© INTERNATIONAL JOURNAL FOR RESEARCH PUBLICATION & SEMINAR
ISSN: 2278-6848 | Volume: 07 Issue: 05 | July - September 2016
Paper is available at www.jrps.in | Email : info@jrps.in

An Efficient Technique for Conversion of Regular Expression to and from Finite Automata

Neha Sharma, Department of Computer Science, *Neha_sh0815@yahoo.com*
South Point Institute of Technology and Management,Sonepat

Abstract— Regular expressions are used to represent certain set of string in algebraic manner. Regular expressions are widely used in the field of compiler design, text editor, search for an email- address, grep filter of unix, train track switches, pattern matching ,context switching and in many areas of computer science. The demand of converting regular expression into finite automata and vice versa motivates research into some alternative so that time taken for above is minimized. For conversion of deterministic finite automata to regular expression, several techniques like Transitive closure method, Brzozowski Algebraic method and state elimination method have been proposed. In this paper, for Conversion of regular expression to NFA we study the Thomson Algorithm; to convert NFA to DFA we use Subset Construction method, to minimized DFA constructed from previous step we use partition method and finally to convert DFA back to RE we use Universal Technique.

***Keywords*— Regular Expression, *DFA, NFA,* ε-closure**

I. INTRODUCTION

In formal language theory regular expressions consist of strings of symbols from a finite alphabet Σ combined by various operators. In computing in general they can be used to match and replace strings, but formally they define regular languages. Regular languages can be roughly defined, somewhat recursively, as any language consisting of a potentially infinite set of sequences of finite symbols from a finite alphabet that can be described by a regular expression or deterministic or nondeterministic finite automaton.

The demand of converting regular expression into finite automata and vice versa motivates research into some alternative so that time taken for above is minimized. For conversion of deterministic finite automata to regular expression, several techniques like Transitive closure method, Brzozowski Algebraic method and state elimination method have been proposed. None of the above specified technique is able to find smallest regular expression. Our purpose is to find the smallest regular expression equivalent to given deterministic finite automata. State elimination approach is the most widely used and efficient approach for converting deterministic finite automata to regular expression.

The presented paper investigates and compares different techniques used for converting deterministic finite automata to regular expression. Brief comparisons amongst different techniques are presented and several heuristics are explored for obtaining smaller regular expression using state elimination approach. Here we define and implement algorithms to convert regular expressions to NFA, to convert these NFA to DFA, minimization of these DFA and finally conversion of these minimized DFA back into a regular expressions. The algorithms addressed include Thompson's Algorithm and the Rabin and Scott's Subset Construction. To minimized DFA constructed from previous step we use partition method and finally to convert DFA back to RE we use Universal Technique.

II. LITERATURE REVIEW

This section describes different techniques used for converting deterministic finite automata to regular expression and vice versa.

[A] Conversion of DFA to RE

Kleene proves that every RE has equivalent DFA and vice versa. On the basis of this theoretical result, it is clear that DFA can be converted into RE and vice versa using some algorithms or techniques. For converting RE to DFA, first we convert RE to NFA(Thomson Construction) and then NFA is converted into DFA(Subset construction).For conversion of DFA to regular expression, following methods have been introduced.[2, 12, 10]