Journal of Automata, Languages and Combinatorics 8 (2003) 4, 593–606

COMPUTING QUASI SUFFIX ARRAYS[1,2]

FRANTIŠEK FRANĚK
Algorithms Research Group, Department of Computing and Software
McMaster University, Hamilton, Ontario, Canada
e-mail: franek@mcmaster.ca

JAN HOLUB
Department of Computer Science and Engineering, Czech Technical University
Prague, Czech Republic
e-mail: holub@fel.cvut.cz

WILLIAM F. SMYTH[3] and XIANGDONG XIAO
Algorithms Research Group, Department of Computing and Software
McMaster University, Hamilton, Ontario, Canada
e-mail: smyth@mcmaster.ca

ABSTRACT

We introduce quasi suffix arrays as a generalization of suffix arrays for character strings. We show that a quasi suffix array encodes enough of the structure of the string to be a useful construct for many applications where the full power of suffix arrays is not necessary, notably in problems that do not require lexicographical order, for example, pattern-matching or calculation of repeating substrings. We are interested in quasi suffix arrays, for we believe that they can be calculated by simple, fast, and space efficient algorithms. As a first step towards this goal, we describe a family DIST of algorithms (inspired by the Crochemore's repetitions algorithm) that compute the quasi suffix array in the average-case in $O(|x| \log |x|)$ time, where x is the input string. Based on experiments conducted by one of us (Xiao), it appears that in practice our algorithms execute faster than all suffix tree and most suffix array construction algorithms. Though at this time we can only prove that the average-case complexity is $O(|x| \log |x|)$, tests carried out by one of us (Holub) strongly suggest that not only the worst-case complexity may be the same as the average-case complexity, but both may in fact be linear. Given the very recent results on computing suffix arrays in linear time by recursive algorithms, the only advantage quasi suffix arrays can have lies in the simplicity and space efficiency of DIST algorithms that do not use recursion.

Keywords: Quasi suffix arrays, pattern matching, suffix trees, string algorithms

[1]Full version of a lecture presented at the *Thirteenth Australasian Workshop on Combinatorial Algorithms* (Kingfisher Bay Resort, Fraser Island, Queensland, Australia, July 7 – 10, 2002).

[2]Supported in part by grants from the NSERC (Franek, Smyth), and NSERC/NATO Science Fellowship grant No. GP201/01/P082 and GAČR grant No. GA201/01/1433 (Holub).

[3]School of Computing, Curtin University, Perth, WA, Australia.