [9];
- prefix normal forms applied to a certain class of graphs [3],
- generalization of prefix normal words to larger alphabets [12].

References

[1] J. Acharya, H. Das, O. Milenkovic, A. Orlitsky, and S. Pan. String reconstruction from substring compositions. *SIAM J. Discrete Math.*, 29(3):1340–1371, 2015.

[2] P. Balister and S. Gerke. The asymptotic number of prefix normal words. *Theoret. Comput. Sci.*, 784:75–80, 2019.

[3] A. Blondin Massé, J. de Carufel, A. Goupil, M. Lapointe, É. Nadeau, and É. Vandomme. Leaf realization problem, caterpillar graphs and prefix normal words. *Theoret. Comput. Sci.*, 732:1–13, 2018.

[4] P. Burcsi, G. Fici, Zs. Lipták, R. Raman, and J. Sawada. Generating a Gray code for prefix normal words in amortized polylogarithmic time per word. *Theret. Comput. Sci.*, 842:86-99, 2020.

[5] P. Burcsi, G. Fici, Zs. Lipták, F. Ruskey, and J. Sawada. On prefix normal words and prefix normal forms. *Theoret. Comput. Sci.*, 659:1–13, 2017.

[6] T. M. Chan and M. Lewenstein. Clustered integer 3SUM via additive combinatorics. In *Proc. of the 47th Ann. ACM on Symp. on Theory of Computing (STOC 2015)*, pages 31–40, 2015.

[7] F. Cicalese, G. Fici, and Zs. Lipták. Searching for Jumbled Patterns in Strings. In *Proc. of Stringology 2009 (PSC 2009)*, pages 105-117, 2009. *Journal version:* P. Burcsi, F. Cicalese, G. Fici, and Zs. Lipták. Algorithms for Jumbled Pattern Matching in Strings. *Int. J. Found. Comput. Sci.*, 23(2): 357-374, 2012.

[8] F. Cicalese, Zs. Lipták, and M. Rossi. Bubble-flip - A new generation algorithm for prefix normal words. *Theoret. Comput. Sci.*, 743:38–52, 2018.

[9] F. Cicalese, Zs. Lipták, and M. Rossi. On infinite prefix normal words. *Theoret. Comput. Sci.*, 859: 134-148, 2021.

[10] G. Fici and Zs. Lipták. On prefix normal words. In *Proc. of the 15th Intern. Conf. on Developments in Language Theory (DLT 2011)*, vol. 6795 of *LNCS*, pages 228–238, 2011.

[11] P. Fleischmann, M. Kulczynski, D. Nowotka, and D. B. Poulsen. On collapsing prefix normal words. In *Proc. of the 14th Int. Conf. on Language and Automata Theory and Applications (LATA 2020)*, vol. 12038 of *LNCS*, pages 412–424, 2020.

[12] Y. Eikmeier, P. Fleischmann, M. Kulczynski, and D. Nowotka. Weighted Prefix Normal Words: Mind the Gap. In *Proc. of the 25th Int. Conf. on Developments in Language Theory (DLT 2021), to appear.*

[13] N. J. A. Sloane. The On-Line Encyclopedia of Integer Sequences. Available electronically at `http://oeis.org`.