Journal of Automata, Languages and Combinatorics **10** (2005) 5/6, 627–640

CONSTRUCTING FACTOR ORACLES [1]

LOEK CLEOPHAS, GERARD ZWAAN
Department of Mathematics and Computer Science, Technische Universiteit Eindhoven, P.O. Box 513, NL-5600 MB Eindhoven, The Netherlands
e-mail: `loek@loekcleophas.com`, `g.zwaan@tue.nl`

and

BRUCE W. WATSON
Department of Computer Science, University of Pretoria, Pretoria 0002, South Africa
e-mail: `bruce@bruce-watson.com`

ABSTRACT

A *factor oracle* is a data structure for weak factor recognition. It is an automaton built on a string p of length m that is acyclic, recognizes at least all factors of p, has $m+1$ states which are all final, and has m to $2m-1$ transitions. In this paper, we give two alternative algorithms for its construction and prove the constructed automata to be equivalent to the automata constructed by the algorithms in a paper by Allauzen et al. Although these new algorithms are practically inefficient compared to the $\mathcal{O}(m)$ algorithm given there, they give more insight into factor oracles. Our first algorithm constructs a factor oracle based on the suffixes of p in a way that is more intuitive. Some of the crucial properties of factor oracles, which in the paper by Allauzen et al. need several lemmas to be proven, are immediately obvious. Another important property however becomes less obvious. A second algorithm gives a clear insight in the relationship between the trie or DAWG (*directed acyclic word graph*) recognizing the factors of p and the factor oracle recognizing a superset thereof.

Keywords: Factor oracle, finite automaton, weak factor recognition, algorithm derivation, pattern matching

1. Introduction

A *factor oracle* is a data structure for weak factor recognition. It can be described as an automaton built on a string p of length m that (a) is acyclic, (b) recognizes at least all factors of p, (c) has $m+1$ states (which are all final), and (d) has m to $2m-1$ transitions (cf. [1]). Some example factor oracles are given in Figures 1 and 2.

Factor oracles are introduced in [1] as an alternative to the use of exact factor recognition in many on-line keyword pattern matching algorithms. In such algorithms, a window on a text is read backward while attempting to match a keyword factor.

---

[1] Full version of a submission presented at the *Prague Stringology Conference* (Czech Technical University in Prague, Czech Republic, September 22–24, 2003).