Experimental results for Multiple serial episodes matching****

Patrick Cégielski * — Irène Guessarian ** — Yuri Matiyasevich ***

* *LACL, UMR-FRE 2673, Université Paris 12, Route forestière Hurtault, F-77300 Fontainebleau, France, cegielski@univ-paris12.fr*

** *LIAFA, UMR 7089 and Université Paris 6, 2 Place Jussieu, 75254 Paris Cedex 5, France; send correspondence to ig@liafa.jussieu..fr*

*** *Steklov Institute of Mathematics, Fontanka 27, St. Petersburg, Russia. yumat@pdmi.ras.ru*

**** *Support by INTAS grant 04-77-7173 is gratefully acknowledged.*

ABSTRACT. *In [BCGM01] we have generalized the Knuth-Morris-Pratt (KMP) pattern matching algorithm and defined a non-conventional kind of RAM, the MP–RAMs which model more closely the microprocessor operations, and designed an $O(n)$ on-line algorithm for solving the serial episode matching problem on MP–RAMs when there is only one single episode. In [CGM05] we gave two extensions of this algorithm to the case when we search for several patterns simultaneously and compare them. More preciseley, given $q+1$ strings (a text t of length n and q patterns $m_1, \ldots, m_q$) and a natural number w, the* multiple serial episode matching problem *consists in finding the number of size w windows of text t which contain patterns $m_1, \ldots, m_q$ as subsequences, i.e. for each m_i, if $m_i = p_1, \ldots, p_k$, the letters $p_1, \ldots, p_k$ occur in the window, in the same order as in m_i, but not necessarily consecutively (they may be interleaved with other letters). We here give the results of our experiments with the algorithms of [CGM05] and sets of synthetic data.*

KEYWORDS: *Subsequence matching, algorithm, frequent patterns, episode matching, datamining.*

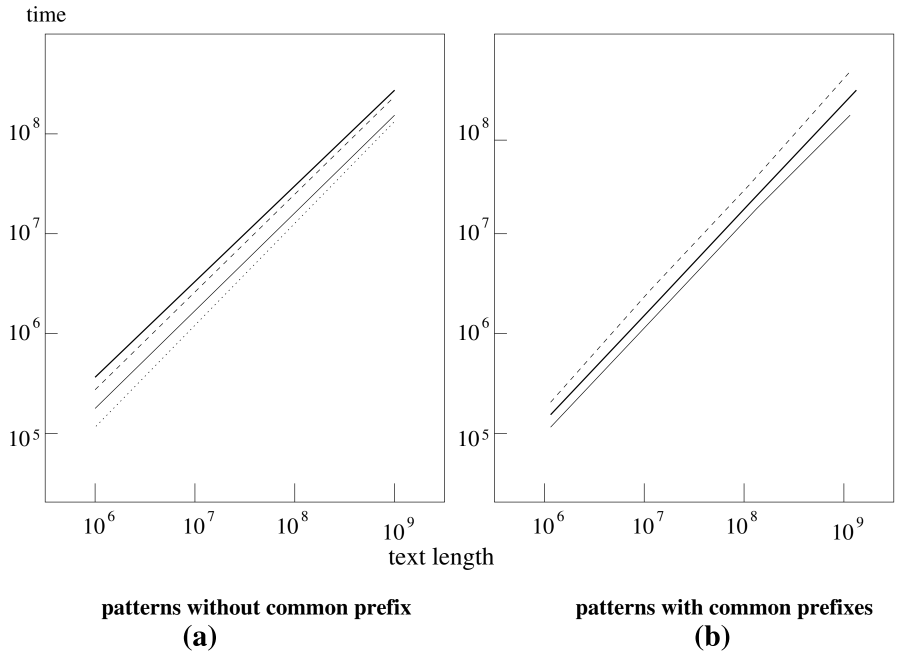

Figure 1. *The continuous thin lines represent the execution time of the MP–RAM algorithm (with trie); the dotted line represents the execution time of the MP–RAM algorithm (with concatenation); the dashed lines the execution time of the standard algorithm (with concatenation) and the continuous thick lines the execution time of the standard algorithm (with trie).*

We present here the results of our experiments with the algorithms presented in the paper Multiple serial episodes matching.

The algorithm on MP–RAM has a better complexity than the standard algorithm, however, the underlying computation models being different, we checked experimentally that the MP–RAM algorithm is faster. We implemented all algorithms in C++. Experiments were realised on a PC (256 Mo, 1Ghz) with Linux. The text was a randomly generated file. We measured the time with machine clock ticks.

For searching multiple patterns, we took 3 to 5 patterns of length 2 to 4; in figure 1, case (a) is the case of patterns having no common prefix, and case (b) is the case of patterns having common prefixes. In case (a), the MP–RAM algorithm where we concatenate the patterns is at least twice as fast as the standard "naive" algorithm where patterns are concatenated; both standard algorithms (with patterns concatenated or organised in a trie) are equivalent, the algorithm with concatenation being slightly faster; this was predictable since a trie organisation will not give a significant advantage in that case; the MP–RAM algorithm where the patterns are organised in a trie is 30 to 50% faster than the standard algorithm with trie, and 10 to 15% slower than the MP–RAM algorithm where the patterns are concatenated. However, as soon as the total length of the patterns is larger than 7 or 8, or the window size is larger than 30, if patterns are concatenated, the automaton state can no longer be encoded in a single 32 bits memory cell, and it is better to use the MP–RAM algorithm with trie (figure 1 case (b)). Figure 1 case (b) shows that, for patterns having common prefixes, the MP–RAM algorithm with trie is 1.3 to 1.5 times faster than the standard algorithm with trie, itself 1.4 to 1.6 times faster than the standard algorithm with concatenation.

1. References

[BCGM01] L. Boasson, P. Cegielski, I. Guessarian, Y. Matiyasevich, Window Accumulated Subsequence Matching is linear, Annals of Pure and Applied Logic Vol. 113 (2001), pp. 59-80.

[CGM05] P. Cegielski, I. Guessarian, Y. Matiyasevich, Multiple serial episode matching, submitted.