

THE OUTPUT SIZE PROBLEM FOR STRING-TO-TREE TRANSDUCERS

MARTIN BERGLUND^(A) FRANK DREWES^(B) BRINK VAN DER MERWE^(C)

^(A)*Department of Information Science and Center for AI Research (CSIR)
Stellenbosch University, 7600 Stellenbosch, South Africa
pberglund@sun.ac.za*

^(B)*Department of Computing Science, Umeå University
901 87, Umeå, Sweden
drewes@cs.umu.se*

^(C)*Department of Computer Science, Stellenbosch University
7600 Stellenbosch, South Africa
abvdm@cs.sun.ac.za*

ABSTRACT

The output size problem, for a string-to-tree transducer, is to determine the asymptotic behavior of the function describing the maximum size of output trees, with respect to the length of input strings. We show that the problem to determine, for a given regular expression, the worst-case matching time of a backtracking regular expression matcher, can be reduced to the output size problem. The latter can, in turn, be solved by determining the degree of ambiguity of a non-deterministic finite automaton.

Keywords: string-to-tree transducers, output size, backtracking regular expression matchers, NFA ambiguity

1. Introduction

The complexity of determining the asymptotic behavior of the maximum output size for trees produced by a given top-down tree transducer, as a function of the size of input trees, was initially studied in [5]. It was shown that the exponential output size problem is NL-complete for total top-down tree transducers, and DEXPTIME-complete for top-down tree transducers in general. Naturally, this problem asks whether the size of the output trees grows exponentially in the size of the input trees. We investigate the output size problem for string-to-tree transducers, and consider in particular the complexity of determining the degree of the polynomial, in cases where the maximum output size is polynomial in the size of input strings.

The motivation for this research is provided by the observation that the problem of determining the worst-case matching time of a backtracking regular expression matcher [2, 3] in terms of the length of the input string, can be reduced to an output