

Some Properties of Brzozowski Derivatives of Regular Expressions

N.Murugesan^{#1}, O.V.Shanmuga Sundaram^{*2}

^{#1}Assistant Professor, Dept of Mathematics, Government Arts College (Autonomous),
Coimbatore – 641018, Tamil Nadu, India,

^{*2}Assistant Professor, Dept of Mathematics, Sri Shakthi Institute of Engineering & Technology,
Coimbatore – 641062, Tamil Nadu, India.

Abstract :- Brzozowski's derivatives of a regular expression are developed for constructing deterministic automata from the given regular expression in the algebraic way. In this paper, some lemmas of the regular expressions are discussed and the regular languages of the derivatives are illustrated. Also the generalizations of the Brzozowski's derivatives are proved as theorems with help of properties and known results.

AMS MSC2010 Certification: 68Q45, 68Q70

Keywords— Regular expressions, derivatives, and Kleene Closure.

I. INTRODUCTION

Regular expressions are declarative way of defining regular languages recognized by a DFA or a NFA. They are equivalent to one another in the sense that, for a given regular expression, it can be constructed a finite state automata recognizing the same language described by the regular expression, and vice-versa.

All over the years, various attempts have been made to accomplish this task. In the year 1960, R.McNaughton and H.Yamada [6] provided an algorithm to construct a non – deterministic finite automaton from a regular expression.

G.Berry and R.Sethi [1] discussed the theoretical background for the R.McNaughton and H.Yamada algorithm. V.M.Glushkov [4] has also given a similar algorithm in the year 1961. An elegant construction of deterministic finite automata based on the derivatives of regular expressions was proposed by J.A Brzozowski [2] in the year 1964. J.E.Hopcroft and J.D.Ullman [5] discussed the construction of ε - NFA from the given regular expression. J.M.Champarnaud and others [3] described a variant of the step by step construction which associates standard and trim automata to regular languages.

In this paper, we discuss some basic set theoretic properties involved in Brzozowski way of constructions of automata have been discussed.

II. REGULAR EXPRESSIONS

Let Σ be an alphabet of symbols. A word over an alphabet Σ is a finite sequence of symbols from that

alphabet Σ . The set of all words over Σ is denoted by Σ^* . The empty word is denoted by ε . A regular expression is defined inductively as

- (i) ε is a regular expression.
- (ii) For any $a \in \Sigma$, the symbol 'a' is a regular expression.
- (iii) If E and F are regular expressions, then $E + F, EF, (E)^*$ are all regular expressions.

The regular expressions $E + F, EF, (E)^*$ are called respectively union, concatenation, Kleene closure of the corresponding regular expressions. The language of a regular expression E is denoted as $L(E)$, and defined the same for various regular expressions as follows.

- (i) $L(\varepsilon) = \{\varepsilon\}$
- (ii) $L(a) = \{a\}$
- (iii) $L(E + F) = L(E) \cup L(F)$
- (iv) $L(EF) = L(E)L(F)$
- (v) $L(E^*) = (L(E))^*$

The empty set \emptyset is also considered as a language of regular expression denoted by the symbol \emptyset itself. It is assumed that

$$E + \emptyset = \emptyset + E = E; \quad E\emptyset = \emptyset E = \emptyset; \quad \varepsilon E = E\varepsilon = E$$

The properties of the regular languages are discussed in [9].

The following lemma gives some algebraic type identities with respect to regular expressions.

2.1 Lemma

Let E and F are any two regular expressions. Then,

- (i) $E + F = F + E$
- (ii) $EF = FE$ only when
 - (a) $E = F$ or (b) one of E, F is ε or \emptyset .
- (iii) $(E + F) + G = E + (F + G)$
- (iv) $(E^*)^* = E^*$
- (v) $E^*(F + G) = E^*F + E^*G$

Not all algebraic type identities are hold in the case of regular expressions.

2.2 Lemma:

For any two regular expressions E and F, then,

- (i) $(E + F)^* \neq E^* + F^*$
- (ii) $(EF)^* \neq E^*F^*$ (iii) $EF \neq FE$

Provided that E and F are not equal to ϵ or \emptyset .

2.3 Lemma:

- (i) $(\epsilon + a)^* = a^*$ (vi) $\epsilon + aa^* = a^*$
- (ii) $a^*(\epsilon + a) = a^*$ (vii) $(a+b)^* = (a^*b^*)^*$
- (iii) $(\epsilon + a) + a^* = a^*$ (viii) $\emptyset a = a\emptyset = \emptyset$
- (iv) $b + a^*b = a^*b$ (ix) $\emptyset + a = a + \emptyset = a$
- (v) $b + ba^* = ba^*$ (x) $\epsilon + a^* = a^*$
- (xi) $a(b+c) = ab+ac$ (xiv) $\epsilon a = a\epsilon = a$
- (xii) $(a+b)c = ac+bc$ (xv) $\epsilon^* = \epsilon$
- (xiii) $(a+b)^* = (a^* + b^*)^* = (a^*b^*)^*$

Some of the proofs of the equivalent regular expressions given in the above lemmas are proved in [7].

III Derivatives of Regular Expressions

3.1 Definition

Given a language L and a symbol 'a', the derivative of L with respect to a symbol a is defined as

$$D_a(L) = \{b | ab \in L\}.$$

The derivatives of regular expressions with respect to a symbol are defined as follows:

- 1 $D_a(\emptyset) = D_a(\epsilon) = \emptyset$ $D_\epsilon(a) = a$
- 2 $D_a(b) = \begin{cases} \epsilon & \text{if } b = a \\ \emptyset & \text{Otherwise} \end{cases}$
- 3 $D_a(E + F) = D_a(E) + D_a(F)$
- 4 $D_a(EF) = \begin{cases} D_a(E)F + D_a(F) & \text{if } \epsilon \in L(E) \\ D_a(E)F & \text{otherwise} \end{cases}$
- 5 $D_a(E^*) = D_a(E)E^*$
- 6 $D_\epsilon(E) = E$ 7 $D_{w_a}(E) = D_a(D_w(E))$

The operator D is treated as a prefix operator with high precedence than "+", ".", and "*". The derivatives involving the operators intersection, and complement are defined by

$$D_a(E \cap F) = D_aE \cap D_aF$$

and $D_a(E - F) = D_aE - D_aF$.

It can be verified that

$$D_a(ab^* \cap a) = D_a(ab^*) \cap D_a(a)$$

$$D_a(ab^* - a) = D_a(ab^*) - D_a(a)$$

3.2 Examples

1 Let $E = a(a+b)^*$

Then $D_a(E) = (a+b)^*$

$$D_b(E) = \emptyset$$

2 Let $E = ab(a+b)^*$

Then $D_a(E) = b(a+b)^*$

$$D_b(E) = \emptyset$$

3 Let $E = (a+b)^*a$

$$\begin{aligned} \text{Then } D_a(E) &= [D_a((a+b)^*)]a + D_a(a) \\ &= [[D_a((a+b))](a+b)^*]a + \epsilon \\ &= [D_a(a) + D_a(b)](a+b)^*a + \epsilon \\ &= (\epsilon + \emptyset)(a+b)^*a + \epsilon \\ &= \epsilon(a+b)^*a + \epsilon \\ &= (a+b)^*a + \epsilon \end{aligned}$$

$$\begin{aligned} D_b(E) &= [D_b((a+b)^*)]a + D_b(a) \\ &= [[D_b((a+b))](a+b)^*]a + \emptyset \\ &= [D_b(a) + D_b(b)](a+b)^*a \\ &= (\emptyset + \epsilon)(a+b)^*a \\ &= \epsilon(a+b)^*a \\ &= (a+b)^*a \end{aligned}$$

3.3 Definition

Let L be a regular language. We define

$$\delta(E) = \begin{cases} \epsilon & \text{if } \epsilon \in L \\ \emptyset & \text{if } \epsilon \notin L \end{cases}$$

It can be easily seen that

- (i) $\delta(a) = \emptyset$, for any $a \in \Sigma$
- (ii) $\delta(\epsilon) = \epsilon$, and $\delta(\emptyset) = \emptyset$
- (iii) $\delta(E + F) = \delta(E) + \delta(F)$
- (iv) $\delta(E^*) = \epsilon$

3.4 Definition

Let $w = a_1a_2 \dots a_n$ and E be a regular expression. Then,

$$D_{a_1a_2}(E) = D_{a_2}(D_{a_1}(E))$$

$$D_{a_1a_2a_3}(E) = D_{a_3}(D_{a_1a_2}(E))$$

In general, we have

$$D_w(E) = D_{a_1 a_2 a_3 \dots a_n}(E) \\ = D_{a_n}(D_{a_1 a_2 \dots a_{n-1}}(E))$$

3.5 Theorem

Let E, F are two regular expressions and the word $w = a_1 a_2 \dots a_n$ a string over the Kleene closure of an alphabet Σ . Then,

$$D_w(E + F) = D_w(E) + D_w(F)$$

3.6 Theorem

Let E, F are two regular expressions and the word $w = a_1 a_2 \dots a_n$. Then,

$$D_{a_1}(EF) = (D_{a_1}(E))F + \delta(E)D_{a_1}(F) \\ D_{a_1 a_2}(EF) = (D_{a_1 a_2}(E))F + \delta(E)D_{a_1}(E)D_{a_1}(F) \\ + \delta(E)D_{a_1 a_2}(F)$$

In general,

$$D_{a_1 a_2 \dots a_n}(EF) = (D_{a_1 a_2 \dots a_n}(E))F \\ + \delta(D_{a_1 a_2 \dots a_{n-1}}(E))D_{a_n}(F) \\ + \delta(D_{a_1 a_2 \dots a_{n-2}}(E))D_{a_{n-1} a_n}(F) \\ + \delta(D_{a_1 a_2 \dots a_{n-3}}(E))D_{a_{n-2} a_{n-1} a_n}(F) \\ + \dots + \delta(D_{a_1}(E))D_{a_2 a_3 \dots a_n}(F) \\ + \delta(P)D_{a_2 a_3 \dots a_n}(F)$$

3.7 Theorem

Let the word $w = a_1 a_2 \dots a_n$. Then,

$$D_{a_1}(E^*) = (D_{a_1}(E))E^* \\ D_{a_1 a_2}(E^*) = (D_{a_1 a_2}(E))E^* \\ + \delta(D_{a_1}(E))D_{a_2}(E^*)$$

$$D_{a_1 a_2 a_3}(E^*) = (D_{a_1 a_2 a_3}(E))E^* \\ + \delta(D_{a_1}(E))D_{a_2 a_3}(E^*) \\ = (D_{a_1 a_2 a_3}(E))E^* + \delta(D_{a_1}(E)) \\ \times [D_{a_2 a_3}(E)E^* + \delta(D_{a_2}(E))D_{a_3}(E^*)] \\ = (D_{a_1 a_2 a_3}(E))E^* + \delta(D_{a_1}(E))D_{a_2 a_3}(E)E^* \\ + \delta(D_{a_1}(E))\delta(D_{a_2}(E))D_{a_3}(E^*)$$

Similarly, we can generalize

$$D_{a_1 a_2 a_3 \dots a_n}(E^*) = (D_{a_1 a_2 a_3 \dots a_n}(E))E^* \\ + \delta(D_{a_1}(E))D_{a_2 a_3 \dots a_n}(E)E^* \\ + \dots + \delta(D_{a_1}(E))\delta(D_{a_2}(E))\dots\delta(D_{a_{n-1}}(E))D_{a_n}(E^*)$$

3.8 Lemma

$$D_a[(aw)^*] = \varepsilon + w(aw)^*, \text{ where } a \in \Sigma, w \in \Sigma^*$$

Proof:

$$L[(aw)^*] = \{\varepsilon, aw, awaw, awawaw, \dots\} \\ D_a[L[(aw)^*]] = \{\varepsilon, w, waw, wawaw, \dots\} \\ = \{\varepsilon\} \cup \{w\} \{\varepsilon, aw, awaw, \dots\} \\ = \{\varepsilon\} \cup w(aw)^*$$

Hence the corresponding regular expression is $\varepsilon + w(aw)^*$.

3.9 Lemma

Let E be regular expression, then

$$D_a[L(aE)^*] = L(E)L(aE)^*, \text{ where } a \in \Sigma. \\ L(aE)^* = \{\varepsilon, aE, aEaE, \dots\} \\ D_a[L(aE)^*] = \{E, EaE, \dots\} \\ = \{E\} \{\varepsilon, aE, aEaE, \dots\} \\ = L(E)L(aE)^*$$

3.10 Theorem

Let E be any regular expression and a be any symbol over the alphabet Σ .

$$L(D_a(E)) = D_a(L(E)).$$

Proof:

Case (i): Let $E = \varepsilon$, then $L(E) = \{\varepsilon\}$, and $D_a(L(E)) = \emptyset$.

Also $D_a(\varepsilon) = \emptyset$, and $L(D_a(\varepsilon)) = \emptyset$.

On the other hand, if $E = \emptyset$, then

$$L(D_a(E)) = D_a(L(E)) = \emptyset.$$

Hence the theorem is true when $E = \varepsilon$ and $E = \emptyset$.

Case (ii): Let $E = a$.

Then, $D_a(a) = \varepsilon$. Hence $L(D_a(E)) = \{\varepsilon\}$.

Also $L(E) = \{a\}$, and $D_a(L(E)) = \{a\}$.

If $E = b \neq a$, then $L(D_a(E)) = D_a(L(E)) = \emptyset$.

Case (iii): Let $E = F + G$. We prove first $D_a(E) = D_a(F) + D_a(G)$.

Suppose, if $F = aF'$ and $G = aG'$, then $D_a(E) = F' + G'$ and

$$L(D_a(E)) = L(F') \cup L(G')$$

Let F and G are two regular expressions begin with a symbol other than 'a'. Then

$$D_a(F) = \emptyset = D_a(G)$$

Hence $D_a(E) = \emptyset$ and $L(D_a(E)) = \emptyset$.

In the first case, $L(F) = \{aw \mid w \in \Sigma^*\}$,

and $L(G) = \{au \mid u \in \Sigma^*\}$.

Hence $L(E) = \{aw, au \mid w, u \in \Sigma^*\}$.

Hence $L(E) = aL(F') \cup aL(G')$.

i.e., $D_a(L(E)) = L(F') \cup L(G')$.

In the second case, $L(F) = \{bw \mid b \neq a, w \in \Sigma^*\}$

and $L(G) = \{bu \mid b \neq a, u \in \Sigma^*\}$.

$$L(E) = \{bw, bu \mid b \neq a, w, u \in \Sigma^*\}.$$

Therefore, $D_a(L(E)) = \emptyset \cup \emptyset = \emptyset$.

Hence $L(D_a(E)) = D_a(L(E))$,

When $E = F + G$.

Case (iv):

Let $E = FG$. Suppose if F and G are two regular expressions begin with a symbol 'a', then it can be found as in the case (iii), that

$$L(D_a(E)) = D_a(L(E)) = L(F')L(G')$$

Similarly, if F and G are regular expressions begin with other than 'a', it can be found as

$$L(D_a(E)) = D_a(L(E)) = \emptyset.$$

Case (v):

Let $E = F^*$, then $D_a(E) = D_a(F^*) = D_a(F)F^*$.

Again there are two possibilities, say

$F = aF'$ or $F = bF'$ when $a \neq b$.

In the first case, $D_a(F) = F'$ and $D_a(F) = F'F^*$.

In the second case, $D_a(F) = \emptyset$. Therefore $D_a(E) = \emptyset$.

Hence the statement $L(D_a(E)) = D_a(L(E))$ is trivially true.

If $D_a(F) = F'F^*$, then

$$\begin{aligned} L(D_a(E)) &= L(F')L(F^*) \\ &= L(F')(L(F))^* \end{aligned}$$

On the other hand, if $E = F^*$ and $F = aF' = aw$. Then

$$L(E) = L(F^*) = L((aF')^*)$$

Hence

$$\begin{aligned} D_a(L(E)) &= D_a(L(aF')^*) \\ &= L(F')(L(F))^* \end{aligned}$$

Hence $L(D_a(E)) = D_a(L(E))$.

This proves the theorem.

3.11 Theorem

Let $w = a_1a_2 \dots a_n$, and E be a regular expression over an

alphabet Σ . Then $D_w(E) = D_{a_n}(D_{a_1a_2a_3 \dots a_{n-1}}(E))$.

Suppose if $w = \varepsilon$ then $D_w E = E$,

As an illustration, let $w = aba$ and $E = (a+b)ab$. then

$$\begin{aligned} D_w(E) &= D_{aba}((a+b)ab) \\ &= D_a(D_{ab}(a+b))ab \\ &= D_a(D_b(D_a(a) + D_a(b)))ab \\ &= D_a(D_b(\varepsilon + \emptyset))ab \\ &= D_a(\emptyset)ab = \emptyset ab = \emptyset \end{aligned}$$

Generalizing the above illustration, the following theorems are obtained.

3.12 Theorem

Let $|w| = n$ and $\|E\| = m$; $n > m$; . then

$$D_w(E) = \emptyset.$$

3.13 Theorem

If $E = w$, then $D_w(E) = \varepsilon$.

3.14 Theorem

If $w = au$, $u \in \Sigma^*$, and $E = av$, $v \in \Sigma^*$. Then

$$D_w(E) = D_u(v), \text{ where } a \in \Sigma.$$

Proof:

Let $u = a_1a_2 \dots a_n$. Then

$$\begin{aligned}
 D_w(E) &= D_{a_n} \left(D_{a_1 a_2 a_3 \dots a_{n-1}} (E) \right) \\
 &= D_{a_n} \left(D_{a_{n-1}} \left(D_{a_1 a_2 a_3 \dots a_{n-2}} (E) \right) \right) \\
 &= D_{a_n} \left(D_{a_{n-1}} \left(D_{a_{n-2}} \dots \right) \right) D_a(E) \\
 &= D_{a_n} \left(D_{a_{n-1}} \left(D_{a_{n-2}} \dots \right) \right) v \\
 &= D_{a_n} \left(D_{a_1 \dots a_{n-1}} \right) v \\
 &= D_{a_1 a_2 a_3 \dots a_n} (v) \\
 &= D_u (v)
 \end{aligned}$$

3.15 Theorem

If $w = au; E = bv$, where $a, b \in \Sigma$ and $u, v \in \Sigma^*$,
 Then $D_w(E) = \emptyset$.

III. CONCLUSION

Brzowski derivatives of the regular expressions are always helpful tool for constructing DFA. The generalizations of the derivatives are useful for transforming

the size of the derivatives of the expressions research oriented work.

REFERENCES

- [1]. Berry G, Sethi R, *From regular expressions to deterministic automata*, Theoretical Computer Science 48 (1986), 117 – 126.
- [2]. Brzowski J.A, *Derivatives of regular expressions*, J.ACM 11(4):481- 494, 1964.
- [3]. Champarnaud J.M, Ponty J.L, Ziadi D, *From regular expressions to finite automata*, Int.J. Com.M, 72(4), 1999, 415 – 431
- [4]. Glushkov V.M, *The Abstract Theory of Automata*, Russian Mathematical Surveys 16(1961), 1-53.
- [5]. Hopcroft J.E and Ullman J.D, *Introduction to Automata Theory, Languages and Computation*, Addison – Wesley, 1979.
- [6]. McNaughton R, Yamada H, *Regular expressions and state graphs for automata*, IRE Trans. on Electronic Computers EC-9:1 (1960) 38 – 47.
- [7]. Murugesan N, *Principles of Automata theory and Computation*, 2004, Sahithi Publications.
- [8]. Murugesan N and Shanmugasundaram O.V, *Construction of State diagram of a regular expression using Derivatives*, www.m-hikari.com/ams/ams-2012/ams.../sundaramAMS21-24-2012.pdf, Applied Mathematical Sciences, Vol. 6, 2012, no. 24, 1173 – 1179.
- [9]. Yu S, Regular Languages, in: A.Salaomaa, eds., *Handbook of Formal Languages*, Vol. I, Springer-Verlag, Berlin, 1997, 41 – 110.