Suffix Array for Large Alphabet[1]

Radovan Šesták Jan Lánský Michal Žemlička

Charles University, Faculty of Mathematics and Physics, Prague, Czech Republic
{radofan, zizelevak}@gmail.com, michal.zemlicka@mff.cuni.cz

Burrows-Wheeler Transform (BWT) is used as the main part in block compression which has a good balance of speed and compression ratio. Suffix arrays are used in the coding phase of BWT and we focus on creating them for an alphabet larger than 256 symbols. The motivation for this work has been software project XBW [3] – an application for compression of large XML files using word- and syllable-based BWT. The role of BWT is to reorder input before applying other algorithms. We describe and implement three families of algorithms for encoding. The first is inspired by the work of Sadakane and further improved by Larsson [4]. The second family includes algorithm by Seward [5] and algorithm by Itoh further improved by Kao [1]. Finally we present algorithm by Kärkkäinen and Sanders [2] for constructing suffix arrays in linear time.

We use files of different sizes from several corpora for testing. Block size for BWT was for testing set to 100 MB; every file has been therefore compressed in a single block. According compression ratio the most successful was the compression based on words and syllables that give very similar results. Significantly less successful was the compression using alphabet of bytes, least successful was the use of 2-byte alphabet.

The most successful in speed comparison of algorithms using single-character alphabet was the Seward's algorithm. As this algorithm uses two-level bucket sort, it is not reasonable to implement it for larger alphabets – syllables and words inclusive. We have therefore modified it to use only one-level bucket sort. So modified algorithm was able to reach the second place in compression using words or syllables as alphabet. For word- or syllable-based compression the most successful was the Kao's modification of Itoh's algorithm.

We have moreover compared time necessary for BWT and the entire compression (BWT, MTF, RLE, and arithmetic coding) for the alphabets of words, syllables and characters. The use of syllables results in about one half slower time and the use of characters gives 2.5× higher times than the use of words.

For the BWT-based compression of textual files when unlimited memory is available we can recommend to use alphabet of words and Kao's modication of Itoh's algorithm for the sorting in the BWT phase of the compression.

References

[1] Kao, T. H.: Improving suffix-array construction algorithms with applications. Master Thesis. Gunma University, Kiryu, Japan, (2001).

[2] Kärkkäinen, J., Sanders, P.: Simple linear work suffix array construction. In: Proceedings of 13th International Conference on Automata, Languages and Programming. Springer, (2003) 943–955.

[3] Lánský, J., Šesták, R., Uzel, P. et al.: XBW – word-based compression of non-valid XML documents, (2007). http://xbw.sourceforge.net/

[4] Larsson, N. J.: Notes on suffix sorting. Technical report LU-CS-TR:98-204, Lund University, Sweden, (1998).

[5] Seward, J.: On the performance of bwt sorting algorithms. In: Proceedings of the IEEE Data Compression Conference, (2000), page 173.

[1] This research was partially supported by Charles University Grant Agentur in the project "Text compression" (GAUK no. 1607, section A) and by the Program "Information Society" under projects 1ET100300419 and 1ET100300517.

1068-0314/08 $25.00 © 2008 IEEE
DOI 10.1109/DCC.2008.22