Technical Report No. 9
August, 1980.

NON-REDUNDANT REGULAR EXPRESSIONS AND REAL ARITHMETIC:

PROOF OF A LEMMA DUE TO TARJAN

Roland C. Backhouse
Department of Computer Science
Heriot-Watt University
79 Grassmarket,
Edinburgh.

1. Introduction

Path-finding problems appear in a variety of disguises, mostly obvious but some of which are not readily apparent. Examples of the latter can be found in [Backhouse, 1979; Carré, 1979; Tarjan, 1979]. In recent years it has become evident that there is a small central core of path-finding algorithms, variants of which have been repeatedly rediscovered in novel applications. Indeed, this central core consists almost exclusively of algorithms known to numerical analysts for many, many years which solve a system of linear equations $Ax = b$ where A is a real $n{\times}n$ matrix, x is an $n{\times}1$ vector of variables and b is an $n{\times}1$ vector of real constants.

The connection of path-finding problems with the solution of linear equations has been a fruitful one, both for numerical analysts interested in sparse systems [see e.g. Duff, 1977] and computer scientists in general [e.g. Aho, Hopcroft and Ullman, 1974], but it still cannot be claimed that the connection is completely understood. Some authors have attempted to find a minimal axiom system for which the classical techniques will work [Aho, Hopcroft and Ullman, 1974; Lehman, 1977]. An alternative approach initiated by Backhouse and Carré [1975] and further developed by Tarjan [1979] is to use the algebra of regular expressions, expressing path-finding problems via homomorphisms which map all paths through a graph to the given problem domain.

One particularly important contribution made by Tarjan [1979] was to introduce the notion of a non-redundant regular expression. Tarjan then related the problem of solving the linear system of equations $Ax = b$ in real arithmetic to the problem of finding non-redundant regular expressions, thus clarifying greatly our understanding of the relationship between the two problems. Unfortunately, Tarjan's results were based on a lemma (lemma 2, "The hardest result in this paper") whose proof was incomplete and unnecessarily hedged with qualifications. The sole objective of the present paper is to provide a complete and accurate proof of Tarjan's lemma.

The proof we give of Tarjan's lemma is long and often tedious. Essentially, it consists of a reworking of the well-known technique [Ginzburg, 1968] for establishing whether two regular expressions are equal, but with the additional handicap of establishing whether they are equal when viewed as functions of real numbers. However, the proof does introduce, we believe, novel insights into the relationship between matrix algebra and graph theory. For example,lemmas 7 and 8 prove that the process of reducing a deterministic finite-state machine has an equally valid analogue in real arithmetic. To allay the reader's boredom we have tried to highlight sight insights in introductory comments scattered through the text.

2. Regular Expressions

Let Σ be a finite alphabet containing neither "Λ" nor "$\emptyset$". A regular expression over Σ is any expression built by applying the following rules:

(1a) "Λ" and "$\emptyset$" are atomic regular expressions; for any $a \in \Sigma$, "a" is an atomic regular expression.

(1b) If P and Q are regular expressions then $P + Q$ is a compound regular expression.

(1c) If P and Q are regular expressions then $P \cdot Q$ is a compound regular expression

(1d) If P is a regular expression then P^* is a compound regular expression.

In a regular expression, Λ may be interpreted as the empty word, $\emptyset$ as the empty set, + as set union, $\cdot$ as concatenation and * as reflexive, transitive closure (under concatenation). Thus one interpretation of a regular expression R is a set $\sigma(R)$ of strings. More precisely, we define $\sigma(R)$ as follows:

(2a) $\sigma(\Lambda) = \{\Lambda\}$; $\sigma(\emptyset) = \emptyset$; $\sigma(a) = \{a\}$ for each a in Σ.

(2b) $\sigma(P+Q) = \{w \mid w \in P \text{ or } w \in Q\}$

(2c) $\sigma(P \cdot Q) = \{w \mid w = w_1 \cdot w_2 \text{ where } w_1 \in P \text{ and } w_2 \in Q\}$

(2d) $\sigma(P^*) = \{w \mid w = \Lambda \text{ or } w = w_1 w_2 \ldots w_n \text{ where } w_i \in P., \; 1 \leq i \leq n\}$.

A second interpretation of a regular expression is as a function of real numbers. Specifically, suppose a mapping r is defined from Σ to $\mathbb{R}$ (the set of real numbers). Then we extend r to all regular expressions as follows:

(3a) $r(\Lambda) = 1;\ r(\emptyset) = 0$

(3b) $r(P+Q) = r(P) + r(Q)$

(3c) $r(P\cdot Q) = r(P)\cdot r(Q)$

(3d) $r(P*) = \dfrac{1}{1-r(P)}$

Note that r(R) is not always defined - since inverses don't always exist.

It is useful to introduce some additional terminology: A constant expression is a regular expression built from $\emptyset$ and Λ using only the + operation. A linear expression is a regular expression built from Σ using only the + operation. A constant + linear expression is one built from the atomic expressions using only the + operation. A constant (linear, constant + linear) matrix is a matrix all of whose elements are constant (linear, constant + linear) expressions.

3. Redundancy

A regular expression R is <u>simple</u> if R = $\emptyset$ or R does not contain $\emptyset$ as a subexpression. A regular expression R is <u>non-redundant</u> if each string in $\sigma(R)$ is represented uniquely in R. A more precise definition is as follows:

(4a) Λ, $\emptyset$ and a, for $a \in \Sigma$, are non-redundant.

(4b) Let P and Q be non-redundant.

P+Q is non-redundant if $\sigma(P) \cap \sigma(Q) = \emptyset$

$P \cdot Q$ is non-redundant if each $w \in \sigma(P \cdot Q)$ is uniquely decomposable into $w = uv$ with $u \in \sigma(P)$ and $v \in \sigma(Q)$.

P* is non-redundant if $\Lambda \notin \sigma(P)$ and each $w \neq \Lambda$ in $\sigma(P^*)$ is uniquely decomposable into $w = w_1 w_2 \dots w_k$ with $w_i \in \sigma(P)$ for $1 \leq i \leq k$.

4. Equivalence of Regular Expressions

Let R and Q be regular expressions. R and Q are said to be equivalent (written $R \equiv Q$) if and only if

(i) $\sigma(R) = \sigma(Q)$

(ii) $r(R)$ and $r(Q)$ are both defined, and

(iii) $r(R) = r(Q)$

Below we observe some simple equivalences between regular expressions. These equivalences will be used without comment in the sequel.

Let P,Q,R be regular expressions and suppose r(P), r(Q), and r(R) are all defined. Then, we have

A1 $(P+Q)+R \equiv P+(Q+R)$

A2 $P\cdot(Q+R) \equiv (P\cdot Q) + (P\cdot R)$

A3 $(P+Q)\cdot R \equiv (P\cdot R) + (Q\cdot R)$

A4 $P\cdot(Q\cdot R) \equiv (P\cdot Q)\cdot R$

A5 $P+Q \equiv Q+P$

A6 $P+\emptyset \equiv P$

A7 $P\cdot\Lambda \equiv P \equiv \Lambda\cdot P$

A8 $P\cdot\emptyset \equiv \emptyset \equiv \emptyset\cdot P$

Suppose R is a regular expression and suppose r(R*) is defined. Then

A9 $R^* \equiv \Lambda + R\cdot R^*$

Note that, using A5,A6,A7 and A8 and the observation $\emptyset^*\equiv\Lambda$, it is possible to transform any regular expression R such that r(R) is defined into an equivalent, simple regular expression. This is achieved by repeating the following transformations until none is applicable:

(i) replace any subexpression of the form $\emptyset.P$ or $P.\emptyset$ by $\emptyset$;

(ii) replace any subexpression of the form $\emptyset+P$ or $P+\emptyset$ by P;

(iii) replace any subexpression of the form $\emptyset^*$ by Λ.

We shall use matrix notation as a shorthand for a set of equivalences. Specifically, suppose $Y = [y_1,\ldots,y_m]'$, $B = [b_1,\ldots,b_m]'$ and $C = [c_1,\ldots,c_m]'$ are $m \times 1$ vectors of regular expressions and $A = [a_{ij}]$ is an $m \times m$ matrix of regular expressions.

Then $Y \equiv B$

is short for $y_i \equiv b_i$ $(\forall i,\ 1 \le i \le m)$.

$Y \equiv B+C$

is short for $y_i \equiv b_i + c_i$ $(\forall i,\ 1 \le i \le m)$

$Y \equiv A \cdot B$

is short for $y_i \equiv a_{i1}b_1 + a_{i2}b_2 \ldots +a_{im}b_m$ $(\forall i,\ 1 \le i \le m)$

and $Y \equiv A \cdot B+C$

is short for $y_i \equiv a_{i1}b_1 + a_{i2}b_2 + \ldots + a_{im}b_m + c_i$ $(\forall i,\ 1 \le i \le m)$.

We shall use the notation $\underline{a}_{io}$ and $\underline{a}_{oi}$ for the ith row and ith column (respectively) of the matrix A. $\underline{e}_{io}^{(m)}$ is used to denote the 1xm row vector $[\emptyset,\ldots,\emptyset,\Lambda,\emptyset,\ldots,\emptyset]$ in which the ith element is Λ and the remainder are $\emptyset$. $\underline{e}_{oi}^{(m)}$ is the transpose of $\underline{e}_{io}^{(m)}$ - i.e. it is the corresponding column vector. $\emptyset^{(mxn)}$ is used to denote the mxn matrix all of whose entries are $\emptyset$. The abbreviation $\emptyset^{(m)}$ is used instead of $\emptyset^{(mx1)}$ and $\emptyset^{(m)'}$ instead of $\emptyset^{(1xm)}$. $\underline{1}^{(m)}$ is used to denote the mxm unit matrix (with elements in $\mathbb{R}$).

Finally, suppose $A=[a_{ij}]$ is an mxn matrix of regular expressions. Then $\sigma(A)$ is defined to be $[\sigma(a_{ij})]$ and $r(A)$ is defined to be $[r(a_{ij})]$.

5. The main lemma and its proof - an outline

Our objective is to prove the following lemma.

Lemma M. Suppose P and Q are non-redundant regular expressions and r(P) and r(Q) are both defined. Then P ≡ Q if and only if $\sigma(P) = \sigma(Q)$.

To prove lemma M we show that all the processes used to prove (or disprove) the claim $\sigma(P) = \sigma(Q)$ can be used to prove (or disprove) r(P) = r(Q).

Let us review the steps used to prove that $\sigma(P) = \sigma(Q)$. [Ginzburg, 1967] Firstly, any regular expression may be naturally associated with a transition diagram having a single start node and a single final node. Λ-arcs may then be eliminated from the transition diagram thus producing a non-deterministic machine. The "subset method" may then be applied to construct a deterministic machine all of whose states are accessible from the start state. Finally, equivalent states in the deterministic machine may be "coalesced" to form a reduced machine. When this has been done for the two expressions P and Q, $\sigma(P) = \sigma(Q)$ if and only if the two machines are identical (up to renaming of states).

Algebraically, each of these steps amounts to constructing an equational characterisation $\sigma(Y) = \sigma(A)\sigma(Y) + \sigma(B)$ of the given expression P in which Y and B are column vectors and A is a matrix. The different types of machine are expressed by constraints on A. B is a constant vector indicating which states are the final states and the element y_s of Y corresponding to the start state s is P. (We shall always arrange that s = 1.)

Our proof that P ≡ Q iff $\sigma(P) = \sigma(Q)$ therefore consists of showing that the algebraic manipulations implicit in constructing reduced machines for P and Q do not violate the rules of real arithmetic and that <u>the resulting equations have a unique solution</u> whether the regular expressions are interpreted as sets or as functions of real numbers. Thus we can conclude that if $\sigma(P) = \sigma(Q)$, r(P) and r(Q) are solutions of the same <u>non-singular</u> system of simultaneous equations and hence are equal.

6. Preliminary Lemmas

Suppose $A = [a_{ij}]$ is an mxm matrix of regular expressions. We say that A is definite if and only if r(A) is defined and, forall mx1 vectors T,

$$T \equiv A \cdot T \implies T \equiv \emptyset^{(m)} .$$

Lemma 1 Suppose B is an mx1 vector of regular expressions and suppose A is a definite mxm matrix of regular expressions. Then the equation

$$Y = \sigma(A)Y + \sigma(B)$$

has the unique solution

$$Y = \sigma(A)^*\sigma(B)$$

and the equation

$$Y = r(A)Y + r(B)$$

has the unique solution

$$Y = (\underline{1}^{(m)} - r(A))^{-1} r(B)$$

Proof If A is definite then σ(A) does not possess the "empty-word property" [Backhouse and Carré, 1975], hence $Y = \sigma(A)Y + \sigma(B)$ has the unique solution $Y = \sigma(A)^*\sigma(B)$ [Salomaa, 1969].

Also, if A is definite then r(A) is defined and $\underline{1}^{(m)} - r(A)$ is non-singular. Thus the equation $Y = r(A)Y + r(B)$ has the unique solution $Y = (\underline{1}^{(m)} - r(A))^{-1} r(B)$. □

In view of lemma 1 let us use the notation A* as a shorthand within equivalences as follows:

Let f(P,Q,...) be an expression involving P,Q,... and the operations + and · . Then the equivalence

$$Y \equiv f(A^*, P, Q)$$

is a shorthand for

$$\sigma(Y) = f(\sigma(A)^*, \sigma(P), \sigma(Q))$$

and

$$r(Y) = f((\underline{1}^{(m)} - r(A))^{-1}, r(P), r(Q)).$$

Using this shorthand we can rephrase lemma 1 as:

Lemma 1 Suppose B is an mx1 vector of regular expressions such that r(B) is defined, and suppose A is a definite mxm matrix of regular expressions.

Then the equation

$$Y \equiv AY + B$$

has the unique solution

$$Y \equiv A^*B \qquad \square$$

One tactic which is used in constructing a deterministic machine from a regular expression is to add extra Λ-arcs to an existing graph (possibly also adding extra nodes). As we shall see, this is modelled algebraically as adding the product CD of an mx1 vector C and a 1xm vector D to an mxm matrix B. We therefore need two general results concerning the definiteness of B+CD.

Lemma 2 Suppose B, C and D are, respectively, an mxm, mx1 and 1xm matrix of regular expressions. Suppose $A \equiv B+CD$. Then A is definite if B and DB*C are definite.

Proof Suppose $T \equiv AT$, $A \equiv B+CD$ and B and CB*D are definite. We will prove that $T \equiv \underline{\emptyset}^{(m)}$.

Now, since $\quad T \equiv AT,$

$$T \equiv (B+CD)T \equiv BT + CDT \qquad (1)$$

So, since B is definite,

$$T \equiv B^*CDT$$

$$\therefore \qquad DT \equiv DB^*CDT \qquad (2)$$

But DB*C is definite. We conclude from (2) that

$$DT \equiv \emptyset$$

Substituting in (1),

$$T \equiv BT$$

But, since B is definite, we conclude that

$$T \equiv \emptyset^{(m)}. \qquad \square$$

Lemma 3 Suppose the mxm matrix A is definite. Suppose also that $A \equiv B+CD$ where B is mxm, C is mx1 and D is 1xm. Then $\begin{bmatrix} B & C \\ D & \emptyset \end{bmatrix}$ is also definite.

Proof Suppose $T \equiv \begin{bmatrix} B & C \\ D & \emptyset \end{bmatrix} T$

Define the mx1 and 1x1 vectors T_1 and T_2 by $T = \begin{bmatrix} T_1 \\ T_2 \end{bmatrix}$.

Then $T_1 \equiv BT_1 + CT_2$

and $T_2 \equiv DT_1$

Hence $T_1 \equiv (B + CD)T_1 \equiv AT_1$

But A is definite. Therefore $T_1 \equiv \underline{\emptyset}^{(m)}$.

Thus $T_2 \equiv DT_1 \equiv \emptyset$.

I.e. $T \equiv \underline{\emptyset}^{(m+1)}$ and $\begin{bmatrix} B & C \\ D & \emptyset \end{bmatrix}$ is definite. $\square$

7. Equational Characterisation

Let R be a simple and non-redundant regular expression. We say that R is <u>equationally characterised in terms of</u> Y,A,B (or Y,A,B is <u>an equational characterisation</u> of R) if and only if

1. Y is $[y_1,\ldots,y_m]'$, an mxl vector of regular expressions, for some $m \geq 1$, such that

   (a) y_i is simple and non-redundant $(1 \leq i \leq m)$

   (b) R is y_1

   (c) For all i, $1 \leq i \leq m$, $\exists\, u \in \Sigma^*$ such that $\{u\}\cdot\sigma(y_i) \subseteq \sigma(R)$.

2. A is a definite, constant + linear matrix of order mxm.

3. B is an mxl constant vector.

4. $Y \equiv AY + B$

5. $AY + B$ is an mxl vector of non-redundant regular expressions.

Remarks: We can regard Y,A,B as specifying a transition diagram with start node 1 (condition 1(b)). The final nodes are those nodes j such that b_j is Λ. The diagram is all-admissible (condition 1(c)) and y_i is a regular expression describing the set of all words taking node i to a final node (condition 4).

Note that condition 1(b) can be expressed by the matrix formula

$$R \equiv \underline{e}^{(m)}_{10} Y$$

We shall make use of this formula later.

8. The Proof

The first step in the proof of lemma M is to show how to construct a transition diagram from a given regular expression (lemma 4, below). Much of lemma 4 is tedious, being concerned with checking the definiteness and non-redundancy of the equational characterisation. By far the most interesting part of the proof of lemma 4 is (e) in which it is shown that a definite equational characterisation of a non-redundant regular expression of the form P* can always be constructed. The property of definiteness was not established by Tarjan; consequently he was only able to claim the validity of his lemma "except on a set of measure zero" rather than everywhere as we shall establish.

<u>Lemma 4</u> (Construction of a transition diagram from a regular expression.)

Let $R \neq \emptyset$ be a simple and non-redundant regular expression such that $r(R)$ is defined. Then $\exists\, Y' = [y_1, \ldots, y_m]'$, $A = [a_{ij}]$ and $B = [b_1 \ldots b_m]'$ such that R is equationally characterised by Y, A and B. Moreover A and B have the additional properties.

$$6. \quad B = \underline{e}_{om}^{(m)}$$

$$7. \quad \underline{a}_{o1} = \underline{\emptyset}^{(m)} \text{ and } \underline{a}_{mo} = \underline{\emptyset}^{(m)'}.$$

(i.e. node m is the only final node, and there are no arcs entering node 1 or leaving node m.)

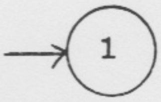
<u>Proof</u> Suppose $R \neq \emptyset$ is a simple and non-redundant regular expression such that $r(R)$ is defined. We argue by induction on the structure of R.

(a) Suppose R is Λ.

Then $\Lambda \equiv [\emptyset]\, \Lambda + \Lambda$.

i.e. $[\Lambda]$, $[\emptyset]$, $[\Lambda]$ is an equational characterisation of R.

⌈ This corresponds to the diagram



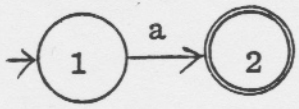
⌋

(b) Suppose R is a, an element of Σ.

Then $$\begin{bmatrix} a \\ \Lambda \end{bmatrix} \equiv \begin{bmatrix} \emptyset & a \\ \emptyset & \emptyset \end{bmatrix} \begin{bmatrix} a \\ \Lambda \end{bmatrix} + \begin{bmatrix} \emptyset \\ \Lambda \end{bmatrix}$$

i.e. $\begin{bmatrix} a \\ \Lambda \end{bmatrix}$, $\begin{bmatrix} \emptyset & a \\ \emptyset & \emptyset \end{bmatrix}$, $\begin{bmatrix} \emptyset \\ \Lambda \end{bmatrix}$ is an equational characterisation of R.

⌈This corresponds to the diagram



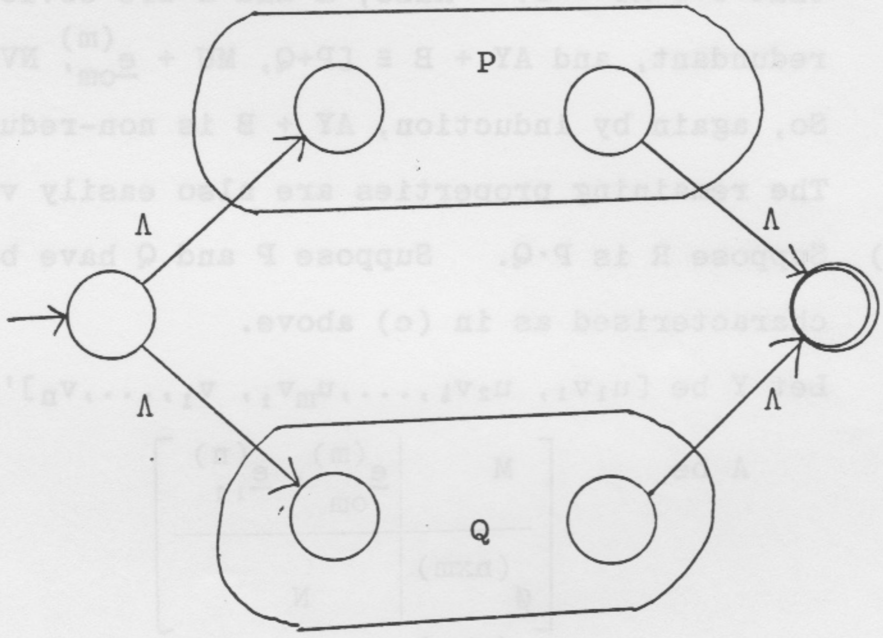
⌋

(c) Suppose R is P+Q. Then if R is simple and non-redundant so too must be P and Q. Thus, by induction we may suppose that P is characterised by $U = [u_1, \ldots, u_m]'$, M and $\underline{e}_{om}^{(m)}$ and Q is characterised by V = $[v_1,\dots,v_n]'$, N, and $\underline{e}_{on}^{(n)}$. Let Y be $[P+Q, u_1,\dots,u_m,v_1,\dots,v_n,\Lambda]'$,

A be
$$\left[\begin{array}{c|c|c|c} & \underline{e}_{10}^{(m)} & \underline{e}_{10}^{(n)} & \emptyset \\ \hline \underline{\emptyset}^{(m+n+1)} & M & \underline{\emptyset}^{(mxn)} & \underline{e}_{om}^{(m)} \\ \hline & \underline{\emptyset}^{(nxm)} & N & \underline{e}_{on}^{(n)} \\ \hline \emptyset & \multicolumn{3}{c}{\underline{\emptyset}^{(m+n+1)'}} \end{array}\right]$$

and B be $\underline{e}_{o,m+n+2}^{(m+n+2)}$.

This construction is equivalent to



Clearly, properties 1(a),(b) and (c) hold by induction and the assumption that P+Q is simple and non-redundant.

To prove that A is definite, suppose T ≡ AT.

Let T = $\begin{bmatrix} T_1 \\ T_2 \\ T_3 \\ T_4 \end{bmatrix}$ where T_1, T_2, T_3, and T_4 are

are respectively of order 1x1, mx1, nx1 and 1x1.

Then

$$
\begin{aligned}
T_1 &\equiv \underline{e}_{10}^{(m)} T_2 + \underline{e}_{10}^{(n)} T_3 \\
T_2 &\equiv M\cdot T_2 + \underline{e}_{om}^{(m)} T_4 \\
T_3 &\equiv N\cdot T_3 + \underline{e}_{on}^{(n)} T_4 \\
T_4 &\equiv \emptyset\cdot T_4 \equiv \emptyset
\end{aligned}
$$

Hence, by induction, $T_2 \equiv \underline{\emptyset}^{(m)}$ and $T_3 \equiv \underline{\emptyset}^{(n)}$.

$\therefore$ $T_1 \equiv \emptyset$ and $T \equiv \underline{\emptyset}^{(m+n+2)}$. i.e. A is definite.

It is easily verified using the induction hypothesis that $Y \equiv AY + B$. Also, A and B are obviously non-redundant, and $AY + B \equiv [P+Q,\ MU + \underline{e}_{om}^{(m)},\ NV + \underline{e}_{on}^{(n)},\ \Lambda]$. So, again by induction, AY + B is non-redundant. The remaining properties are also easily verified.

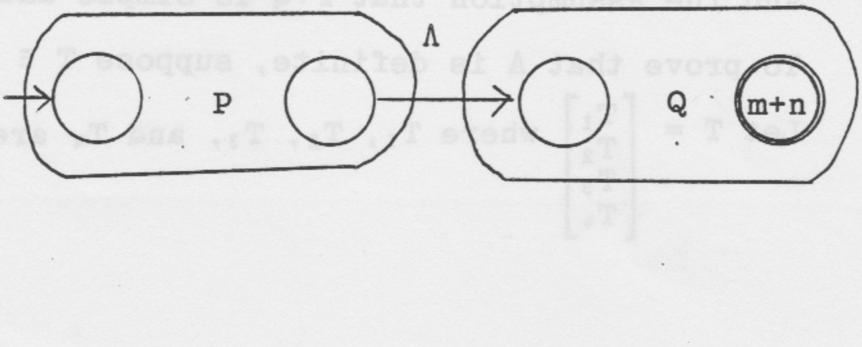
(d) Suppose R is P·Q. Suppose P and Q have been characterised as in (c) above.

Let Y be $[u_1v_1, u_2v_1, \ldots, u_mv_1, v_1, \ldots, v_n]'$,

A be
$$
\left[\begin{array}{c|c}
M & \underline{e}_{om}^{(m)} \cdot \underline{e}_{10}^{(n)} \\
\hline
\underline{\emptyset}^{(nxm)} & N
\end{array}\right]
$$

and B be $\underline{e}_{o,m+n}^{(m+n)}$

This construction is equivalent to



Firstly, each element in Y is non-redundant.

For, if not, $\exists i, w, w_1, w_2, w_1', w_2'$ such that

$$w = w_1 w_2 = w_1' w_2' \ \varepsilon\ \sigma(u_i v_1)$$

where $w_1, w_1' \ \varepsilon\ \sigma(u_i), w_2, w_2' \ \varepsilon\ \sigma(v_1)$

and $w_1 \neq w_1'$ and $w_2 \neq w_2'$.

But property 1(c) holds on U. That is, $\exists w_3 \ \varepsilon\ \Sigma^*$ such that $w_3 w_1 \ \varepsilon\ \sigma(u_1)$ and $w_3 w_1' \ \varepsilon\ \sigma(u_1)$.

Hence $(w_3 w_1) w_2 = (w_3 w_1') w_2' \ \varepsilon\ \sigma(u_1 v_1) = \sigma(P \cdot Q)$

contradicting the non-redundancy of $P \cdot Q$.

Properties 1(b) and 1(c) are easily established on Y.

To establish property 2, suppose $T = \begin{bmatrix} T_1 \\ T_2 \end{bmatrix}$ where

T_1 is mxl and T_2 is nxl. Suppose $T \equiv AT$. where

Then $T_1 \equiv MT_1 + \underline{e}_{om}^{(m)} \cdot \underline{e}_{10}^{(n)} T_2$

and $T_2 \equiv NT_2$

Thus, $T_2 \equiv \underline{\emptyset}^{(n)}$ (since N is definite)

$\therefore$ $T_1 \equiv MT_1$

Hence $T_1 \equiv \underline{\emptyset}^{(m)}$ (since M is definite)

i.e. $T \equiv \underline{\emptyset}^{(m+n)}$.

To prove that $Y \equiv AY + B$, note that

$$U \equiv MU + \underline{e}_{om}^{(m)}$$

Hence $Uv_1 \equiv MUv_1 + \underline{e}_{om}^{(m)} v_1$

But $v_1 \equiv \underline{e}_{10}^{(n)} V$

$\therefore$ $Uv_1 \equiv MUv_1 + \underline{e}_{om}^{(m)} \cdot \underline{e}_{10}^{(n)} V$

Also, $V \equiv NV + \underline{e}_{on}^{(n)}$

Hence
$$\left[\begin{array}{c} Uv_1 \\ \hline V \end{array}\right] \equiv \left[\begin{array}{c|c} M & \underline{e}_{om}^{(m)} \cdot \underline{e}_{10}^{(n)} \\ \hline \emptyset^{(nxm)} & N \end{array}\right] \left[\begin{array}{c} Uv_1 \\ \hline V \end{array}\right] + \underline{e}_{o,m+n}^{(m+n)}$$

i.e. $Y \equiv AY + B$.

Consider now property 4. By the induction hypothesis, AY + B is non-redundant if $MUv_1 + \underline{e}_{om}^{(m)}\ \underline{e}_{10}^{(n)}V$ is non-redundant.

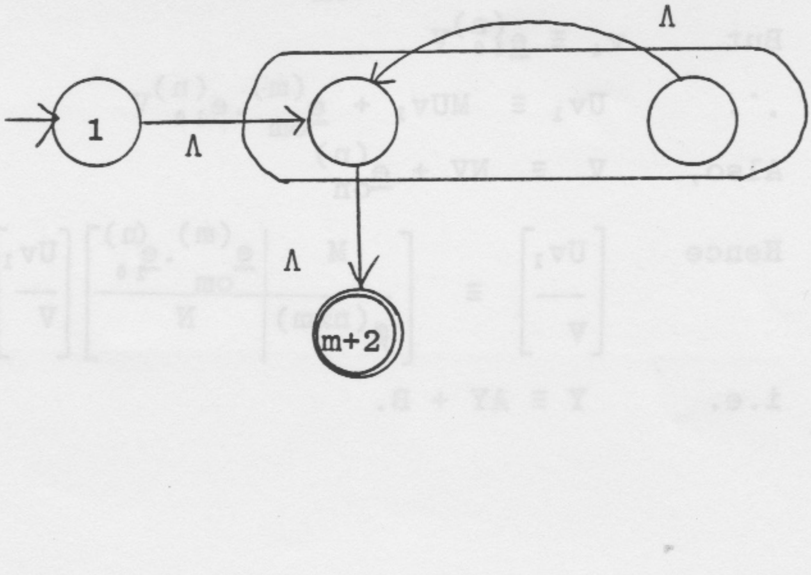
Now, $U \equiv MU + \underline{e}_{om}^{(m)}$, by the induction hypothesis, and $MU + \underline{e}_{om}^{(m)}$ is non-redundant. Therefore, MU is non-redundant and $\sigma(MU) \subseteq \sigma(u_1)$. Hence, since u_1v_1 is non-redundant by assumption, MUv_1 must be non-redundant. Finally, $\underline{e}_{om}^{(m)} \cdot \underline{e}_{10}^{(n)}V$ is obviously non-redundant and so (again making use of the non-redundancy of $MU + \underline{e}_{om}^{(m)}$) $(MUv_1) + (\underline{e}_{om}^{(m)}\ \underline{e}_{10}^{(n)}V)$ is non-redundant.

(e) Suppose R is P* and P has been characterised as in (c) above. Let Y be $[P^*, UP^*+\underline{e}_{10}^{(m)}, \Lambda]'$ (i.e. $[R, u_1R+\Lambda, u_2R, \ldots, u_mR, \Lambda]'$),

A be $\left[\begin{array}{c|c|c} & \underline{e}_{10}^{(m)} & \emptyset \\ \hline \underline{\emptyset}^{(m+2)} & M+\underline{e}_{om}^{(m)}\ \underline{e}_{10}^{(m)} & \underline{e}_{01}^{(m)} \\ \hline & \underline{\emptyset}^{(m)'} & \emptyset \end{array}\right]$

and B be $\underline{e}_{o,m+2}^{(m+2)}$.

This construction is equivalent to



Note: $M + \underline{e}_{om}^{(m)}\ \underline{e}_{10}^{(m)}$ is constructed by adding Λ to the (m,1)th entry of M which, by induction, is $\emptyset$. Hence

$$r(M + \underline{e}_{om}^{(m)}\ \underline{e}_{10}^{(m)}) = r(M) + r(\underline{e}_{om}^{(m)}\ \underline{e}_{10}^{(m)})$$

$$= r(M) + r(\underline{e}_{om}^{(m)})r(\underline{e}_{10}^{(m)}).$$

Properties 1(a)-(c) are easily established on Y.

Consider the definiteness of A (property 2). It is easy to show that A is definite if $M + \underline{e}_{om}^{(m)}\ \underline{e}_{10}^{(m)}$ is definite. But M is definite.

Also, $\sigma(\underline{e}_{10}^{(m)})\sigma(M)^*\sigma(\underline{e}_{om}^{(m)}) = \sigma(P)$ (1)

and $r(\underline{e}_{10}^{(m)})\ (\underline{1}^{(m)} - r(M))^{-1}r(\underline{e}_{om}^{(m)}) = r(P)$ (2)

Thus, since P* is non-redundant,

$$\Lambda \notin \sigma(\underline{e}_{10}^{(m)})\sigma(M)^*\sigma(\underline{e}_{om}^{(m)})$$

and, since $r(P^*) = (1 - r(P))^{-1}$ is defined,

$$r(\underline{e}_{10}^{(m)})\ (\underline{1}^{(m)} - r(M))^{-1}r(\underline{e}_{om}^{(m)}) \neq 1$$

i.e. $\underline{e}_{10}^{(m)}M^*\underline{e}_{om}^{(m)}$ is definite and, using lemma 2, A is definite.

[Eq.(1) is proved as follows:

$$P \equiv \underline{e}_{10}^{(m)}U \text{ and } U \equiv MU + \underline{e}_{om}^{(m)}$$

by the inductive hypothesis. Whence, since M is definite,

$$\sigma(U) = \sigma(M)^*\sigma(\underline{e}_{om}^{(m)}).$$

Thus $\sigma(P) = \sigma(\underline{e}_{10}^{(m)})\sigma(M)^*\sigma(\underline{e}_{om}^{(m)})$.

Similarly, $r(U) = (\underline{1}^{(m)} - r(M))^{-1}\ r(\underline{e}_{om}^{(m)})$

and so $r(P) = r(\underline{e}_{10}^{(m)})\ (\underline{1}^{(m)} - r(M))^{-1}\ r(\underline{e}_{om}^{(m)})$.

This is equation (2).]

Now, consider property 4.

We have, $U \equiv MU + \underline{e}_{om}^{(m)}$ (induction hypothesis)

Hence, $UP^* + \underline{e}_{01}^{(m)} \equiv (MU + \underline{e}_{om}^{(m)})P^* + \underline{e}_{01}^{(m)}$

Also, $P \equiv \underline{e}_{10}^{(m)}U$

and $P^* \equiv PP^* + \Lambda$

$\therefore UP^* + \underline{e}_{01}^{(m)} \equiv MUP^* + \underline{e}_{om}^{(m)}(\underline{e}_{10}^{(m)}UP^* + \Lambda) + \underline{e}_{01}^{(m)}$

But $M\underline{e}_{01}^{(m)} \equiv \underline{\emptyset}^{(m)}$ (property 7 of the induction hypothesis)

and $\underline{e}_{om}^{(m)} \equiv \underline{e}_{om}^{(m)} . \underline{e}_{10}^{(m)} . \underline{e}_{01}^{(m)}$

$\therefore UP^* + \underline{e}_{01}^{(m)} \equiv (M + \underline{e}_{om}^{(m)}\ \underline{e}_{10}^{(m)})(UP^* + \underline{e}_{01}^{(m)}) + \underline{e}_{01}^{(m)}$

It is now straightforward to show that $Y \equiv AY+B$.

Finally, consider property 5. (The remainder are obvious). We have to show that $AY+B$ is a vector of non-redundant regular expressions. Now, we have already observed that $M + \underline{e}_{om}^{(m)}\ \underline{e}_{10}^{(m)}$ is non-redundant and B is obviously so. Thus, it amounts to showing that $(M + \underline{e}_{om}^{(m)}\ \underline{e}_{10}^{(m)})(UP^* + \underline{e}_{01}^{(m)}) + \underline{e}_{01}^{(m)}$ is a vector of non-redundant regular expressions knowing that U, $P \equiv \underline{e}_{10}^{(m)}U$, P^* and $MU + \underline{e}_{om}^{(m)}$ are all non-redundant. The proof parallels that given in case (d) above and so is omitted. □

The next step is to show in algebraic terms how to construct a deterministic finite state machine from the transition diagram corresponding to R.

It simplifies the proofs if we consider two separate processes - one which introduces Λ-arcs followed by one which eliminates them. The next lemma relates to the elimination of Λ-arcs.

Lemma 5

Let $R \neq \emptyset$ be a simple, non-redundant regular expression and suppose Y, A, B is an equational characterisation of R. Then there is an equational characterisation Y, M, D of R related to Y, A, B as follows:

(a) $A \equiv C+L$ where C and L are non-redundant constant and and linear matrices, respectively.

(b) $M \equiv C^*L$ and $D \equiv C^*B$.

Proof Since, by definition, A is a constant+linear matrix and each entry a_{ij} is non-redundant it is easy to abstract constant and linear matrices C and L such that

$$A \equiv C+L$$

Hence $\quad Y \equiv (C+L)Y+B \equiv CY+LY+B$

Now, C must be acyclic (since A is definite).

Therefore, C is definite, and

$$Y \equiv C^*LY+C^*B. \qquad (1)$$

Construct the non-redundant, linear matrix M as follows:

Each entry m_{ij} is $\emptyset$ or the sum of elements $a \epsilon \Sigma$ where $a \epsilon m_{ij} \iff a \epsilon \ell_{ij}$ or $\exists\, k \geq 1$ and indices $i_0, i_1, \ldots, i_k$ such that $i_0 = i$, $\Lambda \epsilon c_{i_{s-1}i_s}$ $(1 \leq s \leq k)$ and $a \epsilon \ell_{i_s j}$.

Clearly, $\sigma(M) = \sigma(C)\sigma(M) + \sigma(L)$

Hence, since C is definite,

$$\sigma(M) = \sigma(C)^*\sigma(L) \qquad (2)$$

To prove that

$$r(M) \underline{=} (1^{(m)} - r(C))^{-1} r(L) \qquad (3)$$

consider the matrix Q of regular expressions, where

$$q_{ij} = \sum_{k=1}^{m} c_{ik} m_{kj} + \ell_{ij}$$

We claim that Q is non-redundant.

Suppose otherwise. Since M,C and L are, by construction, non-redundant there are only two possibilities

(i) $c_{ik_1} m_{k_1 j} + c_{ik_2} m_{k_2 j}$ is non-redundant for some i,j,k_1 and k_2

(ii) $c_{ik} m_{kj} + \ell_{ij}$ is non-redundant for some i,j and k.

<u>Case (i)</u>. Consider the non-redundant expression

$$a_{i1} y_1 + a_{i2} y_2 + \dots + a_{im} y_m + b_m.$$

Now, $\sigma(Y) = \sigma(C)^*\sigma(L)\sigma(Y) + \sigma(C)^*\sigma(B)$

and $\sigma(M) = \sigma(C)^*\sigma(L)$.

$\therefore \quad \sigma(Y) = \sigma(M)\sigma(Y) + \sigma(C)^*\sigma(B).$ (4)

Thus, $a \varepsilon \sigma(c_{ik_1} m_{k_1 j}) \cap \sigma(c_{ik_2} m_{k_2 j})$

$\Rightarrow \quad a \varepsilon \sigma(m_{k_1 j}) \cap \sigma(m_{k_2 j})$

and $\Lambda \varepsilon \sigma(a_{ik_1}) \cap \sigma(a_{ik_2})$

$\Rightarrow \exists$ $w \varepsilon \sigma(y_j)$ such that

$aw \varepsilon \sigma(y_{k_1}) \cap \sigma(y_{k_2})$

and $\Lambda \varepsilon \sigma(a_{ik_1}) \cap \sigma(a_{ik_2})$

$\Rightarrow \quad \sigma(a_{ik_1} y_{k_1}) \cap \sigma(a_{ik_2} y_{k_2}) \neq \emptyset$, i.e. AY+B is non-redundant.

This is a contradiction.

<u>Case (ii)</u> is proved similarly. Suppose $a \varepsilon \sigma(c_{ik} m_{kj}) \cap \sigma(\ell_{ij})$. Hence $a \varepsilon \sigma(m_{kj}) \cap \sigma(a_{ij})$ and $\Lambda \varepsilon a_{ik}$. Let $w \varepsilon \sigma(y_j)$. Then $aw \varepsilon y_k$ (using (4)) and so $aw \varepsilon \sigma(a_{ik} y_k)$. Also, $a \varepsilon \sigma(a_{ij} y_j)$. Hence

$a_{ik}\, y_k + a_{ij}\, y_j$ is non-redundant which contradicts the non-redundancy of AY + B.

Now, $$\sigma(m_{ij}) = \bigcup_{k=1}^{m} \sigma(c_{ik})\sigma(m_{kj}) \cup \sigma(\ell_{ij})$$

and, since both m_{ij} and $\sum_{k=1}^{m} c_{ik}m_{kj} + \ell_{ij}$ are linear, non-redundant expressions, it is clear that

$$r(m_{ij}) = \sum_{k=1}^{m} r(c_{ik})r(m_{kj}) + r(\ell_{ij}).$$

i.e. $r(M) = r(C)r(M) + r(L)$.

and, so $r(M) = (\underline{1}^{(m)} - r(C))^{-1} r(L)$.

We have thus proved (3).

Now, construct the non-redundant constant vector D as follows. Each entry d_i is $\emptyset$ or Λ. d_i is Λ if b_i is Λ or $\exists\, k \geq 1$ and indices $i_0, i_1, \ldots, i_k$ such that $\Lambda \in c_{i_{s-1} i_s}$ $(1 \leq s \leq k)$, $i_0 = i$ and $b_{i_s} = \Lambda$.

We can now repeat the argument used on M to show that

$$\sigma(D) = \sigma(C)^* \sigma(B) \qquad (5)$$

and $$r(D) = (\underline{1}^{(m)} - r(C))^{-1} r(B) \qquad (6)$$

Finally, combining (1),(2),(3),(5) and (6) we have

$$\underset{\sim}{Y} \equiv MY + D$$

and $$M \equiv C^*L$$

M is definite because

$$T \equiv MT$$

$$\Rightarrow \quad T \equiv C^*LT \equiv (\Lambda + CC^*)LT$$

$$\equiv LT + C(C^*LT)$$

$$\equiv LT + CT$$

$$\equiv AT$$

$$\Rightarrow \quad T \equiv \underline{\emptyset}^{(m)} \quad \text{since A is definite.}$$

Lastly, we can prove that MY + D is non-redundant by a similar argument to that used in cases (i) and (ii) above. □

Now we show how to construct a deterministic machine for a simple, non-redundant regular expression R.

Lemma 6

Let $R \neq \emptyset$ be a simple, non-redundant regular expression. Then $\exists$ an equational characterisation Y,A,B of R such that $A = [a_{ij}]$ ($1 \le i,j \le m$, for some m) is a linear matrix and $\sigma(a_{ij}) \cap \sigma(a_{ik}) = \emptyset$ for all i ($1 \le i \le m$) and all j,k, $1 \le j \ne k \le m$.

Proof We begin by describing two procedures both of which transform an equational characterisation U,M,C in which M is linear into another equational characterisation Y,A,B in which A is linear.

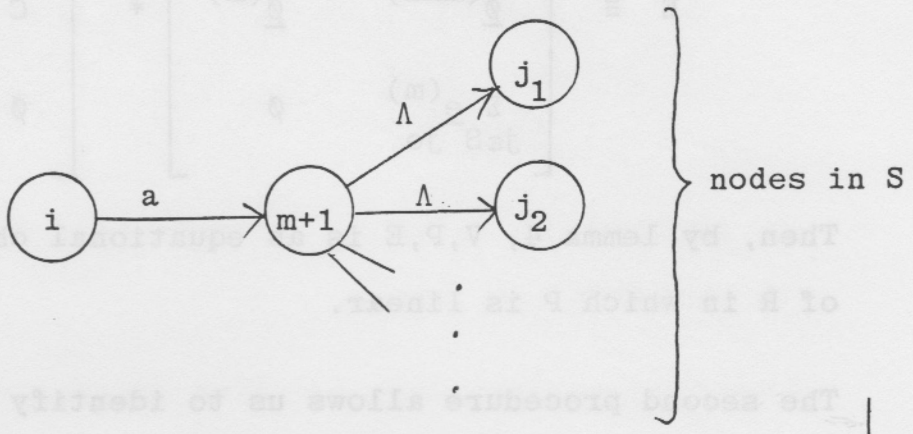
Procedure N (Removal of non-determinism).

Suppose U,M,C is an equational characterisation of R where M is a linear, mxm matrix. Consider node i of M and suppose there is an arc labelled a ($a \in \Sigma$) from i to each node j in some set S. Suppose the cardinality of S is greater than 1. Define the non-redundant matrix M by

$$M \equiv \mathit{M} + a\underline{e}_{oi}^{(m)} . \sum_{j \in S} \underline{e}_{jo}^{(m)}$$

Construct $N = \left[\begin{array}{c|c} \mathit{M} & a\underline{e}_{oi}^{(m)} \\ \hline \sum\limits_{j \varepsilon S} \underline{e}_{jo}^{(m)} & \emptyset \end{array}\right]$

This corresponds to introducing an (m+1)th node and connecting it to node i and nodes j ε S as shown below:



Let V be $[u_1, \ldots, u_m, \sum_{j \varepsilon S} u_j]'$
and D be $[C, \emptyset]'$.

We claim that V,N,D is an equational characterisation of R. In fact this is easily checked - V and NV+D are non-redundant because of the non-redundancy of MU+C, N is definite by lemma 3 and, obviously, V ≡ NV+D.

Now, eliminate the Λ-arcs from N using lemma 4. That is, construct the non-redundant matrix P such that

$$P \equiv \begin{bmatrix} \underline{\emptyset}^{(m\text{x}m)} & \underline{\emptyset}^{(m)} \\ \sum\limits_{j\varepsilon S} \underline{e}^{(m)}_{jo} & \emptyset \end{bmatrix}^{*} \cdot \begin{bmatrix} M & a\underline{e}^{(m)}_{oi} \\ \underline{\emptyset}^{(m)'} & \emptyset \end{bmatrix}$$

Note that $p_{ij} = [M]_{ij}$ when $1 \le j \le m$.

Thus the only arc labelled a leaving node i enters node m+1 (which corresponds to S).

Construct also the non-redundant vector

$$E \equiv \begin{bmatrix} \underline{\emptyset}^{(mxm)} & \underline{\emptyset}^{(m)} \\ \sum_{j\varepsilon S} \underline{e}_{jo}^{(m)} & \emptyset \end{bmatrix} * \begin{bmatrix} C \\ \emptyset \end{bmatrix}$$

Then, by lemma 4, V,P,E is an equational characterisation of R in which P is linear.

The second procedure allows us to identify and coalesce 'similar' nodes constructed by procedure N.

<u>Procedure C</u> (Coalescing similar nodes).

Suppose U,M,C is an equational characterisation of R in which M is an mxm, linear matrix. We say that i and j $(1 \leq i,j \leq m)$ are <u>similar</u> in the characterisation if $\underline{\underline{m}}_{io} \equiv \underline{\underline{m}}_{jo}$ and $c_i \equiv c_j$. Note that if i and j are similar then $u_i \equiv u_j$, although the converse is not necessarily true. Testing similarity is straightforward since, by the linearity and non-redundancy of M, $\underline{\underline{m}}_{io} \equiv \underline{\underline{m}}_{jo}$ if and only if the vectors of finite sets $\sigma(\underline{\underline{m}}_{io})$ and $\sigma(\underline{\underline{m}}_{jo})$ are equal. Also, since C is a non-redundant, constant matrix, $c_i \equiv c_j$ if and only if c_i and c_j are both Λ or both $\emptyset$.

If i and j are similar in U,M,C and $i \neq 1 \neq j$ we can coalesce them into one node as follows. Define the non-redundant matrix $\mathcal{M}$ by

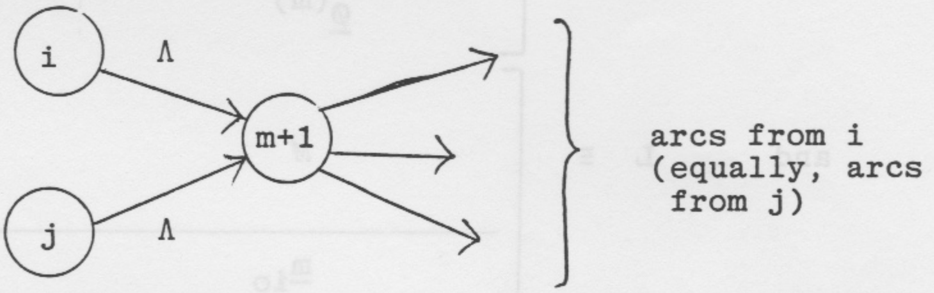
$$M \equiv \mathcal{M} + \underline{\underline{m}}_{io} + \underline{\underline{m}}_{jo}.$$

Construct

$$N = \left[\begin{array}{c|c} M & \underline{e}^{(m)}_{oi} + \underline{e}^{(m)}_{oj} \\ \hline \underline{m}_{io} & \emptyset \end{array}\right]$$

and $D = [c_1,\ldots,c_{i-1},\emptyset,c_{i+1},\ldots,c_{j-1},\emptyset,c_{j+1},\ldots,c_m,c_i]'$.

This corresponds to introducing an (m+1)th node, connecting it to i and j as shown below, making i and j non-final nodes and m+1 a final node if i was a final node.



Let V be $[U,u_i]$. Once again it is easy to show that V,N,D is an equational characterisation of R.

Now eliminate the Λ-arcs and the nodes i and j as follows:

Let W be $[v_1,\ldots,v_{i-1},v_{i+1},\ldots,v_{j-1},v_{j+1},\ldots,v_{m+1}]'$

Let P be $[p_{st}]$ where

$$\begin{array}{lcll} p_{st} & = & m_{st} & \text{if} \quad 1 \le s,t \le m \\ p_{s,m+1} & = & m_{si} & \text{if} \quad 1 \le s \le m \\ p_{m+1,s} & = & m_{is} & \text{if} \quad 1 \le s \le m \\ p_{m+1,m+1} & = & m_{ii}+m_{ij} & \end{array}$$

Construct Q from P by removing rows i and j and columns i and j.

Finally, let E be $[d_1,\dots,d_{i-1},d_{i+1},\dots,d_{j-1},d_{j+1},\dots,d_{m+1}]'$. We claim that W,Q,E is an equational characterisation of R.

The proof of this claim relies on noting the following subsidiary claims:

(i) $P \equiv LF^*$

(ii) $\underline{e}_{ok}^{(m+1)}F^* \equiv \underline{e}_{ok}^{(m+1)}$ when $k \neq i$ and $k \neq j$.

(iii) $\underline{p}_{io} \equiv \underline{p}_{jo} \equiv \emptyset^{(m+1)}$,

where
$$F \equiv \left[\begin{array}{c|c} \underline{\emptyset}^{(m\times m)} & \underline{e}_{oi}^{(m)} + \underline{e}_{oj}^{(m)} \\ \hline \underline{\emptyset}^{(m)'} & \emptyset \end{array}\right]$$

and
$$L \equiv \left[\begin{array}{c|c} M & \emptyset^{(m)} \\ \hline \underline{m}_{io} & \emptyset \end{array}\right]$$

Now $\quad V \equiv (L+F)V + D$

Hence, since $N \equiv L+F$ is definite,

$$V \equiv (L+F)^*D$$

But F is obviously definite. Hence LF^* is also definite

and $\quad V \equiv F^*(LF^*)^*D$

(i.e. $r(V) = (1^{(m+1)}-r(F))^{-1}(1^{(m+1)}-r(L)\cdot(1^{(m+1)}-r(F))^{-1})^{-1}r(D)$

and $\sigma(V) = \sigma(F)^*(\sigma(L)\sigma(F)^*)^*\sigma(D)$).

Therefore, if $k \neq i$ and $k \neq j$,

$$\begin{aligned} v_k \equiv \underline{e}_{ok}^{(m+1)}V &\equiv \underline{e}_{ok}^{(m+1)}F^*(LF^*)^*D \\ &\equiv \underline{e}_{ok}^{(m+1)}(LF^*)^*D \qquad \text{(using (ii))} \\ &\equiv \underline{e}_{ok}^{(m+1)}(LF^*(LF^*)^*D + D) \\ &\equiv \underline{e}_{ok}^{(m+1)}(LF^*V + D) \end{aligned}$$

$$\equiv \underline{e}_{ok}^{(m+1)}(PV + D) \qquad \text{(using (i))}.$$

Thus, using (iii),

$$W \equiv QW + E.$$

It remains to check that QW+E is non-redundant. This is straightforward except to observe that $p_{m+1,m+1}$ is non-redundant. This is because i and j are similar and MU+C is non-redundant. Thus $m_{ij}u_j + m_{ii}u_i \equiv m_{ij}u_j + m_{ii}u_j \equiv (m_{ij} + m_{ii})u_j$ is non-redundant.

At last we have all the bits and pieces needed to describe the process of constructing a deterministic machine characterising the simple, non-redundant expression R.

Firstly, by lemma 4, we can construct a constant + linear equational characterisation Y_1, A_1, B_1 of R. Using lemma 5 this can be transformed into a linear equational characterisation Y_2, A_2, B_2. Suppose A_2 is of order nxn. Then with each j, $1 \leq j \leq n$, we define the set associated with node j to be {j}. Now apply the following process starting with i = 0.

Suppose U,M,C is the linear equational characterisation of R constructed so far. Suppose m is of order mxm. Increment i by 1 and if i > m stop. Otherwise apply procedure N to U,M,C for each $a \in \Sigma$. Now each application of procedure N may introduce an (m+1)th node associated with some set S, say. If a node j ($1<j\leq m$) exists which is already associated with S then nodes m+1 and j are similar and can be coalesced using procedure C (renumbering the nodes appropriately). Otherwise m is incremented by 1.

It is well-known that this "subset method" will terminate and constructs a deterministic machine recognising R. That is, it constructs a linear matrix A and a constant vector B such that

$$\sigma(R) = \sigma(\underline{e}_{10})\sigma(A)^*\sigma(B)$$

and $\sigma(a_{ij}) \cap \sigma(a_{ik}) = \emptyset$ for all i, and all j, k s.t. j.≠ k. Thus we have proved that the subset method has the stronger property of constructing an equational characterisation Y,A,B of R. □

We now need to discuss the construction of a reduced deterministic machine recognising σ(R).

Suppose U,M,C is an equational characterisation of R where M is a deterministic mxm matrix. Because M is deterministic we can define $m_i(t)$, for each i ($1 \le i \le m$) and each $t \varepsilon \Sigma$, by $m_i(t) = j$ if and only if $t \varepsilon \sigma(m_{ij})$. If no such j exists then $m_i(t)$ is defined to be m+1.

Suppose $1 \le i,j \le m$. Then m+1 is said to be <u>distinguishable</u> from i. Also i and j are said to be <u>distinguishable</u> if $c_i \neq c_j$ or $\exists\, t \varepsilon \Sigma$ such that $m_i(t)$ is distinguishable from $m_j(t)$. Now, indistinguishability defines an equivalence relation on {1..m} which can be used to reduce M. Specifically, suppose [i] denotesthe equivalence class containing i and suppose there are n such classes. Number the classes from 1 to n and let s(i) denote the number assigned to [i]. Without loss of generality, assume s(1) = 1. Define the non-redundant, nxn matrix A as follows:

Let p and q be integers in the range 1..n. Suppose $p = s(i)$ and $q = s(j)$. Then a_{pq} is the sum of elements $t \varepsilon \Sigma$ such that $t \varepsilon \sigma(m_{ij})$. (Note: $t \varepsilon \sigma(m_{ij}) \Rightarrow t \varepsilon \sigma(m_{k\ell})$ for all $k \varepsilon [i]$ and $\ell \varepsilon [j]$. Thus A is well-defined.) A is called the reduced form of M.

Lemma 7 A is definite.

Proof Let us define $a_p(t)$ $(1 \le p \le n, t \varepsilon \Sigma)$ analogously to $m_i(t)$: $a_p(t) = q$ if $t \varepsilon \sigma(a_{pq})$. If no such q exists then $a_p(t)$ is defined to be n+1. Note that

$$a_{s(i)}(t) = s(m_i(t)) \qquad (1)$$

where s(n+1) is defined to be n+1.

Now, suppose $Y = [y_1, \ldots, y_n]' \equiv A \cdot Y$.

Let $y_{n+1} = \emptyset$. Then

$$y_p \equiv \sum_{t \varepsilon \Sigma} t \cdot y_{a_p(t)} \quad (1 \le p \le n) \qquad (2)$$

From Y we can construct an mx1 vector W by defining

$$w_i = y_{s(i)}$$

Let us also define $w_{m+1} = \emptyset$.

Suppose $p = s(i)$.

Then, $w_i \equiv y_p$

$$\equiv \sum_{t \varepsilon \Sigma} t \cdot y_{a_p(t)} \equiv \sum_{t \varepsilon \Sigma} t \cdot y_{s(m_i(t))} \quad \text{(by (1))}$$

$$\equiv \sum_{t \varepsilon \Sigma} t \cdot w_{m_i(t)}$$

In other words, $W \equiv M.W$.

But M is definite, hence $W \equiv \emptyset^{(m)}$. Consequently, $Y \equiv \emptyset^{(n)}$ and A is definite. □

Analogously to the reduction of M, let us reduce the mx1 vector C to the nx1 vector D. Specifically, we define $d_{s(i)} = c_i$ and refer to D as the reduced form of C.

Lemma 8 i and j are distinguishable if and only if $u_i \not\equiv u_j$.

Proof It is easy to prove (and, indeed, well-known) that if i and j are distinguishable then $\sigma(u_i) \neq \sigma(u_j)$ and hence $u_i \not\equiv u_j$. What we have to prove is the converse - i and j indistinguishable $\Rightarrow u_i \equiv u_j$.

Consider the reduced form A and D of M and C.

Suppose $X = [x_1,\ldots,x_n]'$ is a vector of reals and suppose

$$X = r(A)X + r(D) \qquad (1)$$

Such a vector exists and is unique by the definiteness of A. Define the mx1 vector $Y = [y_1,\ldots,y_m]'$ of reals by $y_i = x_{s(i)}$. It is straightforward to verify that

$$Y = r(M)Y + r(C) \qquad (2)$$

(The proof parallels the verification of the definiteness of A.) But M is definite and hence Y is the unique solution to equation (2).

Hence $Y = r(U)$

since $r(U) = r(M)r(U)+r(C)$.

Thus i is indistinguishable from j $\Rightarrow r(u_i) = r(u_j)$ (3)

Similarly, we may define the vector $P = [p_1,\ldots,p_n]'$ of subsets of Σ^* to be the unique solution of the equation

$$P = \sigma(A)P+\sigma(D)$$

and extend P to a solution Q of

$$Q = \sigma(M)Q+\sigma(C)$$

Whence $Q = \sigma(U)$, and

$$i \text{ is indistinguishable from } j \Rightarrow \sigma(u_i) = \sigma(u_j) \quad (4)$$

Combining (3) and (4) we have proved our lemma. □

<u>Corollary</u> Define the reduced form $V = [v_1,\ldots,v_n]'$ of U by $v_{s(i)} = u_i$. Then V,A,D is an equational characterisation of R.

<u>Theorem</u>

Suppose Q and R are non-redundant regular expressions and r(Q) and r(R) are both defined. Then $Q \equiv R$ if and only if $\sigma(Q) = \sigma(R)$.

<u>Proof</u> We have already observed that a non-redundant regular expression R can always be transformed into a simple, non-redundant regular expression P such that $R \equiv P$. So, without loss of generality, we may assume that R and Q are simple. Also, by definition, $Q \equiv R \Rightarrow \sigma(Q) = \sigma(R)$, so we only need prove the converse.

Now, $\sigma(Q) = \emptyset$ if and only if Q is $\emptyset$ (since it's simple). Thus we only need to prove that

$$R \neq \emptyset \text{ and } Q \neq \emptyset \text{ and } \sigma(Q) = \sigma(R) \Rightarrow Q \equiv R.$$

Suppose $R \neq \emptyset$, $Q \neq \emptyset$ and $\sigma(Q) = \sigma(R)$.

Then by lemma 6 there are equational characterisations U,M,C and V,N,D of R and Q (respectively) where M is a deterministic mxm matrix (for some m) and N is a deterministic nxn matrix (for some n). Construct the reduced forms A,E and B,F of M,C and N,D respectively. Then it is well-known that

$\sigma(Q) = \sigma(R) \Rightarrow \sigma(A) = \sigma(B)$ and $\sigma(E) = \sigma(F)$ (after possibly some renaming of nodes). But since A,B,E and F are non-redundant, $r(A) = r(B)$ and $r(E) = r(F)$. Consequently, $r(R) = r(Q)$ and so, trivially, $Q \equiv R$. □

References:

Aho, A.V., Hopcroft, J.D. and Ullman, J.E. [1974]
"The Design and Analysis of Computer Algorithms"
Addison-Wesley, Reading, Mass.

Backhouse, R.C. [1979]
"Syntax of Programming Languages: Theory and Practice"
Prentice-Hall International: London.

Backhouse, R.C. and Carré, B.A. [1975]
"Regular algebra applied to path-finding problems"
J.Inst.Maths.Applics. 15, 161-186.

Carré, B.A. [1971]
"An algebra for network routing problems"
J.Inst.Maths.Applics. 7, 273-294.

Carre, B.A. [1979]
"Graphs and Networks"
Oxford Applied Mathematics and Computing Science
Series: Oxford.

Duff, I.S. [1977]
"A survey of sparse matrix research"
Proc. IEEE 65, 500-535.

Ginzburg, A. [1967]
"A procedure for checking equality of regular expressions"
J.ACM 14, 355-362.

Lehman, D.J. [1977]
"Algebraic structures for transitive closure"
Theoretical Comp. Sci. 4, 59-76.

Tarjan, R.E. [1979]
"A unified approach to path problems"
Tech.Rep. STAN-CS-79-729, Computer Science Department,
Stanford University, (April, 1979).