SUDO-LYNDON

Gwénël Richomme
LaRIA, Université de Picardie Jules Verne
33 Rue Saint Leu, 80039 Amiens cedex 01, France
gwenael.richomme@u-picardie.fr
http://www.laria.u-picardie.fr/~richomme

Abstract

Based on Lyndon words, a new Sudoku-like puzzle is presented and some relative theoretical questions are proposed.

1 Introduction

Lyndon words are basic tools in Combinatorics on Words (see for instance [6, 7]) and appear in many problems that can be expressed using words (see for instance [4, 5, 9]). While preparing a course, I thought about the puzzle presented in Section 2 for a pleasant first contact with Lyndon words (see also Section 5 for some possible variants). I propose to nickname *Sudo-Lyndon* this puzzle since, as in the now famous (who did not hear about it?) Sudoku game, also known as Number Place puzzle, a grid has to be filled from partial informations, and only a unique solution is expected as a result (may be a better name should be *Lyndon place*).

2 The puzzle

For our purpose a *word* is nothing else than a finite *non-empty* sequence of letters: here the meaning of the word does not care.

A *Lyndon word* is a word smaller, with respect to the lexicographic order, than all its suffixes except itself (See for instance [6] for equivalent definitions and properties). For example, when considering the usual ordering of the letters, one can verify that *cocoon* or *acacias* are Lyndon words while *bananas*, *acacia*, *anagram* or *eighteen* are not.

Here we will play only with two letters a and b, but we will simultaneously consider Lyndon words over $\{a < b\}$ (that is over the alphabet $\{a, b\}$ with $a < b$)

and over $\{b < a\}$. The shortest Lyndon words over $\{a < b\}$ are *a*, *b*, *ab*, *aab*, *abb*, *aaab*, *aabb*, *abbb*, *aaaab*, *aaabb*, *aabab*, *aabbb*, *ababb* and *abbbb*. The shortest ones over $\{b < a\}$ are of course obtained replacing each *a* by a *b* and each *b* by an *a*. Lyndon words of length 6 over $\{b < a\}$ are *bbbbba*, *bbbbaa*, *bbbaaa*, *bbbaba*, *bbabaa*, *bbaaba*, *bbaaaa*, *babaaa*, *baaaaa*.

The *aim of the puzzle* is to fill each cell of a grid with a letter *a* or *b* so that each row read from left to right and each column read top-down yields a Lyndon word over $\{a < b\}$ or over $\{b < a\}$. For each initial grid, the set of predetermined cells is defined in such a way that one and only one correct solution exists. Let us give an example with its solution :

Puzzle

	a	b	
	b	a	

Solution

a	a	b	b
a	a	b	b
b	b	a	a
b	b	a	a

Here follow two other examples (solutions can be found on my home page) :

Puzzle 1

a				
	b	b		
	a	b		a
a		a		
			b	

Puzzle 2

							a	
	a	b			a	a	b	
	b		a					
b	a		b					
a				a				
a	a		b	a	a	b		
			a	a		b		a

3 Educational matters

As explained in the introduction, my aim when designing my first grid was educational. Let me narrate my experience with it. I tested it with some students (Puzzle 1 was made as an introductory exercise and Puzzle 2 was given as homework). Most of the students managed to find the solution, often empirically, sometimes bactracking. We then discussed on the way to obtain the solution the most directly as possible. Quickly it was observed by students that the first and last letters of a Lyndon word must be different. I then introduced the fact that "*a Lyndon word is* *unbordered*" (a word w is unbordered if the only word which is both prefix and suffix of w is w itself w) and its corollary "*a Lyndon word is primitive*" (that is, it is not a power of a strictly smaller word). We deduced basic rules to fill a (enough large) grid, as for instance:

$$\textit{Rule 1}: ?a \ldots b? \rightarrow aa \ldots bb$$
$$\textit{Rule 2}: ab \ldots ?? \rightarrow ab \ldots bb$$

Symbol ? in rules denotes an unknown letter. Rule 1 means that if we do not know the first and penultimate letter of a Lyndon word, if the second letter is a and if the last letter is a b, then the Lyndon word must start with aa and must end with bb. Rule 2 means that any Lyndon word starting with ab must end with bb.

One can observe that Puzzle 1 can be solved using only the unborderedness of Lyndon words. Hence Puzzle 2 should be a better introduction. It allows to remark that if a Lyndon word w starts with $a^n b$ for a non-zero integer n, then a^{n+1} is not a factor of w (a factor of a word is a subsequence of consecutive letters).

To end with educational matters, I would like to notice that before starting the exercise with students, I just defined Lyndon words and give few examples. While discussing the solution, I mentioned that another approach could have been to enumerate Lyndon words of length 5 (only 6 such words exist over $\{a < b\}$) and to try to put them in the grid. To explain it, I gave the lists of Lyndon words over $\{a < b\}$ for each length from 1 to 5. This lead a student to ask for the number of Lyndon words for each length (see [6] for an answer).

4 Some theoretical questions

All grids presented here were handmade. Hence a natural question is:
Problem 1: How to generate (effectively) a Sudo-Lyndon puzzle?

To answer this question, one would certainly need to know the structure of possible solutions of a Sudo-Lyndon puzzle. Without any information of this kind, a basic idea is to generate a candidate grid until it has a unique solution. But for this we need an answer to the following second natural question:
Problem 2: Given a grid partially filled with letters a and b, how to know (effectively) if there exists a unique solution?

Of course, for each of the previous questions, we vould like to know its complexity class. In particular is Problem 2 NP-complete as is the similar question for Sudoku game [10]?

We observe that it can be determined in linear time with respect to the number of cells whether a grid filled with letters a and b is a possible solution of a puzzle, that is, whether each row and column yields a Lyndon word. This is an immediate consequence of the existence of a linear time algorithm to check whether a word is a Lyndon word (see [8, chapter 1] for references).

A sub-question to Problem 2 concerns words or more precisely partial words as defined by J. Berstel and L. Boasson [1] and studied in depth by F. Blanchet-Sadri (see for instance [2]). A *partial word* is a word with *holes*, that is, with positions where letters are undetermined. Each row/column in a Sudo-Lyndon puzzle is a partial word. Hence:
Problem 3: given a partial word, can we replace a letter in such a way that the result yields a Lyndon word?
This third problem is being studied by Blanchet-Sadri and Davis [3].

The grid on the right shows that a positive answer to Problem 3 is not sufficient to solve Problem 2 since in the unique hole an occurrence of the letter b is needed to have a horizontal Lyndon word whereas an occurrence of the letter a is needed to have a vertical Lyndon word.

No solution!

a	a	b	b	b
a	a	b	a	b
a	a	b	b	b
a	b	a		b
b	b	a	a	a

To end this section, we consider the question, connected to Problems 1 and 2, of the enumeration of all the possible solutions of the puzzle. The scheme below shows that the number of such grids grows exponentially with the number of rows and columns. Indeed in the scheme each $\ast$ symbol can be replaced independently by an a or a b in order to obtain a possible solution of a puzzle. Hence $2^{(\lfloor \frac{n}{2} \rfloor - 1)(\lfloor \frac{m}{2} \rfloor - 1)}$ different solutions of the puzzle can be available with this scheme.

	1	2	$\dots$	$\lfloor \frac{m}{2} \rfloor$	$\lfloor \frac{m}{2} \rfloor + 1$	$\dots$	m-1	m
1	a	a	$\dots$	a	b	$\dots$	b	b
2	a	a	$\dots$	a	b	$\dots$	b	b
	$\vdots$	$\vdots$	$\ddots$	$\vdots$	$\vdots$	$\ddots$	$\vdots$	$\vdots$
$\lfloor \frac{n}{2} \rfloor$	a	a	$\dots$	a	b	$\ddots$	b	b
$\lfloor \frac{n}{2} \rfloor + 1$	b	b	$\dots$	b	$\ast$	$\ddots$	$\ast$	a
	$\vdots$	$\vdots$	$\ddots$	$\vdots$	$\vdots$	$\ddots$	$\vdots$	$\vdots$
n-1	b	b	$\dots$	b	$\ast$	$\dots$	$\ast$	a
n	b	b	$\dots$	b	a	$\dots$	a	a

5 Variants

When constructing the grids presented here, I was annoying by the feeling that two much cells were filled in the initial grid of the puzzle. This leads to the problems:

Problem 4: given a puzzle for which we know there exists a unique solution, can we determine if it is minimal in the sense that no letter in the grid can be deleted without losing the uniqueness of the solution?
Problem 5: given integers n, m, what is the minimal number $f(n,m)$ such that there exists at least one grid that starts with $f(n,m)$ initial values and has a unique solution?

In the particular case of a word, one can see that the ratio of this minimal amount over the length of a word tends to 0 when the length of the word tends to infinity. Indeed the partial word (each hole is indicated by a ? character)

$$ab^p?a?^{2p+2}[a?^{2p+3}]^p$$

is of length $2p^2+7p+5$ and contains only $2p+2$ known letters, and the Lyndon word $ab^{p+1}ab^{2p+2}[ab^{2p+3}]^p$ is the unique solution to this one dimension puzzle.

Finally, to deal with the problem of having too much information, I have thought about the following variants. For each variant, the aim is still to fill each cell of a grid with a letter a or b so that each row read from left to right and each column read top-down yields a Lyndon word over $\{a < b\}$ or over $\{b < a\}$.

Variant 1: For each row/column, the number of occurrences of the letter a is given.
Variant 2: As for variant 1, but the value of some cells are also given.
Variant 3:(As for the original Sudoku puzzle), the grid is divided into subgrids. For each subgrid, we consider the word obtained reading the rows of the subgrid from the top one and from left to right (One can prefer to concatenate columns and so naturally other variant is to consider simultaneously the two possibilities). In this variant the global grid should be filled in such a way a Lyndon word is written on each row, column and subgrid.
Variant 4: As for variant 3, but moreover some cells can be initially filled with the * character meaning that in the final solution the value of the cell can be equally the letter a and b. Of course, as for the initial puzzle, there is only one manner to fill a cell (except those marked with a *) for which the value is not known at the beginning.

Variant 1

	4	5	3	3	2	2
2						
2						
4						
2						
5						
4						

Variant 2

	3	4	2	2	2
2	a				
3					
2					
3					
3					

Variant 3

	a			b	
					b
	b	a	a	a	
		a			

Variant 4

		b	a			a		
	a		b	b		b	b	
	a	b			b			a
	b			b	*			
				*			*	
				b	*			
	b	*		a	*		a	

6 Conclusion

We have already mentioned that Puzzle 1 could be solved using only unborderdness. This kind of puzzle can of course be generalized to larger grid, but it can also open the question to find interesting puzzles based on other properties of words. For instance all Lyndon words in the grid below do not contain the words *aaa* and *bbb* as factors.

a	b					a		
	a			b				
					b			
							a	
							b	

I hope that you get fun playing Sudo-Lyndon.

Acknowledgements. Many thanks to my beta-testers students, to Richard Groult and especially to Patrice Séébold who encourages me to widen the communication of the present puzzle.

References

[1] J. Berstel and L. Boasson. Partial words and a theorem of Fine and Wilf. *Theoret. Comput. Sci.*, 218:135–141, 1999.

[2] F. Blanchet-Sadri. Primitive partial words. *Discret. Appl. Math.*, 148:195–213, 2005.

[3] F. Blanchet-Sadri and C.D. Davis. Lyndon partial words. Work in progress (see http://www.uncg.edu/mat/reu/summer2006/).

[4] M. Chemillier. Periodic musical sequences and Lyndon words. *Soft Computing, special issue on Formal Systems and Music, G. Assayag, V. Cafagna, M. Chemillier (eds)*, 8(9):611–616, 2004.

[5] Y. Dieudonné and F. Petit. Circle Formation of Weak Robots and Lyndon Words. *Information Processing Letters*, pages 156–162, 2007. doi:10.1016/j.ipl.2006.09.08.

[6] M. Lothaire. *Combinatorics on words*, volume 17 of *Encyclopedia of Mathematics*. Addison-Wesley, 1983. Reprinted in 1997 by Cambridge University Press in the Cambridge Mathematical Library, Cambridge, UK, 1997.

[7] M. Lothaire. *Algebraic Combinatorics on words*, volume 90 of *Encyclopedia of Mathematics*. Cambridge University Press, Cambridge, UK, 2002.

[8] M. Lothaire. *Applied Combinatorics on Words*, volume 105 of *Encyclopedia of Mathematics*. Cambridge University Press, Cambridge, UK, 2005.

[9] B. Smyth. *Computing Patterns in Strings*. Addison Wesley, 2003.

[10] T. Yato and T. Seta. Complexity and Completeness of Finding Another Solution and Its Application to Puzzles. In *Proceedings of the National Meeting of the Information Processing Society of Japan*, 2002. (also T. Yato's Master's thesis, The University of Tokyo, 2003).