Parallel Generation of Postfix and Tree Forms

ELIEZER DEKEL and SARTAJ SAHNI
University of Minnesota Twin Cities

Efficient parallel algorithms to obtain the postfix and tree forms of an infix arithmetic expression are developed. The shared memory model of parallel computing is used.

Categories and Subject Descriptors: D.3.4 **[Programming Languages]**: Processors—*parsing*

General Terms: Algorithms

Additional Key Words and Phrases: Arithmetic expressions, postfix, infix, tree form, parallel computing, complexity

1. INTRODUCTION

The parallel parsing and evaluation of arithmetic expressions has been the focus of research for many years. References [1, 2, 10, 11, 13] are some of the important papers written on the parallel evaluation of arithmetic expressions. The most significant result here is due to Brent [1]. He has shown [1] that arithmetic expressions containing n operands, $n \geq 1$; operators (+, *, and /); and parentheses can be evaluated in $4 \log_2 n + 10(n - 1)/p$ time when p processors are available. Parallel parsing of arithmetic expressions has been considered by Fischer [5], Krohn [8], Lipkie [12], and Schell [16] (among others). Fischer's work is restricted to vector (or pipelined) computers. While Krohn's work was intended primarily for pipelined computers (specifically for the CDC STAR-100), the ideas contained in [8] can be extended to parallel multiprocessor computers. Krohn, however, does not consider the asymptotic performance that could be obtained from his parallel parsing algorithm. Lipkie [12] and Schell [16] explicitly consider parsing on parallel multiprocessor computers. Lipkie [12] provides some grammar rules for parallel parsing but does not develop a formal algorithm. Schell [16] is a thorough study of parallel techniques for several of the phases normally encountered in compiling (scanning, syntax analysis, parsing, error recovery, etc.). Schell develops a parallel LR parser. The complexity of this parser is, however, quadratic

This research was supported in part by the Office of Naval Research under contract N00014-80-C-0650.

Authors' present addresses: E. Dekel, Mathematical Sciences Program, University of Texas at Dallas, Richardson, TX 75080; S. Sahni, Department of Computer Science, University of Minnesota Twin Cities, 136 Lind Hall, 207 Church Street S.E., Minneapolis, MN 55455.

Permission to copy without fee all or part of this material is granted provided that the copies are not made or distributed for direct commercial advantage, the ACM copyright notice and the title of the publication and its date appear, and notice is given that copying is by permission of the Association for Computing Machinery. To copy otherwise, or to republish, requires a fee and/or specific permission.
© 1983 ACM 0164-0925/83/0700-0300 $00.75