REACHABILITY DETERMINATION
FOR A NON-SYNTACTIC SUBCLASS
OF VECTOR REPLACEMENT SYSTEMS (PETRI NETS)

by

C. M. Laucht

A thesis
presented to the University of Manitoba
in partial fulfillment of the
requirements for the degree of
DOCTOR OF PHILOSOPHY
in
Computer Science

Winnipeg, Manitoba, 1982

REACHABILITY DETERMINATION

FOR A NON-SYNTACTIC SUBCLASS

OF VECTOR REPLACEMENT SYSTEMS (PETRI NETS)

BY

C.M. LAUCHT

A thesis submitted to the Faculty of Graduate Studies of the University of Manitoba in partial fulfillment of the requirements of the degree of

DOCTOR OF PHILOSOPHY

ABSTRACT

This thesis introduces a new, non-syntactic subclass of Vector Replacement Systems and Petri Nets called the RT-subclass. This subclass is characterized as containing precisely those Vector Replacement Systems or Petri Nets for which a Reachability Tree Construction Algorithm, also introduced here, permits determination of Reachability.

In order to develop this algorithm, a notation called "Cube Notation" is introduced which permits folding certain infinite sets of states, such as $\{ \langle 0, 1, 2 \rangle, \langle 0, 1, 3 \rangle, \langle 0, 1, 4 \rangle, \ldots \}$ into finite sets of cubes, such as $\langle 0, 1, 2^{+} \rangle$. Various properties of this notation are then developed.

By using this Cube Notation and by modifying Karp and Miller's Coverability Tree Construction Algorithm, a Reachability Tree Construction Algorithm is developed.

It is then shown that, for those Vector Replacement Systems for which this algorithm terminates normally (i.e. those in the RT-subclass), not only can the decidability of reachability be determined, but that the Reachability Tree so created is equal to the Reachability Set.

The Reachability Tree construction algorithm is then compared to Karp and Miller's Coverability Tree construction algorithm and the modifications introduced to permit Reachability determination are discussed.

Some of the common syntactic subclasses, Marked Graphs, Conflict Free Nets, Free Choice Nets, State Machines, and non-syntactic subclasses, Bounded Nets, Safe Nets and Persistent Nets, are described briefly.

As well several new syntactic subclasses are introduced, namely Extended State Machines, Further Extended State Machines and Extended Marked Graphs.

Of these, State Machines, Extended State Machines, Further Extended State Machines and Bounded Nets are found to be proper subsets of the RT-subclass. The others are found to be incomparable.

Lastly, several areas suitable for further examination are outlined.

DEDICATION

This thesis is dedicated to my wife,

Marlene E. Pauls Laucht,

whose patience, understanding and encouragement

enabled me to finish it.

ACKNOWLEDGEMENT

I would like to express my sincere gratitude to Dr. M.S. Doyle, my supervisor, for his valuable advice and encouragement, to Drs. J.C. Muzio and J.L. Peterson for their help and continued interest, and to Dr. M. Doob for his constructive criticism and suggestions.

NOTATION

The following notation will be used throughout this thesis:

SYMBOL	MEANING
=>	implication (if then)
<=>	iff (if and only if)
$\in$	is an element of
N	the non-negative integers i.e. 0,1,2,3,4,5,...
Z	the integers i.e. ...,-2,-1,0,1,2,...
w	lower case omega ω
U	set union
SUB	is a subset, or a subcube, of
~SUB	is not a subset, or not a subcube, of
i = 1, k, 1	all i for i going from 1 to k in steps of 1
i = k, 1, -1	all i for i going from k to 1 in steps of -1
SUM(i=1,k,1)	summation over the range specified

CONTENTS

Chapter I

INTRODUCTION

1.1 INTRODUCTION

Vector Replacement Systems (and the equivalent graphic representation called Petri Nets) may be used to represent, model and analyze the behaviour of asynchronous concurrent systems.

Vector Replacement Systems were first described by Keller in [Keller 1972] and have since been shown equivalent to Generalized Petri Nets [Hack 1975] and Vector Addition Systems [Karp & Miller 1968] by Peterson and Bredt [Peterson & Bredt 1974] and Lipton et al [Lipton et al 1974].

Vector Replacement and related systems are useful for representing, modelling and analyzing asynchronous concurrent systems. They have been shown useful for:

Analysis of production schemata [Hack 1974]
Modelling an operating system [Noe 1971]
Describing digital systems [Patil & Dennis 1972]
Representing speed independent circuits [Misunas 1973]
Analysis of office information systems [Ellis & Nutt 1980]
Modelling of legal systems [Meldman & Holt 1971]
Representing man-machine interfaces [Meldman 1977]

to name just a few.

A good introduction to the subject of Petri Nets and related systems may be found in [Peterson 1977 and 1981] and in [Agerwala 1979].

In order to permit Vector Replacement System representation of a given system:

That system must have an initial state representable as a positive, integral n-tuple.

The possible actions of the system must be expressible as a finite number of "change of state" rules called transitions, which yield only new states which are also positive, integral n-tuples (n is fixed).

Associated with each transition must be a threshold and an output.

If at any time the current system state exceeds the threshold for some transition, then that transition is said to be enabled and any enabled transition may fire. It is the firing of a transition which causes a state change in the given system. The new state of the system after a transition firing depends upon the current state and the threshold and output of the firing transition. The simultaneous firing of two transitions is assumed to be impossible.

The set of all states which can be reached from the given initial state by such transition firing is called the Reachability Set. This set may have an infinite number of members.

The problem of interest here is the Reachability Problem for Vector Replacement Systems (and Petri Nets). Viz. Is some arbitrary state a member of the Reachability Set for a given Vector Replacement System (with a given initial state).

The decidability of the Reachability Problem in general has not yet been shown. [Sacerdote & Tenney 1977] appeared to prove that the problem was decidable. This proof was retracted later and a new, simplified proof was promised. It has not yet appeared.

The Reachability Problem has been shown decidable for several subclasses of Vector Replacement Systems and Petri Nets (or equivalent notations):

a) for Vector Addition Systems of dimension less than or equal to 3 in [Van Leeuwen 1974]

b) for Vector Addition Systems of dimension less than or equal to 5 in [Hopcroft & Pansiot 1976]

c) for Conflict Free Nets in [Crespi-Reghizzi & Mandrioli 1975].

d) for finite nets such as Bounded and Safe Nets and State Machines

e) for Marked Graphs as reported in [Peterson 1977].

This thesis will add another subclass, the RT-subclass, for which reachability can be determined.

The work presented here is based on the Coverability Tree construction algorithm of [Karp & Miller 1968] as described in [Hack 1975] and the Vector Replacement System Notation of [Keller 1972].

The Karp & Miller Coverability Tree (called a Reachability Tree in [Karp & Miller 1968]) does not contain enough information to permit determination of Reachability.

By using the Cube Notation developed here to increase the amount of information stored in the tree and by restricting the tree construction algorithm somewhat, a new algorithm is created. This algorithm, called the Reachability Tree construction algorithm, defines a new subclass of Vector Replacement Systems and Petri Nets, called the RT-subclass, for which Reachability can be determined by inspection of the Reachability Tree.

The RT-subclass of Vector Replacement Systems and Petri Nets so defined is called non-syntactic because membership in this subclass is described not in syntactic terms, but rather in terms of algorithm termination.

1.2 BRIEF DESCRIPTION OF THESIS

The three original results presented in this thesis are:

a) Cube Notation and its properties,

b) Reachability Tree Construction Algorithm, the proof that the Reachability Tree is equal to the Reachability Set, and the RT-subclass, and

c) Introduction of three other, new subclasses, namely Extended State Machines, Further Extended State Machines and Extended Marked Graphs.

Chapter I sets the stage by providing an overview including an introduction to Vector Replacement System notation, the equivalent Petri Net notation, and Reachability.

Chapter II provides an introduction to those properties of transition sequences which will be required for later proofs.

Chapters III, IV, and V introduce the Cube Notation and its properties.

Chapter VI presents the major result, the Reachability Tree Construction Algorithm and its proof. The Cube Notation was developed to enable this result to be formulated and proved.

Chapter VII provides not only a definition of the RT-subclass, a discussion of the result and a comparison of the RT-subclass with common subclasses, but also introduces the three new subclasses mentioned in c) above.

Chapter VIII summarizes the results and outlines possible further work.

1.3 VECTOR REPLACEMENT SYSTEMS

Vector Replacement Systems can be built up from the natural numbers. Each state is an n-tuple of natural numbers, where each component of the n-tuple is called a state-component.

1.3.1 States

DEFINITION 1.1 STATE COMPONENT and STATE

A state component, denoted q_j is a non-negative integer.

i.e. $q_j \in N$

A state q is an n-tuple of state components.

i.e. $q = < q_1, \dots, q_n >$

1.3.2 State Operations

The arithmetic operations of + and - and the relational operations $<$, $\leq$, $=$, $\neq$, $\geq$, $>$ for integers are extended componentwise to states and n-tuples.

DEFINITION 1.2 STATE +,- OPERATIONS

For q any state and x any n-tuple ($x_j \in N$)

$$q' = q \pm x = < q'_1, \dots, q'_n > \mid q'_j = q_j \pm x_j \text{ for } j = 1,\dots,n$$

Note that q' is a state only if all components are non-negative.

LEMMA 1.3

q any state and x an n-tuple such that all $x_j \in N$

$\Rightarrow$

$q + x$ is a state

LEMMA 1.4

For any state q and any two n-tuples x, y with all x_j, $y_j < N$ for $j = 1, \ldots, n$:

a) $(q + x) \pm y = q + (x \pm y)$

$= \langle q'_1, \ldots, q'_n \rangle \mid q'_j = q_j + x_j \pm y_j$ for $j = 1, \ldots, n$

b) $(q - x) \pm y = q - (x \mp y)$

$= \langle q'_1, \ldots, q'_n \rangle \mid q'_j = q_j - x_j \mp y_j$ for $j = 1, \ldots, n$

Proof:

This follows from associativity for integer addition and subtraction.

Note that q′ is a state only if all components are non-negative.

DEFINITION 1.5

For x and y, any two n-tuples, the relational operations $<$, $\leq$, $=$, $\neq$, $\geq$, $>$ are extended as follows:

$x \leq y \iff x_j \leq y_j$ for $j = 1, \ldots, n$

$x = y \iff x_j = y_j$ for $j = 1, \ldots, n$

$x \geq y \iff x_j \geq y_j$ for $j = 1, \ldots, n$

$x \neq y \iff$ There exist at least one j such that $x_j \neq y_j$ for $j = 1, \ldots, n$

$x < y \iff x \leq y$ and $x \neq y$

$x > y \iff x \geq y$ and $x \neq y$

1.3.3 Transitions

Transitions represent events or actions in a system which result in a change from one state to another. When this happens a transition is said to fire.

Such state changes cannot occur at just any time but only when certain conditions have been fullfilled. In production systems for example, production cannot start until sufficient raw materials are present. Similarly, in Vector Replacement Systems, the condition for transition firing is that the current state must exceed a threshold or input vector associated with the transition. When this happens a transition is said to be enabled.

Only enabled transitions are capable of firing. Any enabled transition can fire independently of any other enabled transition except that the firing of one transition may alter conditions so that a previously enabled transition is no longer enabled.

When a transition fires the state change represented by D^t will take place.

DEFINITION 1.6 TRANSITION

A transition, denoted t, is a pair $[I^t, O^t]$ of n-tuples

$$[< I_1^t, \ldots, I_n^t > \text{ and } < O_1^t, \ldots, O_n^t >]$$

where t, is an index identifying the particular transition, such that $I_j^t \in N$ and $O_j^t \in N$ for $j = 1, \ldots, n$. I^t is called the input or threshold vector and O^t is called the output vector. Rather than continually writing $[I^t, O^t]$ every time reference is made to a transition, the index t will be used instead.

DEFINITION 1.7 D^t

D^t, called the change vector for t is defined as:

$$D^t = O^t - I^t$$

$$= \langle D_1^t, \ldots, D_n^t \rangle \text{ where } D_j^t = O_j^t - I_j^t \text{ for } j = 1, \ldots, n$$

It should be noted that $D_j^t \in Z$.

Thus a transition t can be uniquely specified by giving any two of I^t, D^t, or O^t. Typically I^t and D^t or I^t and O^t will be used.

DEFINITION 1.8 STATE ENABLED

A transition t is said to be state enabled in a state q if

$$q - I^t \geq 0$$

where $0 = \langle 0, \ldots, 0 \rangle$.

$$\text{i.e. } q_j - I_j^t \geq 0 \text{ for } 1 \leq j \leq n.$$

This is denoted $q \xrightarrow{t}$.

1.3.4 State Change

The concept of state change as a result of transition firing can now be formalized.

DEFINITION 1.9 TRANSITION FIRING

Given a state q and a transition $t \in T$ which is enabled in q, i.e. $q \xrightarrow{t}$, then q′ given by

$$q' = q - I^t + O^t = q + D^t$$

is said to be immediately state reachable from q by the firing of t.

This is denoted $q \xrightarrow{t} q'$.

LEMMA 1.10

q′ as defined in Definition 1.9 is a state.

The state resulting from the firing of a particular transition for a particular given state is unique.

LEMMA 1.11 UNIQUENESS OF STATES

$$q \xrightarrow{t} q^1 \text{ and } q \xrightarrow{t} q^2$$

$$\Rightarrow$$

$$q^1 = q^2$$

DEFINITION 1.12 IMMEDIATE ANTECEDENT and IMMEDIATE SUCCESSOR OF A STATE

Where $q \xrightarrow{t} q'$, q is said to be an immediate antecedent or an immediate predecessor of q′, and q′ is said to be an immediate successor of q.

1.3.5 Transition Sequences

In order to more easily discuss a series of consecutive transition firings, the concept of a transition sequence is introduced.

DEFINITION 1.13 TRANSITION SEQUENCE

A *transition sequence* s, is an ordered sequence of transitions or a concatenation of transition sequences (called subsequences).

It should be noted that the transitions in s need not be unique.

For example:

$t^4 t^1 t^2 t^2$

t^1

$t^3 t^3 t^3$

are transition sequences.

It is often convenient to refer to transitions in terms of their relative positions within a transition sequence. Thus i represents the relative position of t^{b^i} in $t^{b^1} \ldots t^{b^k}$ for k = transition sequence length and $i = 1, \ldots, k$.

For example:

a) $s = t^{b^1}, \ldots, t^{b^k}$

b) $s^1 = t^1, \ldots, t^{b^i}$

c) $s^2 = t^{b^{i+1}}, \ldots, t^{b^k}$

d) $s = s^1 s^2 = t^{b^1}, \ldots, t^{b^i}, t^{b^{i+1}}, \ldots, t^{b^k}$

Furthermore, when working with a transition sequence it is often convenient to simplify this labelling. When it can be done without introducing ambiguity the transition t^{b^i} will be relabelled as t^i.

Thus s, s^1 and s^2 as above become:

a) $s = t^1, \ldots, t^k$

b) $s^1 = t^1, \ldots, t^i$

c) $s^2 = t^{i+1}, \ldots, t^k$

d) $s = s^1s^2 = t^1, \ldots, t^i, t^{i+1}, \ldots, t^k$

A further notation used for transition subsequences is

s(start index, stop index)

For example, for s, s^1 and s^2 as above:

a) $s = s(1,k) = t^1, \ldots, t^k$

b) $s^1 = s(1,i) = t^1, \ldots, t^i$

c) $s^2 = s(i+1,k) = t^{i+1}, \ldots, t^k$

d) $s = s^1s^2 = s(1,i)s(i+1,k) = t^1, \ldots, t^i, t^{i+1}, \ldots, t^k$

A replication factor, R, for a given transition sequence is also sometimes useful.

eg. $5s = sssss$ $\quad$ $2(t^1t^2) = t^1t^2t^1t^2$

$3t = ttt$ $\quad$ $0s$ = null sequence

DEFINITION 1.14 $q^0 \xrightarrow{s} q^k$

For convenience:

$$q^0 \xrightarrow{t^1} q^1, \quad q^1 \xrightarrow{t^2} q^2, \quad \ldots, \quad q^{k-1} \xrightarrow{t^k} q^k$$

is often denoted

$$q^0 \xrightarrow{t^1} q^1 \xrightarrow{t^2} q^2 \xrightarrow{t^3} \ldots q^{k-1} \xrightarrow{t^k} q^k$$

and, if $s = t^1, \ldots, t^k$, it is also denoted:

$$q^0 \xrightarrow{s} q^k.$$

1.3.6 Vector Replacement Systems

It is now possible to give a formal définition of a Vector Replacement System.

DEFINITION 1.15 VECTOR REPLACEMENT SYSTEMS

A vector replacement system or VRS is a pair [T, q^{init}] where T is a set of transitions t and q^{init} is a state called the initial state.

EXAMPLE 1.16

A simple, yet illustrative example of an asynchronous concurrent system is the traditional producer/consumer example.

This system has essentially three distinct parts, a producer, a transmission mechanism (or buffer) and a consumer.

The producer will be restricted to producing one unit of its product at a time. Once produced it will be ready to ship it. Sometime thereafter it will be shipped and then the producer will be able to produce another unit.

The producer can thus be in one of two states: "ready to produce a unit" or "ready to ship a unit". The change from one state to another is accomplished by firing the transitions "produce one unit" and "ship one unit".

The consumer will similarily be restricted to receiving one unit of its product at a time. Once received it will be ready to consume it. Sometime thereafter it will be consumed and then the consumer will be ready to receive another unit.

The consumer thus has two states "ready to receive a unit" and "ready to consume a unit". The change from one state to another is accomplished by firing transitions "receive a unit" and "consume a unit".

The shipping or buffer mechanism can be in any one of an unbounded number of states, each representing the number of units (≥ 0) currently in transit. A change of state is effected by firing transitions "ship a unit" and "receive a unit".

The initial state of this system will consist of these substates (called state components in a Vector Replacement System):

Producer:	ready to produce
	not ready to ship
Consumer:	ready to receive
	not ready to consume
Buffer:	zero units in transit

This system may be represented as a Vector Replacement System where each state-component q_j, $j: 1 \leq j \leq 5$ is interpreted as follows:

	State	State Description
Producer:	$q_1 = 0$	not ready to produce a unit
	$= 1$	ready to produce a unit
	$q_2 = 0$	not ready to ship a unit
	$= 1$	ready to ship a unit
Buffer:	$q_3 = n$	n units in transit $(N \geq 0)$
Consumer:	$q_4 = 0$	not ready to receive a unit
	$= 1$	ready to receive a unit
	$q_5 = 0$	not ready to consume a unit
	$= 1$	ready to consume a unit

The initial state as described can then be represented by

$$< q_1, q_2, q_3, q_4, q_5 > = < 1, 0, 0, 1, 0 >$$

Each transition can be interpreted as:

Transition	Transition Description
t^1	produce a unit
t^2	ship a unit
t^3	receive a unit
t^4	consume a unit

The conditions which must hold before a given transition can fire are:

Transition	Condition Description		I
t^1	ready to produce a unit	i.e. $q^1 = 1$	$\langle 1,0,0,0,0\rangle$
t^2	ready to ship a unit	i.e. $q^2 = 1$	$\langle 0,1,0,0,0\rangle$
t^3	at least one unit in transit	i.e. $q^3 \geq 1$	
	and ready to receive a unit	i.e. $q^4 = 1$	$\langle 0,0,1,1,0\rangle$
t^4	ready to consume a unit	i.e. $q^5 = 1$	$\langle 0,0,0,0,1\rangle$

The state change effected by each transition is then:

Transition	Change Description	D
t^1	ready to produce a unit	
	$\longrightarrow$ ready to ship a unit	$\langle -1,1,0,0,0\rangle$
t^2	ready to ship a unit	
	$\longrightarrow$ ready to produce a unit	
	and increase units in transit by 1	$\langle 1,-1,1,0,0\rangle$
t^3	ready to receive a unit and	
	units in transit ≥ 1	
	$\longrightarrow$ decrease units in transit by 1	
	and ready to consume a unit	$\langle 0,0,-1,-1,1\rangle$
t^4	ready to consume a unit	
	$\longrightarrow$ ready to receive a unit	$\langle 0,0,0,1,-1\rangle$

In summary, the Vector Replacement System representation for the producer/consumer example as presented is:

$$VRS = [T, q^{init}]$$

where

$$q^{init} = \langle 1, 0, 0, 1, 0 \rangle$$

and

$$T = \{ t^1\ t^2, t^3, t^4 \}$$

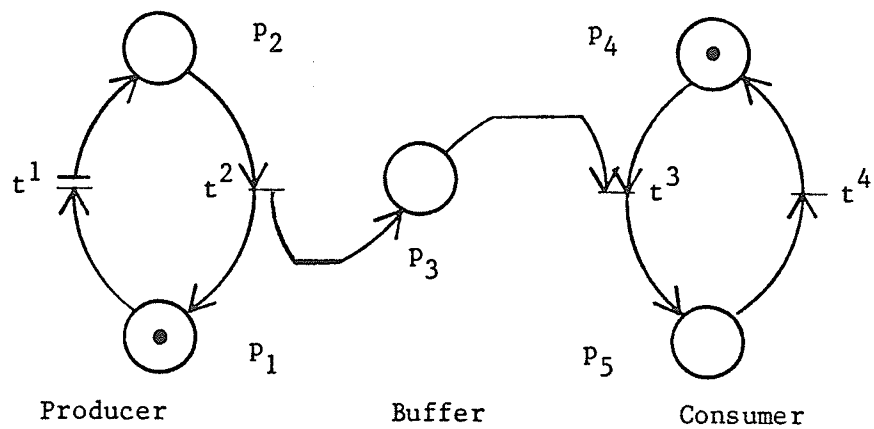
such that

t	I^t	D^t	O^t
1	$\langle 1, 0, 0, 0, 0 \rangle$	$\langle -1, 1, 0, 0, 0 \rangle$	$\langle 0, 1, 0, 0, 0 \rangle$
2	$\langle 0, 1, 0, 0, 0 \rangle$	$\langle 1, -1, 1, 0, 0 \rangle$	$\langle 1, 0, 1, 0, 0 \rangle$
3	$\langle 0, 0, 1, 1, 0 \rangle$	$\langle 0, 0, -1, -1, 1 \rangle$	$\langle 0, 0, 0, 0, 1 \rangle$
4	$\langle 0, 0, 0, 0, 1 \rangle$	$\langle 0, 0, 0, 1, -1 \rangle$	$\langle 0, 0, 0, 1, 0 \rangle$

1.4 PETRI NETS

It is difficult to visualize the original system description of the producer/consumer system in Example 1.16 when presented only with the Vector Replacement System description of it. Visualizing such systems is made considerably easier if the equivalent Petri Net notation is used as well.

It is possible to represent Example 1.16 by the following Petri Net:



where

The circles, called places, denoted p_1, ..., p_n, represent the state components q_1, ..., q_n and

the number of dots, called tokens, in each circle, represents the present value of the corresponding state component. When the number of tokens in a given state is too large to represent by dots, the actual number is inscribed in the circle or place.

the bars, not only represent, but are also called, transitions. Enabled transitions are indicated by a double bar eg. || or $=$.

the directed arcs represent the input and output vectors for each transition. The number of arcs leading from each place p_j, $j: 1 \leq j \leq n$, to a given transition t, represents I^t_j, where $I^t = \langle I^t_1, \ldots, I^t_n \rangle$ and the number of arcs leading from a given transition t to each place p_j, represents O^t_j where $O^t = \langle O^t_1, \ldots, O^t_n \rangle$.

A transition causes a state change, i.e. it is said to fire, only when the present state of the system (also called a marking) exceeds the input or threshold vector (i.e. when it is enabled). In the Petri Net notation a transition is enabled if each of its input places has at least one token in it for each arc leading from that input place to the transition.

Firing of a transition removes, from each place having one or more arcs leading from it to the firing transition, a number of tokens equal to the number of such arcs.

Similarly, firing of the transition also deposits, into each place having one or more arcs leading into it from the firing transition, a number of tokens equal to the number of such arcs.

After firing, the new token distributuion reflects the new system state.

EXAMPLE 1.17

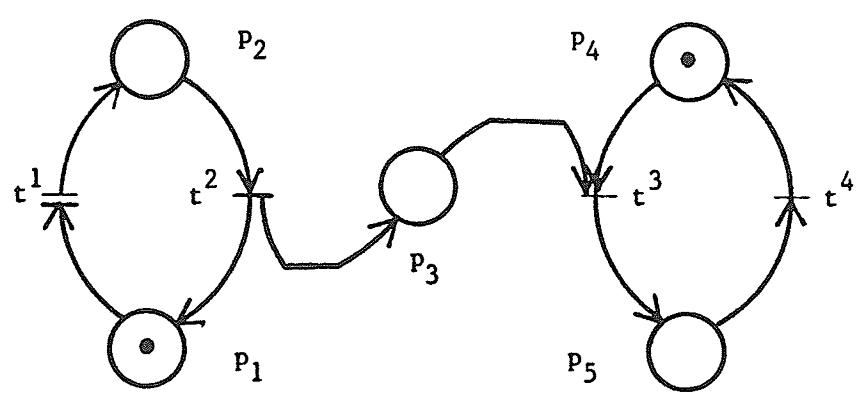
Given a Vector Replacement System as defined in Example 1.20, with an initial state of $\langle 1, 0, 0, 1, 0 \rangle$, the result of firing transition sequence

$$s = t^1 t^2 t^1 t^2 t^3$$

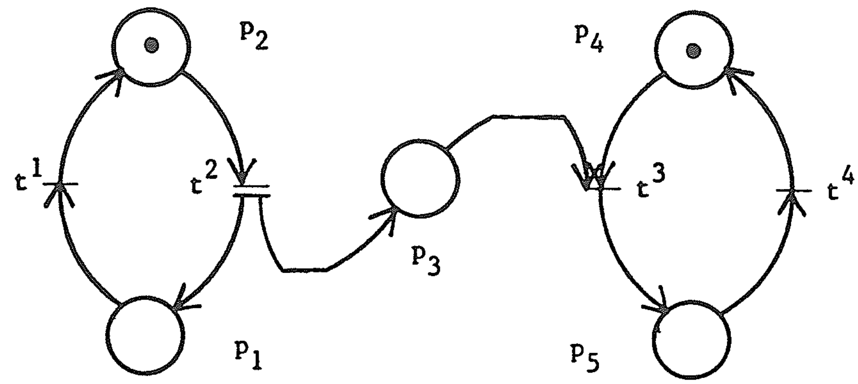
yields:

$$\langle 1, 0, 0, 1, 0 \rangle$$
$$\downarrow t^1$$
$$\langle 0, 1, 0, 1, 0 \rangle$$
$$\downarrow t^2$$
$$\langle 1, 0, 1, 1, 0 \rangle$$
$$\downarrow t^1$$
$$\langle 0, 1, 1, 1, 0 \rangle$$
$$\downarrow t^2$$
$$\langle 1, 0, 2, 1, 0 \rangle$$
$$\downarrow t^3$$
$$\langle 1, 0, 1, 0, 1 \rangle$$

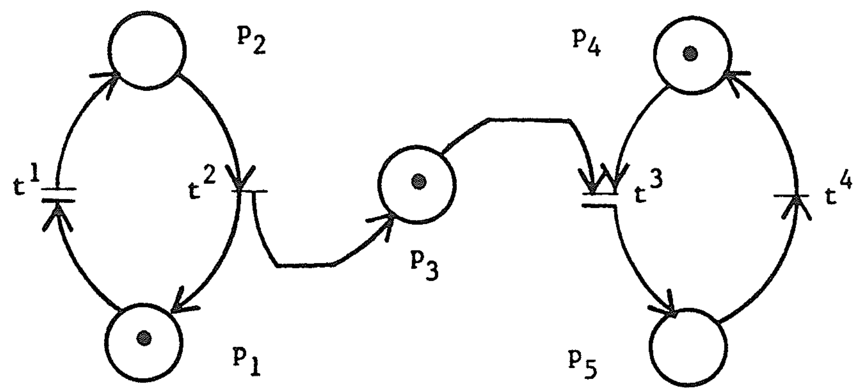
and would be represented by the following sequence of Petri Nets:



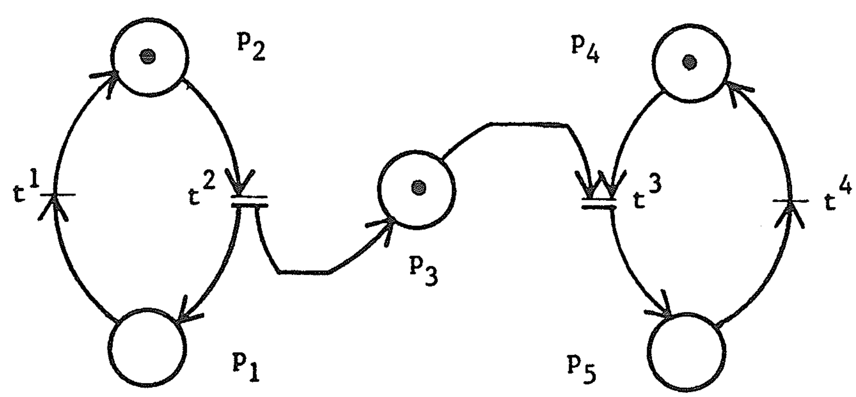
$$\downarrow t^1$$



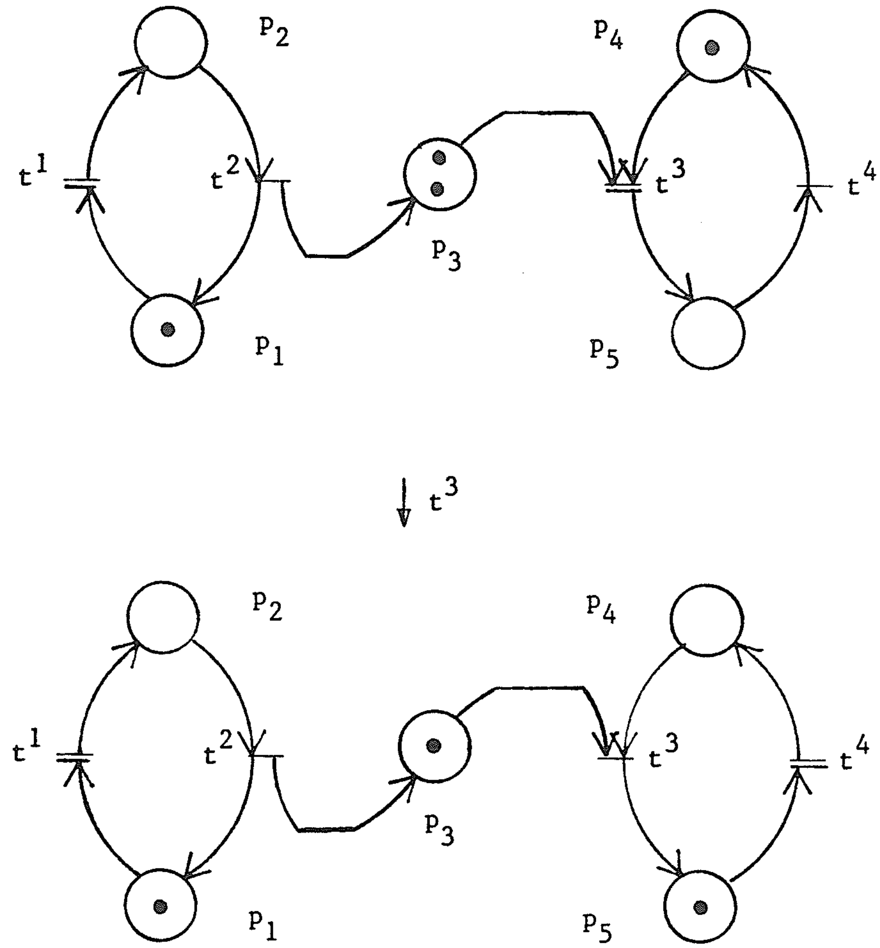
$\downarrow t^2$



$\downarrow t^1$

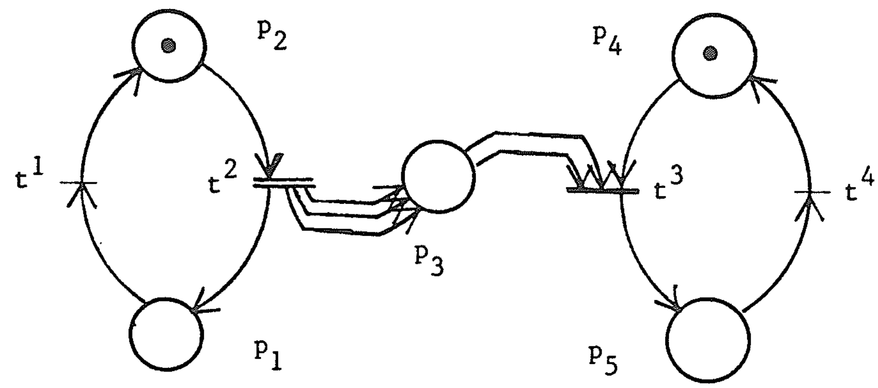


$\downarrow t^2$



It is worthy of note that different (integral) rates of production and/or consumption can be represented by different numbers of output arcs from t^2 and of input arcs to t^3, respectively.

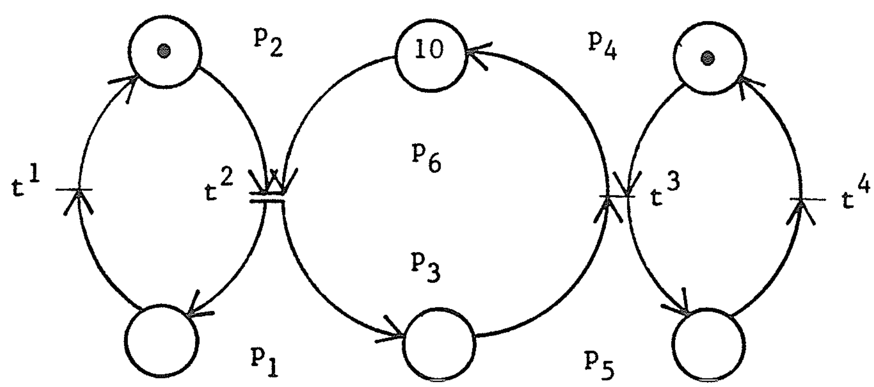
For example production of 3 units at a time and consumption of 2 units at a time would be represented by:



It should be noted that the 3 unit producer, 2 unit consumer shown above will not be in the RT-subclass because p_3 can become unbounded with an increment greater than 1. (i.e. It becomes unbounded and has 3 output arcs from the same transition.)

In addition, the concept of a bounded buffer can be accomodated.

If the bound of the buffer, represented by p_3, were to be 10 units, then adding place p_6 with 10 tokens yields:



Thus the number of tokens in p_3 and p_6 will always total 10.

If p_3 contains all 10 of the tokens then the transition t^2 cannot fire. This represents a full buffer and inhibited production. Consumption can of course occur.

If p_6 contains all 10 tokens, then transition t^3 cannot fire. This represents an empty buffer and inhibited consumption. Production can of course occur.

If both p_3 and p_6 contain tokens then either t^2 or t^3 can fire. This represents a partially full buffer. Production and consumption can occur.

1.5 REACHABILITY

Given a system represented as a Vector Replacement System, then a given state which can be reached from the initial state by repeated application of the change-of-state rules (i.e. the transitions) is said to be reachable from that initial state.

This concept is now formalized and generalized to include reachability of one state from any other state.

DEFINITION 1.18 STATE REACHABILITY

q^k is said to be state reachable from q if and only if there exists some transition sequence s such that

$$q \xrightarrow{s} q^k.$$

i.e.

A state q^k is said to be state reachable from another state q if

a) q^k is immediately state reachable from q or

b) q^k is immediately state reachable from a state q^{k-1} which is state reachable from q.

State reachability will be called reachability unless the context requires the full form.

1.6 REACHABILITY SET

The set of all states reachable by transition firing from some given initial state for a given Vector Replacement System is called the Reachability Set for that Vector Replacement System. This set can be infinite in size, as will be shown in two upcoming examples.

DEFINITION 1.19 REACHABILITY SET

The Reachability Set (denoted RS_{VRS} or just RS) for a vector replacement system, VRS = $[T,q^{init}]$, is the set of those states which are state reachable from the initial state q^{init}. For $q^0 = q^{init}$, the reachability set is defined recursively as follows:

BASE: $q^0 \in RS_{VRS}$

STEP: if $q^{k-1} \in RS_{VRS}$ and $q^{k-1} \xrightarrow{t} q^k$

then q^k is also $\in RS_{VRS}$.

That is the initial state is an element of the reachability set and, recursively, if a state q^{k-1} is in the reachability set, and is the immediate antecedent of another state q^k, then q^k is also in the reachability set.

It should be noted that the Reachability Set depends upon the Transition Set T and the initial state q^{init}. Neither is sufficient to specify a Reachability Set and a change in either can result in a completely different Reachability Set being specified.

The necessary and sufficient conditions for a state to be a member of a given Reachability Set are given in the following lemma.

<u>LEMMA</u> 1.20

For $VRS = [T, q^{init}]$,

$$q \in RS_{VRS}$$

$$\Longleftrightarrow$$

$$\text{There exists some transition sequence } s = t^1 \ldots t^k$$

$$\text{where } t^i \in T \text{ for } i = 1, \ldots, k$$

$$\text{and}$$

$$q^{init} \xrightarrow{s} q$$

If a state q′ is state reachable from a state q by transition sequence s and if q is in the reachability set, then q′ is in the reachability set as well. This is shown in the following lemma.

<u>LEMMA</u> 1.21

$$q \xrightarrow{s} q' \text{ and } q \in RS_{VRS}$$

$$\text{where } VRS = [T, q^{init}]$$

$$\Longrightarrow$$

$$q' \in RS_{VRS}$$

Two examples follow.

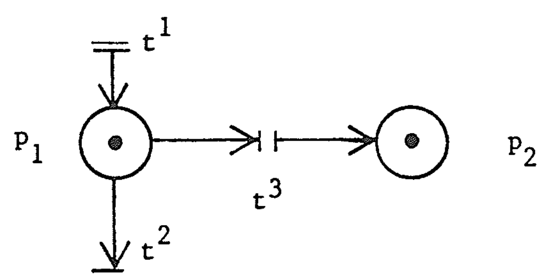
EXAMPLE 1.22

For a Vector Replacement System, VRS = [T, q^{init}] where:

$$q^{init} = \langle 1, 1 \rangle \text{ and}$$

Transition	Threshold Vector	Output Vector
t^1	$I^1 = \langle 0,0 \rangle$	$0^1 = \langle 1,0 \rangle$
t^2	$I^2 = \langle 1,0 \rangle$	$0^2 = \langle 0,0 \rangle$
t^3	$I^3 = \langle 1,0 \rangle$	$0^3 = \langle 0,1 \rangle$

the corresponding Petri Net is:



where

the places, denoted p_1 and p_2, represent the state components q_1 and q_2 and

the number of tokens, in each place, represents the present value of the corresponding state component.

the transitions t^1, t^2 and t^3 are those of the VRS.

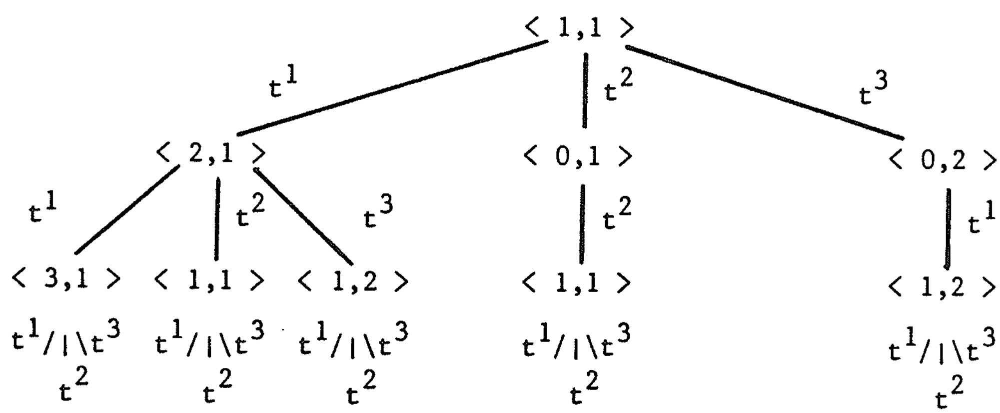
the number of arcs leading from each place p_j, $j: 1 \leq j \leq 2$, to a given transition t, represents the input or threshold vector component I_j^t, where $I^t = \langle I_1^t, \ldots, I_n^t \rangle$

and the number of arcs leading from a given transition t to each place p_j, represents the output vector component O_j^t where $O^t = <O_1^t, \ldots, O_n^t>$.

For example, transition t^3 has one input arc from p_1 and none from p_2. This is represented by $I^3 = <1,0>$. It has one output arc going to p_2. This is represented by $O^3 = <0,1>$.

The reachability set can then be represented as an infinite tree, where the initial state is the root node and each state reachable from it appears on some branch of the tree. The arcs coming out of a node represent enabled transitions for the state in the node and the successor nodes represent the successor states via the firing of these transitions.

The first three levels for this example are shown below:



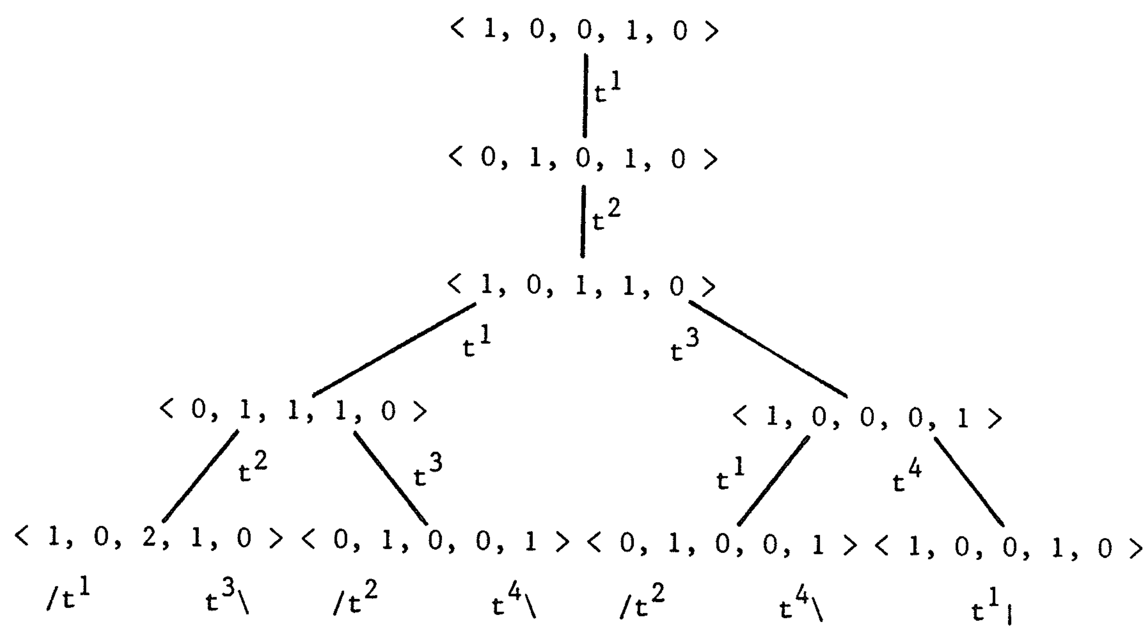
That the tree is indeed infinite can be inferred from the leftmost branch of the tree. Transition t^1 can fire indefinitely, thereby increasing the first component indefinitely.

In fact, this example is simple enough that the reachability set can be seen to be:

$$RS = \{ \langle q^1, q^2 \rangle \mid q^1 \geq 0, q^2 \geq 1 \}.$$

EXAMPLE 1.23

The reachability set for the producer/consumer example can also be expressed as an infinite tree. The first five levels are shown:



Here too the left hand branch can be used to show that this tree is infinite. Repeating transition sequence t^1t^2 indefinitely will cause the 3^{rd} component to increase indefinitely.

Chapter II

TRANSITION SEQUENCES

The properties of transitions with respect to states are now generalized to corresponding analogous properties of transition sequences.

2.1 STATE ENABLING

The concept of state enabling is now extended to transition sequences.

DEFINITION 2.1 STATE ENABLING FOR TRANSITION SEQUENCES

A transition sequence $s = t^1, \ldots, t^k$ is said to be state enabled for a state q^0, denoted $q^0 \xrightarrow{s}$, if there exists some $q^1, \ldots, q^k$ such that $q^0 \xrightarrow{s} q^k$.

The following lemma shows that Definition 2.1 is consistent with Definition 1.8 to the extent that if a state is enabled for some transition sequence s, then it is also enabled for the first transition in s.

LEMMA 2.2

For $s = t^1, \ldots, t^k$,

$$q^0 \xrightarrow{s} \Rightarrow q^0 \xrightarrow{t^1}$$

2.2 STATE CHANGE

The following definition and lemma extend the concept of a change vector, first introduced in Definition 1.9, to transition sequences.

The change vector D^s for a transition sequence s will be defined as the sum of the change vectors for the constituent transitions.

DEFINITION 2.3 CHANGE VECTOR FOR A TRANSITION SEQUENCE

The change vector D^s for a sequence of transitions $s = t^1, \ldots, t^k$ and $q^0 \xrightarrow{s} q^k$ is

$$D^s = D^{s(1,k)} = \mathrm{SUM}(i=1,k,1)\ D^{t^i}$$

It will now be shown that this definition indeed yields a change vector which equals the difference between the final and initial states.

LEMMA 2.4

$$q^0 \xrightarrow{t^1} q^1 \xrightarrow{t^2} q^2 \xrightarrow{t^3} \ldots q^{k-1} \xrightarrow{t^k} q^k$$

$$\Rightarrow$$

$$q^k = q^0 + \mathrm{SUM}(i=1,k,1) \text{ of } (O^{t^i} - I^{t^i})$$

$$\text{i.e. } q^k = q^0 + D^s \text{ or } D^s = q^k - q^0$$

Proof:

This may be shown using repeated application of Definition 1.9.

2.3 MAX OPERATOR

The introduction of the familiar max operator will simplify later notation.

DEFINITION 2.5 MAX

For $x_j, y_j \in N$,

$$\max(x_j, y_j) = x_j \text{ if } x_j \geq y_j$$
$$= y_j \text{ if } x_j < y_j$$

Again for convenience, the following two lemmata formalize well known properties of the max operator.

LEMMA 2.6

$$x_j,\ y_j,\ z_j \in Z \text{ or } \in N$$
$$\Rightarrow$$
$$\max(x_j,\ \max(y_j,\ z_j)) = \max(y_j,\ \max(x_j,\ z_j))$$
$$= \max(z_j,\ \max(x_j,\ y_j))$$

It is readily apparent that all three expressions simply serve to select the largest element of x_j, y_j, z_j. As a convenience, all three expressions will be represented by $\max(x_j, y_j, z_j)$.

LEMMA 2.7

For $x_j, y_j \in N$

$$\max(x_j,y_j) \geq y_j \text{ (or equivalently } \max(x_j,y_j) - y_j \geq 0 \text{)}$$

and

$$\max(x_j,y_j) \geq x_j \text{ (or equivalently } \max(x_j,y_j) - x_j \geq 0 \text{)}$$

2.4 INPUT VECTOR

The input or threshold vector I^s for a transition sequence will be defined in this section.

The complex and seemingly arbitrary definition for I^s will then be shown to yield necessary and sufficient conditions for a transition sequence to be state enabled. These conditions will be consistent with Definition 1.8. In particular

$$q - I^s \geq 0 \iff q \xrightarrow{s}$$

will be shown.

DEFINITION 2.8 I^s

Given transition sequence $s = t^1, \ldots, t^k$, the input or threshold vector, I^s, also denoted $I^{s(1,k)}$, is defined iteratively for i going from k to 1 in steps of -1 as follows:

BASE: $I^{s(k,k)} = I^{t^k}$

where I^{t^k} is the input vector for transition t^k.

STEP: $I^{s(i,k)} = \left\langle \max(I_j^{t^i}, I_j^{s(i+1,k)} - O_j^{t^i} + I_j^{t^i}) \text{ for } j = 1, \ldots, n \right\rangle$

for i going from k-1 to 1.

For a transition sequence of length 1, i.e. consisting of a single transition, this definition should yield I^t. That this is so is shown next.

LEMMA 2.9

$$s = t \implies I^s = I^t$$

Proof:

This follows directly from Definition 2.8 by substituting k = 1.

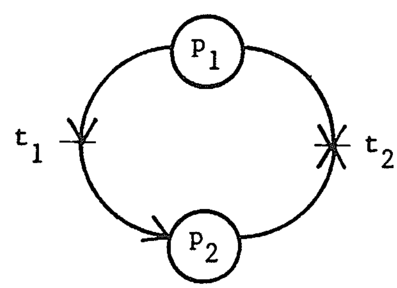
An example should help to clarify the notation being used.

EXAMPLE 2.10

For a VRS = $[T, q^{init}]$ such that

	Transition	
	1	2
I	< 1, 0>	< 1, 1>
O	< 0, 1>	< 0, 0>
D	<-1, 1>	<-1,-1>

the Petri Net equivalent is:



Letting $s^1 = t^1$, $s^2 = t^2$ and $s = t^1t^2$ means that $s^1 = s(1,1)$, $s^2 = s(2,2)$ and $s = s(1,2)$ in terms of the subsequence notation. Applying the backwards iteration of Definition 2.8 to each of the transition sequences s, s^1 and s^2, yields

$$I^s = I^{s(1,2)} = \langle 2, 0\rangle$$
$$I^{s^1} = I^{s(1,1)} = \langle 1, 0\rangle \quad (= I^{t^1} \text{ by Lemma 2.9})$$
$$I^{s^2} = I^{s(2,2)} = \langle 1, 1\rangle \quad (= I^{t^2} \text{ by Lemma 2.9})$$

The following two lemmata and the following corollary, while not required for the main result of this section, are presented here because they follow directly from Definition 2.8 and will be needed later.

LEMMA 2.11

$$s = t^1, \ldots, t^k$$
$$\Rightarrow$$
$$I_j^{s(i,k)} \geq I_j^{t^i} \text{ for } j = 1, \ldots, n$$

Proof:

This follows directly from Definition 2.8. It should be noted that equality cannot, in general, be shown.

LEMMA 2.12

$$s = t^1, \ldots, t^k$$
$$\Rightarrow$$
$$I_j^{s(i,k)} \geq 0 \text{ for } j = 1, \ldots, n$$

Proof:

Definition 1.6

$$\Rightarrow I_j^{t^i} \geq 0$$

Lemma 2.11

$$\Rightarrow I_j^{s(i,k)} \geq I_j^{t^i}$$
$$\Rightarrow I_j^{s(i,k)} \geq 0$$

COROLLARY 2.13

$$s = t^1, \ldots, t^k$$
$$\Rightarrow$$
$$I^s \geq 0$$

The next two lemmata prove the necessary and sufficient conditions of the major result of this section.

LEMMA 2.14

$$q^0 \xrightarrow{s} q^k$$
$$\Rightarrow$$
$$q^0 - I^s \geq 0$$

Proof:

$q^0 \xrightarrow{s} q^k$ and Definition 1.14

$$\Rightarrow q^0 \xrightarrow{t^1} q^1 \xrightarrow{t^2} q^2 \xrightarrow{t^3} \ldots q^{k-1} \xrightarrow{t^k} q^k$$

Definition 1.9

$$\Rightarrow \text{each } q^i = q^{i-1} - I^{t^i} + O^{t^i}$$

The remainder of the proof will be by induction upon the length of the transition subsequence, $s(i,k)$ for $i = k,1,-1$. It will be shown that at each stage $q^{i-1} - I^{s(i,k)} \geq 0$. Hence $q^0 - I^{s(1,k)} \geq 0$ and, since $I^{s(1,k)}$ is just an alternative notation for I^s, therefore $q^0 - I^s \geq 0$.

BASE

Definition 2.8

$$\Rightarrow I^{s(k,k)} = I^{t^k}$$

$q^{k-1} \xrightarrow{t^k} q^k$, Definition 1.9 and Definition 1.8

$$\Rightarrow q^{k-1} - I^{t^k} \geq 0$$
$$\Rightarrow q^{k-1} - I^{s(k,k)} \geq 0$$

STEP:

Letting the induction hypothesis be

$$q^{i-1} - I^{s(i,k)} \geq 0 \text{ (i.e. } q^{i-1} \geq I^{s(i,k)})$$

it will be shown that

$$q^{i-2} - I^{s(i-1,k)} \geq 0.$$

$q^0 \xrightarrow{s} q^k$ and Definition 1.14

$$\Rightarrow q^{i-2} \xrightarrow{t^{i-1}} q^{i-1}$$

Definition 1.1

$$\Rightarrow q^{i-1} = q^{i-2} - I^{t^{i-1}} + 0^{t^{i-1}}$$

$q^{i-1} \geq I^{s(i,k)}$

$$\Rightarrow q^{i-2} - I^{t^{i-1}} + 0^{t^{i-1}} \geq I^{s(i,k)}$$

$$\Rightarrow q^{i-2} \geq I^{s(i,k)} + I^{t^{i-1}} - 0^{t^{i-1}}$$

Definition 1.8

$$\Rightarrow q^{i-2} - I^{t^{i-1}} \geq 0$$

$$\Rightarrow q^{i-2} \geq I^{t^{i-1}}$$

Therefore $q_j^{i-2} \geq \left\langle \max(I_j^{t^{i-1}}, I_j^{s(i,k)} + I_j^{t^{i-1}} - 0_j^{t^{i-1}}) \right\rangle$

But Definition 2.8

$$\Rightarrow I^{s(i-1,k)} = \left\langle \max(I_j^{t^{i-1}}, I_j^{s(i,k)} + I_j^{t^{i-1}} - 0_j^{t^{i-1}}) \right\rangle$$

$$\Rightarrow q^{i-2} \geq I^{s(i-1,k)}$$

$$\Rightarrow q^{i-2} - I^{s(i-1,k)} \geq 0$$

LEMMA 2.15

$$q^0 - I^s \geq 0$$

$$\Rightarrow$$

$$q^0 \xrightarrow{s} q^k$$

Proof:

First proceeding by induction on the transition subsequence length i, it will be shown that $q^i - I^{s(i+1,k)} \geq 0$.

BASE:

Letting $i = 1$

$\Rightarrow I^s = I^{s(i,k)} = I^{s(1,k)}$

and $q^0 - I^s \geq 0$

$\Rightarrow q^0 - I^{s(1,k)} \geq 0$

STEP

Assuming as the induction hypothesis that

$q^{i-1} - I^{s(i,k)} \geq 0$

it will be shown that $q^{i-1} \xrightarrow{t^i} q^i$ and that $q^i - I^{s(i+1,k)} \geq 0$.

$q^{i-1} - I^{s(i,k)} \geq 0$

$\Rightarrow q^{i-1} \geq I^{s(i,k)}$

Definition 2.8

$$\Rightarrow I^{s(i,k)} = \left\langle \max(I_j^{t^i}, I_j^{s(i+1,k)} + I_j^{t^i} - O_j^{t^i}) \right\rangle$$

$$\Rightarrow q^{i-1} \geq \left\langle \max(I_j^{t^i}, I_j^{s(i+1,k)} + I_j^{t^i} - O_j^{t^i}) \right\rangle$$

$$\Rightarrow q^{i-1} \geq I^{t^i} \text{ and } q^{i-1} \geq I^{s(i+1,k)} + I^{t^i} - O^{t^i}$$

$q^{i-1} \geq I^{t^i}$

$\Rightarrow q^{i-1} - I^{t^i} \geq 0$

Definition 1.8

$$\Rightarrow q^{i-1} \xrightarrow{t^i}$$

Definition 1.9

$$\Rightarrow q^{i-1} \xrightarrow{t^i} q^i \text{ where } q^i = q^{i-1} - I^{t^i} + 0^{t^i}$$

$$q^{i-1} \geq I^{s(i+1,k)} + I^{t^i} - 0^{t^i} \text{ and } q^i = q^{i-1} - I^{t^i} + 0^{t^i}$$

$$\Rightarrow q^i + I^{t^i} - 0^{t^i} \geq I^{s(i+1,k)} + I^{t^i} - 0^{t^i}$$

$$\Rightarrow q^i \geq I^{s(i+1,k)}$$

$$\Rightarrow q^i - I^{s(i+1,k)} \geq 0$$

Thus it can be shown that each succeeding state q^i is enabled up to and including q^{k-1}.

But by Definition 2.8, $I^{s(k,k)} = I^{t^k}$. Hence $q^{k-1} - I^{t^k} \geq 0$. Therefore $q^{k-1} \xrightarrow{t^k} q^k$ and $q^0 \xrightarrow{s} q^k$.

The preceding two lemmata are summarized in the following theorem.

<u>THEOREM</u> 2.16

$$q^0 \xrightarrow{s} q^k$$

$$\Leftrightarrow$$

$$q^0 - I^s \geq 0$$

Proof:

This follows from Lemma 2.14 and Lemma 2.15

It can now be shown that a necessary and sufficient condition for a transition sequence s to be enabled for a state q, is that q exceed the input vector corresponding to transition t.

THEOREM 2.17

$$q^0 - I^s \geq 0$$

$$\Longleftrightarrow$$

$$q^0 \xrightarrow{s}$$

Proof:

a) $q^0 - I^s \geq 0$,

$q^0 - I^s \geq 0$ and Lemma 2.15

$\Rightarrow q^0 \xrightarrow{s} q^k$

Definition 2.1

$\Rightarrow q^0 \xrightarrow{s}$

b) $q^0 \xrightarrow{s}$,

$q^0 \xrightarrow{s}$ and Definition 2.1

$\Rightarrow q^0 \xrightarrow{s} q^k$

Lemma 2.14

$\Rightarrow q^0 - I^s \geq 0$

Thus it has been shown that the definition of I^s presented in this section, coupled with the definition of state enabling presented in Definition 2.1 yields necessary and sufficient conditions for state enabling which are consistent with Definition 1.8. Thus Theorem 2.17 could be considered an extension of Definition 1.8 to transition sequences.

2.5 OUTPUT VECTOR

As for transitions, an output vector can also be defined for transition sequences.

DEFINITION 2.18 OUTPUT VECTOR FOR TRANSITION SEQUENCE

For transition sequence $s = t^1, \ldots, t^k$, the output vector, denoted 0^s, is defined to be $0^s = D^s + I^s$. i.e. $D^s = - I^s + 0^s$.

2.6 MONOTONICITY

A transition sequence is enabled for any state which exceeds (is $\geq$) the input or threshold vector. Thus if a transition sequence is enabled for a state q and some other state, say q′, exceeds q then the transition sequence enabled for q is enabled for q′ as well. This property is called monotonicity.

LEMMA 2.19

$$q \xrightarrow{s}, \; q' \geq q$$
$$\Rightarrow$$
$$q' \xrightarrow{s}$$

Not only does monotonicity hold, but the following is also true.

LEMMA 2.20

$$q \xrightarrow{s} q' \text{ and } x \text{ an n-tuple with all components } x_j \in N$$
$$\Rightarrow$$
$$q + x \xrightarrow{s} q' + x$$

Chapter III

CUBE NOTATION

3.1 CUBES

The Cube notation presented here is one way of representing certain sets containing an infinite number of states as a finite number of sets of states called cubes.

The cube notation is such that while a single cube may represent an infinite number of states, it is possible to determine the membership of a particular state in the cube by inspection.

This notation was inspired by the notion of "cubical complexes" as described in [Roth 1956] and by the w-notation in [Karp & Miller 1968]. It was first introduced in [Laucht 1979] and is refined and extended here.

As will be seen, cubes are n-tuples with two kinds of coordinates, here called components, natural numbers and unbounded $^+$-components. The latter are introduced first.

DEFINITION 3.1 $^+$-COMPONENT

For $k \in N$, k^+ is called a $^+$-component or an unbounded component and is defined as:

$$k^+ = \{ k, k+1, k+2, \dots \}$$

For example $5^+ = \{5, 6, 7, \dots \}$.

As is the case for all sets, equality of two $^+$-components requires their having the same elements.

DEFINITION 3.2 $^+$-COMPONENT EQUALITY

Two $^+$-components, k^+ and j^+, are equal, denoted $k^+ = j^+$, if the sets they represent, $\{k,k+1,k+2,...\}$ and $\{j,j+1,j+2,...\}$ respectively, are equal.

The set of all possible $^+$-components is called N^+.

DEFINITION 3.3 N^+

The set N^+ consists of all k^+ for $k \in N$.

$$\text{i.e. } N^+ = \{ 0^+, 1^+, 2^+, \ldots \}$$

All $^+$-components are specified in terms of their "smallest" element. In order to extract this element the glb operator (greatest lower bound) is introduced.

DEFINITION 3.4 glb for $^+$-COMPONENT

For $k^+ \in N^+$, the greatest lower bound or glb of k^+ is defined to be k, where $k \in k^+$ and $k \leq j$ for all $j \in k^+$.

$$\text{i.e. } \mathrm{glb}(k^+) = k$$

The following lemma relates the equality of two $^+$-components to the equality of their respective greatest lower bounds.

LEMMA 3.5

$$k^{+} = j^{+}$$
$$\Longleftrightarrow$$
$$glb(k^{+}) = glb(j^{+})$$
$$\text{i.e. } k = j$$

Proof:

This follows from Definition 3.1, Definition 3.2, Definition 3.4 and set equality.

The following relationship will be useful in later proofs.

LEMMA 3.6

$$k^{+} \in N^{+} \Rightarrow [glb(k^{+})]^{+} = k^{+}$$

Proof:

This follows directly from Definition 3.4.

A cube component is now defined to be either a natural number or a $^{+}$-component.

DEFINITION 3.7 CUBE COMPONENT

A cube component, denoted by c_j is:

$$c_j \in N \cup N^{+}$$
$$\text{i.e. } c_j \in \{ 0, 1, 2, \ldots \} \cup \{ 0^{+}, 1^{+}, 2^{+}, \ldots \}$$
$$\text{i.e. } c_j \in \{ 0, 0^{+}, 1, 1^{+}, 2, 2^{+}, \ldots \}$$

A cube will now be defined as an n-tuple of cube components which represent a finite or infinite number of states.

DEFINITION 3.8 CUBE

A cube c, denoted as an n-tuple of cube components,

i.e. $c = \langle c_1, \ldots, c_n \rangle$

is defined to be the set of states { q } such that

$$c = \{ \langle q_1,\ldots,q_n \rangle \mid q_j \begin{cases} = c_j & \text{for } c_j \in N \\ \geq \mathrm{glb}(c_j) & \text{for } c_j \in N^+ \end{cases}, \text{ for } j = 1,\ldots,n \}$$

EXAMPLE 3.9

The cube $c = \langle c_1,c_2 \rangle = \langle 0^+,1 \rangle$, where $c_1 = 0^+ \in N^+$ and $c_2 = 1 \in N$, consists of the infinite set of states { < 0,1 >, < 1,1 >, < 2,1 >, < 3,1 >, ... }.

3.2 CUBE MINIMUM

Also useful will be the notion of a "smallest" state in a cube. This will be done in terms of a "smallest" cube component.

DEFINITION 3.10 MINIMUM OF A CUBE COMPONENT

The minimum of a cube component c_j, denoted $\min(c_j)$, is defined to be:

$$\min(c_j) = \begin{cases} c_j & \text{for } c_j \leqslant N \\ \text{glb}(c_j) & \text{for } c_j \leqslant N^+ \end{cases}$$

The "smallest" state is now defined to have as its components the minimum cube components.

DEFINITION 3.11 MINIMUM OF A CUBE

The minimum of a cube c, denoted $\text{MIN}(c)$, is an n-tuple q such that:

$$q = \text{MIN}(c) = \langle q_1, \ldots, q_n \rangle \text{ where } q_j = \min(c_j) \text{ for } j = 1, \ldots, n$$

The minimum of a cube as defined in Definition 3.11 is a state and in fact is an element of c.

LEMMA 3.12

$$q = \text{MIN}(c)$$

$$\Rightarrow$$

$$q \text{ a state and } q \leqslant c$$

Proof:

$q = MIN(c)$ and Definition 3.11

$\Rightarrow q_j = \min(c_j)$ for $1 \leq j \leq n$

Definition 1.1

$\Rightarrow q$ a state

Definition 3.8

$\Rightarrow q \preccurlyeq c$

The following property of the MIN operator will be useful later.

<u>LEMMA</u> 3.13

$$q \preccurlyeq c$$

$$\Leftrightarrow$$

$$q = MIN(c) + x$$

$$\text{where } x_j = 0 \text{ for } j : c_j \preccurlyeq N \text{ and}$$

$$\preccurlyeq N \text{ for } j : c_j \preccurlyeq N^+$$

Proof:

$q \preccurlyeq c$

$$\begin{aligned} \Leftrightarrow q_j &= c_j && \text{for } j : c_j \preccurlyeq N \text{ and} \\ &\geq glb(c_j) && \text{for } j : c_j \preccurlyeq N^+ \\ \Leftrightarrow q_j &= c_j + 0 && \text{and} \\ &= glb(c_j) + x_j && \text{respectively for some } x_j \preccurlyeq N \\ \Leftrightarrow q &= MIN(c) + x && \text{for } x \text{ defined as above.} \end{aligned}$$

This proof is reversible.

3.3 CUBE PROPERTIES

3.3.1 Cube Equality

DEFINITION 3.14 CUBE EQUALITY

Two cubes, c and c', are said to be equal if they contain the same states.

3.3.2 Cube Uniqueness

LEMMA 3.15 UNIQUENESS OF CUBES

The cube notation for a given set of states, if it exists, is unique.

Proof:

The contrary is assumed and it is postulated that some set of states, say Q, has 2 cube notations, say c^1 and c^2, associated with it and that $c^1 \neq c^2$. This will be shown to lead to a contradiction.

For any q, q' such that $q, q' \in Q$, Definition 3.8

$$\Rightarrow q, q' \in c^1 \text{ and } q, q' \in c^2$$

Defining $q = \mathrm{MIN}(c^1)$

$$\Rightarrow q = \left\langle q_j \text{ such that } q_j = \min(c_j^1) \right\rangle$$

$$\Rightarrow q = \left\langle q_j \text{ such that } q_j = \begin{matrix} c_j \text{ for } c_j^1 \in N \\ \mathrm{glb}(c_j^1) \text{ for } c_j^1 \in N^+ \end{matrix} \right\rangle$$

and defining $q' = MIN(c^2)$

$$\Rightarrow q' = \left\{ q'_j \text{ such that } q'_j = \min(c^2_j) \right\}$$

$$\Rightarrow q' = \left\{ q'_j \text{ such that } q'_j = \begin{array}{l} c_j \text{ for } c^2_j \in N \\ glb(c^2_j) \text{ for } c^2_j \in N^+ \end{array} \right\}$$

$c^1 \neq c^2$

$\Rightarrow c^1$ and c^2 differ in at least one component.

Let the first such component be j.

There are 4 cases to be examined.

a) for $c^1_j, c^2_j \in N$

Definition 3.8 and $q \in c^1$

$\Rightarrow q_j = c^1_j$

Definition 3.8 and $q \in c^2$

$\Rightarrow q_j = c^2_j$

$\Rightarrow c^1_j = c^2_j$ which is a contradiction.

b) For $c^1_j, c^2_j \in N^+$

Definition 3.8 and $q \in c^2$

$\Rightarrow q_j \geq glb(c^2_j)$

Definition 3.11, Definition 3.10 and $q = MIN(c^1)$

$\Rightarrow q_j = glb(c^1_j)$

$\Rightarrow glb(c^1_j) \geq glb(c^2_j)$

Similarly, Definition 3.8 and $q' \in c^1$

$\Rightarrow q'_j \geq glb(c^1_j)$

Definition 3.11, Definition 3.10 and $q' = MIN(c^2)$

$\Rightarrow q'_j = glb(c_j^2)$

$\Rightarrow glb(c_j^2) \geq glb(c_j^1)$

Therefore $glb(c_j^2) = glb(c_j^1)$

and Lemma 3.5

$\Rightarrow c_j^2 = c_j^1$ but this is a contradiction.

c) For $c_j^1 \in N$, $c_j^2 \in N^+$

$q \in c^1$ and Definition 3.8

$\Rightarrow q_j = c_j^1$

$q \in c^2$ and Definition 3.8

$\Rightarrow q_j \geq glb(c_j^2)$

$\Rightarrow c_j^1 \geq glb(c_j^2)$

This implies that there exist values for the jth component of states in c^2 which do not exist in c^1, specifically those $< c_j^1$ and those $> c_j^1$.

This further implies that there exist states in c^2 which do not exist in c^1. This is a contradiction.

d) For $c_j^1 \in N^+$, $c_j^2 \in N$

This case is symmetric with case c) above.

3.4 SUBCUBES

If one cube is a subset of another, i.e. if it represents a subset of states of the other, then it is called a subcube.

DEFINITION 3.16 SUBCUBE

If c, c′ are cubes and c is a subset of c′, i.e. $q \leqslant c \Rightarrow q \leqslant c'$, then c is said to be a subcube of c′. This is denoted c SUB c′.

The following lemmata will yield more readily usable necessary and sufficient conditions for one cube to be a subcube of another.

LEMMA 3.17

Given two cubes c, c′,

$$\text{if } c_j, c'_j \leqslant N \text{ and } c_j \neq c'_j, \text{ for some } j,\ 1 \leq j \leq n$$
$$\Rightarrow$$
$$c \sim\text{SUB } c' \text{ and } c' \sim\text{SUB } c$$

Proof:

It is sufficient to find a state $q \leqslant c$ and $q \nleqslant c'$.

For any $q \leqslant c$, Definition 3.8

$\Rightarrow q_j = c_j$ for $c_j \leqslant N$

and for any $q' \leqslant c'$, Definition 3.8

$\Rightarrow q'_j = c'_j$ for $c'_j \leqslant N$.

But $c_j \neq c'_j$ and $c_j = q_j$

$\Rightarrow q_j \neq c'_j$ and $q \nleqslant c'$.

Definition 3.16

$\Rightarrow c \sim\text{SUB } c'$.

By symmetry of c and c′, $c' \sim\text{SUB } c$ also holds.

LEMMA 3.18

Given two cubes c, c',

$$\text{glb}(c_j) < \text{glb}(c'_j) \text{ for } c_j,\ c'_j \in N^+ \text{ for some } j,\ 1 \leq j \leq n$$

$$\Rightarrow$$

$$c \text{ \~{}SUB } c'.$$

Proof:

Letting $q = \text{MIN}(c)$,

Definition 3.10 and Definition 3.11

$\Rightarrow q_j = \text{glb}(c_j)$ for $c_j \in N^+$

$\text{glb}(c_j) < \text{glb}(c'_j)$

$\Rightarrow q_j < \text{glb}(c'_j)$

Definition 3.8

$\Rightarrow q \notin c'$.

Definition 3.16

$\Rightarrow c$ ~SUB c'.

LEMMA 3.19

Given two cubes c, c′,

$$c_j < \mathrm{glb}(c'_j),\ c_j \preccurlyeq N$$
$$\text{and } c'_j \preccurlyeq N^+ \text{ for some } j,\ 1 \leq j \leq n$$
$$\Rightarrow$$
$$c \ \tilde{}\mathrm{SUB}\ c'.$$

Proof:

Letting $q = \mathrm{MIN}(c)$

Definition 3.10 and Definition 3.11

$\Rightarrow q \preccurlyeq c$

Definition 3.8

$\Rightarrow q_j = c_j$ for $c_j \preccurlyeq N$.

$c_j < \mathrm{glb}(c'_j)$ and $q_j = c_j$

$\Rightarrow q_j < \mathrm{glb}(c'_j)$

Definition 3.8

$\Rightarrow q \not\preccurlyeq c'$

Definition 3.16

$\Rightarrow c \ \tilde{}\mathrm{SUB}\ c'$

LEMMA 3.20

Given two cubes c, c′,

if there exist some c_b, c'_b such that $c_b \notin N^+$ and $c'_b \notin N$

=>

c ~SUB c′

Proof:

Assuming that there exists some pair of corresponding components of c and c′, say c_b and c'_b, such that $c_b \notin N^+$ and $c'_b \notin N$ and with q chosen as:

$$q = \left\langle q_j \mid q_j = \left\{ \begin{array}{ll} \min(c_j) & \text{for } j \neq b \\ \max(c'_b, \mathrm{glb}(c_b)) + 1 & \text{for } j = b \end{array} \right\}, \text{ for } 1 \leq j \leq n \right\rangle$$

then Definition 3.8

=> $q \in c$

Lemma 2.7

=> $\max(c'_b, \mathrm{glb}(c_b)) \geq c'_b$

=> $\max(c'_b, \mathrm{glb}(c_b)) + 1 > c'_b$

=> $q_j > c'_j$ for $j = b$

Definition 3.8

=> $q \notin c'$

Definition 3.16

=> c ~SUB c′.

LEMMA 3.21

Given two cubes c and c′ such that:

a) $c_j = c'_j$ for $c_j, c'_j \lessdot N$

b) $glb(c_j) \geq glb(c'_j)$ for $c_j, c'_j \lessdot N^+$

c) $c_j \geq glb(c'_j)$ for $c_j \lessdot N$, $c'_j \lessdot N^+$

d) There exist no c_j, c'_j such that $c_j \lessdot N^+$ and $c'_j \lessdot N$

then c SUB c′.

Proof:

It must be shown that any $q \lessdot c$ is also $\lessdot c'$. That is, if $c'_j \lessdot N$ then q must be $= c'_j$ and if $c'_j \lessdot N^+$ then q must be $\geq glb(c'_j)$. There are four possibilities in total:

a) For $c_j, c'_j \lessdot N$

$q \lessdot c$ and Definition 3.8

$$\Rightarrow q_j = c_j$$

but $c_j, c'_j \lessdot N$

$$\Rightarrow c_j = c'_j$$

$$\Rightarrow q_j = c'_j$$

b) For $c_j, c'_j \lessdot N^+$

$q \lessdot c$ and Definition 3.8

$$\Rightarrow q_j \geq glb(c_j)$$

but $c_j, c'_j \lessdot N^+$

$$\Rightarrow glb(c_j) \geq glb(c'_j)$$

$$\Rightarrow q_j \geq glb(c'_j)$$

c) For $c_j \leqslant N$, $c'_j \leqslant N^+$

$q \leqslant c$ and Definition 3.8

$\Rightarrow q_j = c_j$

but $c_j \leqslant N$, $c'_j \leqslant N^+$

$\Rightarrow c_j \geq glb(c'_j)$

$\Rightarrow q_j \geq glb(c'_j)$

d) For $c_j \leqslant N^+$, $c'_j \leqslant N$

The assumptions made in this case cannot occur.

Thus Definition 3.8

$\Rightarrow q \leqslant c'$

$\Rightarrow$ c SUB c'

Summarizing the preceding results yields:

<u>THEOREM</u> 3.22 SUBCUBE ATTRIBUTES

A cube c is a subcube of another cube c' iff

a) $c_j = c'_j$	for c_j, $c'_j \leqslant N$
b) $glb(c_j) \geq glb(c'_j)$	for c_j, $c'_j \leqslant N^+$
c) $c_j \geq glb(c'_j)$	for $c_j \leqslant N$, $c'_j \leqslant N^+$

d) There exist no c_j, c'_j such that $c_j \leqslant N^+$ and $c'_j \leqslant N$

Proof:

The necessity of a), b), c) and d) is shown in Lemma 3.17, Lemma 3.18, Lemma 3.19 and Lemma 3.20 respectively. The sufficiency is shown in Lemma 3.21

COROLLARY 3.23

$$c \text{ SUB } c' \Rightarrow \text{MIN}(c) \leq \text{MIN}(c')$$

COROLLARY 3.24

$$c \text{ SUB } c' \text{ and } c'_j \lessdot N \text{ for some } j,\ 1 \leq j \leq n$$

$$\Rightarrow$$

$$c_j \lessdot N \text{ and } c_j = c'_j$$

COROLLARY 3.25

$$c \text{ SUB } c',\ c' \text{ SUB } c'' \Rightarrow c \text{ SUB } c''$$

3.5 CUBE $\pm$ OPERATIONS

DEFINITION 3.26 CUBE +, - OPERATIONS

For any cube c and any n-tuple x with $x_j \in Z$, the operations of addition and subtraction are defined component-wise as follows:

$$y = c \pm x$$

$$= \langle c_j \pm x_j \rangle$$

$$= \left\langle y_j \mid y_j = \begin{cases} c_j \pm x_j & \text{for } c_j \in N \\ [\mathrm{glb}(c_j) \pm x_j]^+ & \text{for } c_j \in N^+ \end{cases}, \text{ for } j,\ 1 \leq j \leq n \right\rangle$$

Note that this definition has no meaning if any

$$\mathrm{glb}(c_j) \pm x_j \notin N$$

or if any

$$c_j \pm x_j \notin N.$$

The following results will prove useful later.

LEMMA 3.27

For any cube c and any two n-tuples x and y with $x_j, y_j \in N$ for $1 \leq j \leq n$,

a) $(c + x) \pm y = c + (x \pm y)$

b) $(c - x) \pm y = c - (x \mp y)$

provided in each case that the left hand side and right hand side are defined.

Proof:

This follows from Definition 3.26 and from associativity for integer addition and subtraction.

LEMMA 3.28

$$c_j \in N^+,\ x_j \in Z \text{ and } glb(c_j) \pm x_j \in N$$
$$\Rightarrow$$
$$glb(c_j \pm x_j) = glb(c_j) \pm x_j$$

Proof:

For $c_j \in N^+$, $x_j \in Z$ and $glb(c_j) \pm x_j \in N$, Definition 3.26

$\Rightarrow c_j \pm x_j = [glb(c_j) \pm x_j]^+$

Lemma 3.5

$\Rightarrow glb(c_j \pm x_j) = glb([glb(c_j) \pm x_j]^+)$

Definition 3.4

$\Rightarrow glb(c_j \pm x_j) = glb(c_j) \pm x_j$

LEMMA 3.29

$$c_j \in N^+,\ x_j \text{ and } y_j \in N, \text{ and } glb(c_j) - x_j \geq 0$$
$$\Rightarrow$$
$$glb(c_j - x_j + y_j) = glb(c_j) - x_j + y_j$$

Proof:

Lemma 3.28

$\Rightarrow glb(c_j) - x_j = glb(c_j - x_j)$

$\Rightarrow glb(c_j) - x_j + y_j = glb(c_j - x_j) + y_j$

$y_j \in N$

$\Rightarrow y_j \geq 0$

$\Rightarrow glb(c_j - x_j) + y_j \geq 0$

applying Lemma 3.28 again

$$\Rightarrow glb(c_j) - x_j + y_j = glb(c_j - x_j + y_j)$$

<u>LEMMA</u> 3.30

For c a cube and x an n-tuple with all $x_j \leqslant N$,

$$MIN(c) \pm x \geq 0$$

$$\Rightarrow$$

$$c \pm x \text{ is a cube}$$

Proof:

$MIN(c) \pm x \geq 0$, Definition 3.11 and Definition 3.10

$\Rightarrow c_j \pm x_j \geq 0$ for $c_j \leqslant N$ and

$glb(c_j) \pm x_j \geq 0$ for $c_j \leqslant N^+$

Lemma 3.28

$\Rightarrow c_j \pm x_j \geq 0$ for $c_j \leqslant N$ and

$glb(c_j \pm x_j) \geq 0$ for $c_j \leqslant N^+$

$\Rightarrow c_j \pm x_j \leqslant N$ for $c_j \leqslant N$

Definition 3.3

$\Rightarrow [glb(c_j \pm x_j)]^+ \leqslant N^+$ for $c_j \leqslant N^+$

Definition 3.7 and Definition 3.8

$\Rightarrow c \pm x$ is a cube

COROLLARY 3.31

For c a cube and x an n-tuple with all $x_j \in N$, and MIN(c) $\pm$ x $\geq$ 0, and $c' = c \pm x$,

$$c_j \in N \iff c'_j \in N$$

and

$$c_j \in N^+ \iff c'_j \in N^+$$

LEMMA 3.32

For c a cube and x an n-tuple with all $x_j \in N$,

c + x is a cube

Proof:

All $x_j \in N$

=> all $x_j \geq 0$

Definition 3.10, Definition 3.4 and Definition 3.1

=> all $\min(c_j) \geq 0$ for $1 \leq j \leq n$

=> all $\min(c_j) + x_j \geq 0$

Definition 3.11

=> MIN(c) + x $\geq$ 0

Lemma 3.30

=> c + x is a cube

Chapter IV

CUBES AND TRANSITIONS

4.1 CUBE ENABLED

A transition was said to be enabled for a given state if it could fire. Similarly, a transition is said to be enabled for a cube if the cube contains at least one state for which the transition is enabled.

DEFINITION 4.1 CUBE ENABLED

A transition t is said to be cube enabled for a cube c if there exists some state q such that

$$q \leqslant c \text{ and } q \xrightarrow{t}.$$

This is denoted $c \xrightarrow{t}$.

Furthermore, it follows immediately that a transition is enabled for a given cube if it is enabled for some subcube.

LEMMA 4.2

$$c' \xrightarrow{t} \text{ and } c' \text{ SUB } c$$
$$\Rightarrow$$
$$c \xrightarrow{t}.$$

Proof:

$c' \xrightarrow{t}$ and Definition 4.1

$\Rightarrow$ There exists q such that $q \leqslant c'$ and $q \xrightarrow{t}$

c' SUB c

$\Rightarrow q \leqslant c$

Definition 4.1

$\Rightarrow c \xrightarrow{t}$

The next lemma relates the concept of enabling of a cube to the exceeding of the input vector for the given transition.

LEMMA 4.3

There exist some c and c′ such that c′ SUB c and $MIN(c') - I^t \geq 0$

$$\Longleftrightarrow$$

$$c \xrightarrow{t}$$

Proof:

a) There exist some c and c′ such that c′ SUB c and $MIN(c') - I^t \geq 0$

Letting $q = MIN(c')$, Lemma 3.12

$\Rightarrow$ q a state and $q \leqslant c'$

Definition 3.16

$\Rightarrow q \leqslant c$

$MIN(c') - I^t \geq 0$

$\Rightarrow q - I^t \geq 0$

Definition 1.8

$\Rightarrow q \xrightarrow{t}$

Definition 4.1

$\Rightarrow c \xrightarrow{t}$

b) $c \xrightarrow{t}$

$c \xrightarrow{t}$ and Definition 4.1

=> There exists some q such that $q \in c$ and $q \xrightarrow{t}$

Definition 1.8

=> $q - I^t \geq 0$

Letting $c' = \{q\}$, Definition 3.11 and Definition 3.10

=> $MIN(c') = q$

=> $MIN(c') - I^t \geq 0$

$c' = \{q\}$, Definition 3.16

=> c' SUB c

4.2 ENABLING CUBE

If a cube is enabled for a transition, then the subcube which contains precisely those states for which the given transition is enabled, is called the enabling cube with respect to the given cube and transition. This result is developed from the following definition.

DEFINITION 4.4 ENABLING CUBE

The enabling cube, denoted $\underline{c}(t)$, for a given cube c and transition t is an n-tuple defined as:

$$\underline{c}(t) = \left\langle \underline{c}_j(t), \text{ for } j,\ 1 \leq j \leq n \right\rangle$$

where:

$$\underline{c}_j(t) = c_j \text{ for } c_j \in N$$

and

$$\underline{c}_j(t) = [\ \max(\ \mathrm{glb}(c_j), I_j^t\ )\ ]^+ \text{ for } c_j \in N^+.$$

In the following lemma it will be shown that $\underline{c}(t)$ is, in fact, a cube.

LEMMA 4.5

$\underline{c}(t)$ as defined in Definition 4.4 is a cube.

Proof:

It must be shown that all components $\underline{c}_j(t)$ are either $\in N$ or $\in N^+$.

For those $c_j \in N$,

$\underline{c}_j(t) = c_j$

$\quad\Rightarrow \underline{c}_j(t) \in N.$

For those $c_j \leqslant N^+$, Definition 4.4

$$\Rightarrow \underline{c}_j(t) = [\max \text{ of } (\text{glb}(c_j), I_j^t)]^+$$

but $\text{glb}(c_j)$ and I_j^t are both $\leqslant N$

$$\Rightarrow \max(\text{glb}(c_j), I_j^t) \text{ also } \leqslant N.$$

Definition 3.1

$$\Rightarrow [\max(\text{glb}(c_j), I_j^t)]^+ \leqslant N^+.$$

Definition 3.8

$$\Rightarrow \underline{c}(t) \text{ is a cube.}$$

COROLLARY 4.6

$$\underline{c}_j(t) \leqslant N \iff c_j \leqslant N$$

and

$$\underline{c}_j(t) \leqslant N^+ \iff c_j \leqslant N^+$$

The necessary and sufficient condition for membership of a state in an enabling cube are needed often enough to warrant the following lemma.

LEMMA 4.7

$$q \leqslant \underline{c}(t)$$

$$\Longleftrightarrow$$

$$q_j = c_j \quad \text{for } c_j \leqslant N \text{ and}$$
$$\geq \max(\, glb(c_j), I_j^t) \quad \text{for } c_j \leqslant N^+$$

Proof:

a) =>

$q \leqslant \underline{c}(t)$ and Definition 3.8

$$\Rightarrow q_j = \underline{c}_j(t) \quad \text{for } \underline{c}_j(t) \leqslant N \text{ and}$$
$$\geq glb(\underline{c}_j(t)) \text{ for } \underline{c}_j(t) \leqslant N^+$$

But Definition 4.4

$$\Rightarrow \underline{c}_j(t) = c_j \quad \text{for } \underline{c}_j(t),\ c_j \leqslant N \text{ and}$$
$$= [\ \max(\, glb(c_j), I_j^t\ )\ ]^+ \text{ for } \underline{c}_j(t),\ c_j \leqslant N^+$$

$$\Rightarrow q_j = c_j \quad \text{for } c_j \leqslant N \text{ and}$$
$$\geq glb(\ [\ \max(\, glb(c_j), I_j^t\ )\ ]^+\ ) \text{ for } c_j \leqslant N^+$$

$$\Rightarrow q_j = c_j \quad \text{for } c_j \leqslant N \text{ and}$$
$$\geq \max(\, glb(c_j), I_j^t\ ) \quad \text{for } c_j \leqslant N^+$$

b) <=

Each step in the above proof is reversible.

It is now shown that the enabling cube for a given cube and transition is indeed a subcube of the given cube.

<u>LEMMA</u> 4.8

c a cube and $\underline{c}(t)$ defined as in Definition 4.4

$$\Rightarrow$$

$$\underline{c}(t) \text{ SUB } c$$

Proof:

It must be shown that any $q \in \underline{c}(t)$ is also $\in c$.

Letting $q \in \underline{c}(t)$

Lemma 4.7

$$\Rightarrow q_j = c_j \qquad \text{for } c_j \in N \text{ and}$$
$$\geq \max(\, glb(c_j), I_j^t \,) \text{ for } c_j \in N^+$$

$$\Rightarrow q_j = c_j \qquad \text{for } c_j \in N \text{ and}$$
$$\geq glb(c_j) \text{ for } c_j \in N^+$$

Definition 3.8

$$\Rightarrow q \in c$$

Definition 3.16

$$\Rightarrow \underline{c}(t) \text{ SUB } c$$

It will next be shown a necessary and sufficient condition, expressed in terms of $\underline{c}(t)$ and I^t, for t to be enabled for c.

LEMMA 4.9

$$c \xrightarrow{t}$$

$$\Rightarrow$$

$$\mathrm{MIN}(\ \underline{c}(t)\ ) - I^t \geq 0$$

Proof:

Definition 3.11

$$\Rightarrow \mathrm{MIN}(\underline{c}(t)) - I^t = \left\langle \min(\underline{c}_j(t)) - I^t_j \mid 1 \leq j \leq n \right\rangle$$

Definition 1.2, Definition 3.11 and Definition 3.10

$$\Rightarrow \mathrm{MIN}(\underline{c}(t)) - I^t = \left\langle \begin{array}{ll} \underline{c}_j(t) - I^t_j & \text{for } \underline{c}_j(t) \leq N \\ \mathrm{glb}(\underline{c}_j(t)) - I^t_j & \text{for } \underline{c}_j(t) \lessdot N^+ \end{array} \mid 1 \leq j \leq n \right\rangle$$

a) for those $\underline{c}_j(t) \lessdot N$,

Corollary 4.6 and Definition 4.4

$$\Rightarrow c_j \lessdot N \text{ and } \underline{c}_j(t) = c_j$$

furthermore, $c \xrightarrow{t}$ and Lemma 4.3

$$\Rightarrow \text{ there exists } c' \text{ SUB } c \text{ such that } \mathrm{MIN}(c') - I^t \geq 0$$

$c_j \lessdot N$ and Corollary 3.24

$$\Rightarrow c'_j \lessdot N$$

Definition 3.11

$$\Rightarrow \left\langle \min(c'_j) - I^t_j \text{ where } 1 \leq j \leq n \right\rangle \geq 0$$

Definition 3.10

$\Rightarrow c'_j - I_j^t \geq 0$ for $c'_j \in N$

subcube attributes a) and d) in Theorem 3.22

$\Rightarrow$ that for each $c_j \in N$ there corresponds a $c'_j \in N$ such that $c'_j = c_j$

$\Rightarrow c'_j = c_j = \underline{c}_j(t)$

$\Rightarrow \underline{c}_j(t) - I_j^t \geq 0$ for $\underline{c}_j(t) \in N$

b) for those $\underline{c}_j(t) \in N^+$

Lemma 4.7

$\Rightarrow glb(\underline{c}_j(t)) - I_j^t = max(glb(c_j), I_j^t) - I_j^t$

At this point there are two possibilities:

b.1) if: $glb(c_j) \geq I_j^t$

then Definition 2.5

$\Rightarrow max(glb(c_j), I_j^t) - I_j^t = glb(c_j) - I_j^t$

$c \xrightarrow{t}$, Lemma 4.3, Definition 3.11 and Definition 3.10

$\Rightarrow glb(c_j) - I_j^t \geq 0$

b.2) if: $glb(c_j) < I_j^t$

then: Definition 2.5

$\Rightarrow max(glb(c_j), I_j^t) - I_j^t = I_j^t - I_j^t = 0$

Thus $glb(\underline{c}_j(t)) - I_j^t \geq 0$ for $\underline{c}_j(t) \in N^+$

Thus Definition 3.11 and Definition 3.10

$\Rightarrow MIN(\underline{c}(t)) - I^t \geq 0.$

LEMMA 4.10

$$\mathrm{MIN}(\ \underline{c}(t)\ ) - I^t \geq 0$$
$$\Rightarrow$$
$$c \xrightarrow{t}$$

Proof:

Lemma 4.8

$\Rightarrow$ $\underline{c}(t)$ SUB c

Lemma 4.3 and $\mathrm{MIN}(\underline{c}(t)) - I^t \geq 0$

$\Rightarrow$ $c \xrightarrow{t}$

Summarizing the preceding two lemmata yields:

THEOREM 4.11

$$c \xrightarrow{t}$$
$$\Leftrightarrow$$
$$\mathrm{MIN}(\underline{c}(t)\ ) - I^t \geq 0$$

Proof:

Lemma 4.9 shows the => part and Lemma 4.10 shows the <= part.

The proof that the enabling cube is in fact precisely the set of states in c for which t is enabled is shown in the next two lemmata.

<u>LEMMA</u> 4.12

$$q \xrightarrow{t} \text{ and } q \in c$$
$$\Rightarrow$$
$$q \in \underline{c}(t)$$

Proof:

$q \xrightarrow{t}$ and Definition 1.8

$\Rightarrow q - I^t \geq 0$

$q \in c$ and Definition 3.8

$$\Rightarrow q_j = c_j \quad \text{for } c_j \in N \text{ and}$$
$$\geq glb(c_j) \quad \text{for } c_j \in N^+$$

$$\Rightarrow q_j = c_j \quad \text{for } c_j \in N \text{ and}$$
$$\geq max(glb(c_j), I^t_j) \quad \text{for } c_j \in N^+$$

Lemma 4.7

$\Rightarrow q \in \underline{c}(t)$

LEMMA 4.13

$$c \xrightarrow{t}, \; q \in \underline{c}(t)$$
$$\Rightarrow$$
$$q \xrightarrow{t}$$

Proof:

$c \xrightarrow{t}$ and Theorem 4.11

$\Rightarrow \mathrm{MIN}(\underline{c}(t)) - I^t \geq 0$

Definition 3.11 and Definition 3.10

$\Rightarrow \underline{c}_j(t) - I_j^t \geq 0$ for $\underline{c}_j(t) \in N$ and

$\mathrm{glb}(\underline{c}_j(t)) - I_j^t \geq 0$ for $\underline{c}_j(t) \in N^+$

$q \in \underline{c}(t)$ and Definition 3.8

$\Rightarrow q_j = \underline{c}_j(t)$ for $\underline{c}_j(t) \in N$ and

$\geq \mathrm{glb}(\underline{c}_j(t))$ for $\underline{c}_j(t) \in N^+$

$\Rightarrow q_j - I_j^t \geq 0$ for $\underline{c}_j(t) \in N$ and

$q_j - I_j^t \geq 0$ for $\underline{c}_j(t) \in N^+$

Definition 1.8

$\Rightarrow q \xrightarrow{t}$

Thus $\underline{c}(t)$ is precisely the set of states in c for which t is enabled.

4.3 COVER

In order to introduce the concept of monotonicity for cubes, it is necessary to introduce an analogue for the state $\geq$ relation. This analog will be called the cover relation.

DEFINITION 4.14 COVER

A cube c' is said to <u>cover</u> another cube c, denoted $c \ll c'$ or $c' \gg c$, if for $j = 1, \ldots, n$:

a) $c_j \leq c'_j$ for $c_j, c'_j \in N$

b) There exist no c_j, c'_j such that $c_j \in N^+$ and $c'_j \in N$.

(It can also be said that c is covered by c'.)

This implies that any values are permitted for c_j and c'_j when either c_j and c'_j are both $\in N^+$ or when $c_j \in N$ and $c'_j \in N^+$. For example, $\langle 7^+,3\rangle \ll \langle 3^+,7\rangle$ and $\langle 7,3\rangle \ll \langle 3^+,7\rangle$.

It furthermore implies that two cubes may cover each other. For example, $\langle 3^+,7^+\rangle \ll \langle 7^+,3^+\rangle$ and $\langle 7^+,3^+\rangle \ll \langle 3^+,7^+\rangle$.

That the cover relation is transitive is now shown.

LEMMA 4.15

$$c^1 << c^2,\ c^2 << c^3$$

$$=>$$

$$c^1 << c^3$$

Proof:

$c^1 << c^2$ and Definition 4.14

=> a) $c_j^1 \leq c_j^2$ for $c_j^1, c_j^2 \in N$

b) There exist no c_j^1, c_j^2 such that $c_j^1 \in N^+$ and $c_j^2 \in N$

$c^2 << c^3$ and Definition 4.14

=> a) $c_j^2 \leq c_j^3$ for $c_j^2, c_j^3 \in N$

b) There exist no c_j^2, c_j^3 such that $c_j^2 \in N^+$ and $c_j^3 \in N$

=> a) $c_j^1 \leq c_j^3$ for $c_j^1, c_j^3 \in N$

b) There exist no c_j^1, c_j^3 such that $c_j^1 \in N^+$ and $c_j^3 \in N$

Furthermore, a cube covers all its subcubes.

LEMMA 4.16

$$c^1 \text{ SUB } c^2$$

$$=>$$

$$c^1 << c^2$$

Proof:

This follows directly from Theorem 3.22 and Definition 4.14

4.4 MONOTONICITY FOR CUBES

Monotonicity for cubes can now be defined. If a transition is enabled for a cube, then it is also enabled for all covering cubes.

LEMMA 4.17 MONOTONICITY FOR CUBES

$$c \xrightarrow{t}, \; c \ll c' \;\Rightarrow\; c' \xrightarrow{t}$$

i.e. if transition t is enabled for some cube c and some other cube c′ covers c, then t is enabled in c′.

Proof:

a) For $c'_j \in N$

Definition 4.14

$\Rightarrow c_j \in N$ and $c_j = c'_j$

$\Rightarrow \underline{c}_j(t) = \underline{c}'_j(t)$

$c \xrightarrow{t}$ and Lemma 4.9

$\Rightarrow \mathrm{MIN}(\underline{c}(t)) - I^t \geq 0$

Definition 3.10 and Definition 3.11

$\Rightarrow \underline{c}_j(t) - I^t_j \geq 0$

$\Rightarrow \underline{c}'_j(t) - I^t_j \geq 0$

b) For $c'_j \in N^+$

Definition 4.4

$\Rightarrow \underline{c}'_j(t) = [\max(\mathrm{glb}(c'_j), I^t_j)]^+$

$\Rightarrow \mathrm{glb}(\underline{c}'_j(t)) \geq I^t_j$

$\Rightarrow \mathrm{glb}(\underline{c}'_j(t)) - I^t_j \geq 0$

a), b) above, Definition 3.10 and Definition 3.11

$\Rightarrow \mathrm{MIN}(\underline{c}'(t)) - I^t \geq 0$

Lemma 4.10

$\Rightarrow c' \xrightarrow{t}$

4.5 TRANSITION FIRING FOR CUBES

4.5.1 Immediate Cube Reachability

The result of firing a transition t for a cube c will be another cube containing the immediate successor for each state in c for which t is enabled (i.e. for each state in $\underline{c}(t)$), and only such states.

DEFINITION 4.18 IMMEDIATE CUBE REACHABILITY

Given a cube c and a transition t, such that $c \xrightarrow{t}$, then the n-tuple c′ (shown to be a cube shortly) defined by:

$$c' = \underline{c}(t) + D^t = \underline{c}(t) - I^t + O^t$$

is said to be immediately cube reachable from c.

This is denoted $c \xrightarrow{t} c'$.

It must now be shown that the n-tuple so defined is in fact a cube.

LEMMA 4.19

For a cube c, a transition t enabled for c, i.e. $c \xrightarrow{t}$, and the n-tuple c′ defined as in Definition 4.18 (i.e. $c \xrightarrow{t} c'$), c′ is a cube.

Proof:

$c \xrightarrow{t} c'$ and Definition 4.18

$\Rightarrow c \xrightarrow{t}$

$c \xrightarrow{t}$ and Lemma 4.9

$\Rightarrow \mathrm{MIN}(\underline{c}(t)) - I^t \geq 0$

Lemma 3.30

$\Rightarrow (\underline{c}(t) - I^t)$ is a cube

all $O^t_j < N$ and Lemma 3.32

$\Rightarrow (\underline{c}(t) - I^t) + O^t$ is a cube

But Definition 4.18

$$\Rightarrow c' = \underline{c}(t) - I^t + O^t$$

Hence c' is a cube.

COROLLARY 4.20

For $c \xrightarrow{t} c'$

$$c_j \in N \iff c'_j \in N$$

and

$$c_j \in N^+ \iff c'_j \in N^+$$

It will now be shown that for $c \xrightarrow{t} c'$, any state in c for which t is enabled, leads to a state which is in c'.

LEMMA 4.21

For $c \xrightarrow{t} c'$,

$$q \in \underline{c}(t)$$

$$\Rightarrow$$

There exists a q' such that $q \xrightarrow{t} q'$ and $q' \in c'$

Proof:

$c \xrightarrow{t} c'$, Definition 4.18 and Lemma 4.13

$\Rightarrow q \xrightarrow{t}$

$q \xrightarrow{t}$, Definition 1.9 and Lemma 1.10

$\Rightarrow$ there exists a state q' such that $q \xrightarrow{t} q'$

and $q' = q - I^t + O^t$

$q \leqslant \underline{c}(t)$ and Definition 3.8

$$\Rightarrow q_j = \underline{c}(t) \qquad \text{for } \underline{c}_j(t) \leqslant N \text{ and}$$
$$\geq glb(\underline{c}_j(t)) \text{ for } \underline{c}_j(t) \leqslant N^+$$

Thus, $q' = q - I^t + 0^t$

$$\Rightarrow q'_j = \underline{c}_j(t) - I^t_j + 0^t_j \qquad \text{for } \underline{c}_j(t) \leqslant N \text{ and}$$
$$\geq glb(\underline{c}_j(t)) - I^t_j + 0^t_j \text{ for } \underline{c}_j(t) \leqslant N^+$$

Lemma 3.29

$$\Rightarrow q'_j = \underline{c}_j(t) - I^t_j + 0^t_j \quad \text{for } \underline{c}_j(t) \leqslant N \text{ and}$$
$$\geq glb(\underline{c}_j(t) - I^t_j + 0^t_j) \text{ for } \underline{c}_j(t) \leqslant N^+$$

Definition 4.18

$$\Rightarrow c' = \underline{c}(t) - I^t + 0^t$$

Definition 3.8

$$\Rightarrow q' \leqslant c'$$

Conversely, for any state in c', its antecedent, with respect to t, is in the enabling cube $\underline{c}(t)$ for any cube c containing q.

<u>LEMMA</u> 4.22

For $c \xrightarrow{t} c'$,

$$q' \leqslant c'$$
$$\Rightarrow$$
$$\text{There exists a } q \text{ such that } q \xrightarrow{t} q' \text{ and } q \leqslant \underline{c}(t)$$

Proof:

Definition 4.18

$$\Rightarrow c' = \underline{c}(t) - I^t + 0^t$$

$q' \leqslant c'$ and Definition 3.8

$$\Rightarrow q'_j = \underline{c}_j(t) - I^t_j + O^t_j \quad \text{for } \underline{c}_j(t) \leqslant N \text{ and}$$
$$\geq \mathrm{glb}(\underline{c}_j(t) - I^t_j + O^t_j) \text{ for } \underline{c}_j(t) \text{ e } N^+$$

Lemma 3.29

$$\Rightarrow q'_j = \underline{c}_j(t) - I^t_j + O^t_j \quad \text{for } \underline{c}_j(t) \leqslant N \text{ and}$$
$$\geq \mathrm{glb}(\underline{c}_j(t)) - I^t_j + O^t_j \text{ for } \underline{c}_j(t) \leqslant N^+$$

Letting $q = q' + I^t - O^t$

$$\Rightarrow q_j = (\underline{c}_j(t) - I^t_j + O^t_j) + I^t_j - O^t_j \quad \text{for } \underline{c}_j(t) \leqslant N \text{ and}$$
$$\geq (\mathrm{glb}(\underline{c}_j(t)) - I^t_j + O^t_j) + I^t_j - O^t_j \text{ for } \underline{c}_j(t) \leqslant N^+$$

$$\Rightarrow q_j = \underline{c}_j(t) \quad \text{for } \underline{c}_j(t) \leqslant N \text{ and}$$
$$\geq \mathrm{glb}(\underline{c}_j(t)) \text{ for } \underline{c}_j(t) \leqslant N^+$$

&myref(1030)

$\Rightarrow$ q is a state

Definition 3.8

$\Rightarrow q \leqslant \underline{c}(t)$

Lemma 4.13

$\Rightarrow q \xrightarrow{t}$

Definition 1.9

$\Rightarrow q \xrightarrow{t} q''$ where $q'' = q - I^t + O^t$

But $q' = q - I^t + O^t$ and Lemma 1.11

$\Rightarrow q' = q''$

$\Rightarrow q \xrightarrow{t} q'$

Thus it has been shown that if $c \xrightarrow{t} c'$, any state in c for which t is enabled leads to a state in c′ and for any state in c′, its antecedent is in the enabling cube for c.

4.5.2 Additional Properties

The concept of immediate antecedent and immediate successor can be extended to cubes.

DEFINITION 4.23 IMMEDIATE ANTECEDENT and IMMEDIATE SUCCESSOR FOR A CUBE

The definitions of immediate antecedent, and immediate successor, first introduced in Definition 1.12 are now extended for cubes. Where $c \xrightarrow{t} c'$, c is said to be an immediate antecedent or an immediate predecessor of c′, and c′ is said to be an immediate successor of c.

Next is given an expression for c′ in terms of its antecedent, rather than its enabling cube.

LEMMA 4.24

$$c \xrightarrow{t} c'$$

$$\Rightarrow$$

$$\begin{aligned} c'_j &= c_j - I^t_j + O^t_j & \text{for } c_j \in N \text{ and} \\ &= [\max(\mathrm{glb}(c_j), I^t_j) - I^t_j + O^t_j]^+ & \text{for } c_j \in N^+ \end{aligned}$$

Proof:

$c \xrightarrow{t} c'$ and Definition 4.18

$$\Rightarrow c' = \underline{c}(t) - I^t + O^t$$

Definition 4.4

$$\begin{aligned}
\Rightarrow \underline{c}_j(t) &= c_j && \text{for } c_j \leqslant N \text{ and} \\
&= [\max(\text{glb}(c_j), I_j^t)]^+ && \text{for } c_j \leqslant N^+ \\
\Rightarrow c' &= c_j - I_j^t + O_j^t && \text{for } c_j \leqslant N \text{ and} \\
&= [\max(\text{glb}(c_j), I_j^t)]^+ - I_j^t + O_j^t && \text{for } c_j \leqslant N^+ \\
\Rightarrow c' &= c_j - I_j^t + O_j^t && \text{for } c_j \leqslant N \text{ and} \\
&= [\max(\text{glb}(c_j), I_j^t) - I_j^t + O_j^t]^+ && \text{for } c_j \leqslant N^+
\end{aligned}$$

Lastly, it is shown that immediate cube reachability implies the existence of at least one state in each of c and c'.

<u>LEMMA</u> 4.25

$$c \xrightarrow{t} c'$$

$$\Rightarrow$$

There exist q, q' such that $q \leqslant c$, $q \xrightarrow{t} q'$ and $q' \leqslant c'$

Proof:

$c \xrightarrow{t} c'$, Definition 4.18 and Lemma 4.19

$$\Rightarrow c \xrightarrow{t} \text{ and } c' = \underline{c}(t) - I^t + O^t$$

Definition 4.1

$$\Rightarrow \text{There exists } q \leqslant c \text{ such that } q \xrightarrow{t}$$

Definition 1.9

$\Rightarrow$ There exists q' such that $q \xrightarrow{t} q'$

and $q' = q - I^t + O^t$

$q \preccurlyeq c$ and Definition 3.8

$$\Rightarrow q_j = c_j \quad \text{for } c_j \in N \text{ and}$$
$$\geq glb(c_j) \quad \text{for } c_j \in N^+$$

$q \xrightarrow{t}$ and Definition 1.8

$$\Rightarrow q - I^t \geq 0$$

$$\Rightarrow q_j \geq I^t_j \text{ for } 1 \leq j \leq n$$

Lemma 2.7

$$\Rightarrow q_j \geq \max(glb(c_j), I^t_j) \text{ for } 1 \leq j \leq n$$

$$\Rightarrow q_j = c_j \quad \text{for } c_j \in N \text{ and}$$
$$\geq \max(glb(c_j), I^t_j) \quad \text{for } c_j \in N^+$$

$q' = q - I^t + O^t$

$$\Rightarrow q'_j = q_j - I^t_j + O^t_j$$

$$\Rightarrow q'_j = c_j - I^t_j + O^t_j \quad \text{for } c_j \in N \text{ and}$$
$$\geq \max(glb(c_j), I^t_j) - I^t_j + O^t_j \quad \text{for } c_j \in N^+$$

Definition 4.4 and $c' = \underline{c}(t) - I^t_j + O^t_j$

$$\Rightarrow c'_j = c_j - I^t_j + O^t_j \quad \text{for } c_j \in N \text{ and}$$
$$= [\max(glb(c_j), I^t_j)]^+ \quad \text{for } c_j \in N^+$$

Definition 3.8

$$\Rightarrow q' \preccurlyeq c'$$

Chapter V

CUBES AND TRANSITION SEQUENCES

The concept of transition sequences for cubes is introduced in this chapter. As well, earlier results for transitions are extended to transition sequences.

5.1 NOTATION

First the transition sequence notation itself is extended. The following definition is the analogue of Definition 1.14.

DEFINITION 5.1 $c^0 \xrightarrow{s} c^k$

For convenience:

$$c^0 \xrightarrow{t^1} c^1, \qquad c^1 \xrightarrow{t^2} c^2, \qquad \ldots, \qquad c^{k-1} \xrightarrow{t^k} c^k$$

is often denoted

$$c^0 \xrightarrow{t^1} c^1 \xrightarrow{t^2} c^2 \xrightarrow{t^3} \ldots\ c^{k-1} \xrightarrow{t^k} c^k$$

and if $s = t^1, \ldots, t^k$, it is also denoted $c^0 \xrightarrow{s} c^k$.

When a change takes place from c^0 to c^k via transition sequence s, then s is said to fire. Corollary 4.20 is now extended for transition sequences yielding the following lemma.

LEMMA 5.2

For $c^0 \xrightarrow{s} c^k$, and $1 \leq i \leq k$,

$$c_j^{i-1} \in N \Leftrightarrow c_j^i \in N$$

and

$$c_j^{i-1} \in N^+ \Leftrightarrow c_j^i \in N^+$$

Proof:

By induction on the length of the transition sequence s utilizing Corollary 4.20 and Definition 5.1

Lemma 4.25 is also extended for transition sequences.

LEMMA 5.3

$$c^0 \xrightarrow{s} c^k$$

$$\Rightarrow$$

$$\text{There exist } q^0, \ldots, q^k \text{ such that } q^0 \xrightarrow{s} q^k,$$

$$\text{and } q^{i-1} \in c^{i-1} \text{ for all } i,\ 1 \leq i \leq k$$

Proof:

By induction on the length of the transition subsequence s(i,k) for i = k,1,-1.

BASE:

Letting q^k be any state in c^k,

Lemma 4.22 and $c^0 \xrightarrow{s} c^k$

$\Rightarrow$ There exists a q^{k-1} such that $q^{k-1} \in \underline{c}^{k-1}(t^k)$

and $q^{k-1} \xrightarrow{t^k} q^k$

Lemma 4.8

$\Rightarrow q^{k-1} \in c^{k-1}$

STEP:

Assuming as the induction hypothesis that $q^i \xrightarrow{s(i+,k)} q^k$ for $q^i \in c^i$, it must be shown that $q^{i-1} \xrightarrow{s(i,k)} q^k$ for $q^{i-1} \in c^{i-1}$. It suffices, in view of the induction hypothesis and Definition 1.14 to show that $q^{i-1} \xrightarrow{t^i} q^i$ for $q^{i-1} \in c^{i-1}$.

Lemma 4.22 and $c^0 \xrightarrow{s} c^k$

$\Rightarrow$ There exists a q^{i-1} such that $q^{i-1} \in \underline{c}^{i-1}(t^i)$

and $q^{i-1} \xrightarrow{t^i} q^i$.

Lemma 4.8

$\Rightarrow q^{i-1} \in c^{i-1}$

5.2 CUBE ENABLING

DEFINITION 5.4 CUBE ENABLED TRANSITION SEQUENCE

A transition sequence s is said to be cube enabled for a cube c if there exists some state q, $q \leqslant c$, such that s is enabled for q i.e. $q \xrightarrow{s}$.

This is denoted $c \xrightarrow{s}$.

It will now be shown that transition sequences can fire for a given cube iff the transition sequence is cube enabled for that cube.

LEMMA 5.5

$$c^0 \xrightarrow{s} c^k$$
$$\Rightarrow$$
$$c^0 \xrightarrow{s}$$

Proof:

$c^0 \xrightarrow{s} c^k$ and Lemma 5.3

$\Rightarrow$ There exist $q^0, \ldots, q^k$ such that $q \xrightarrow{s} q^k$

Definition 2.1

$\Rightarrow q^0 \xrightarrow{s}$

Definition 5.4

$\Rightarrow c^0 \xrightarrow{s}$

<u>LEMMA</u> 5.6

$$c^0 \xrightarrow{s}$$

$$\Rightarrow$$

$$c^0 \xrightarrow{s} c^k, \text{ for some } k,\ k \leqslant N$$

Proof:

$c^0 \xrightarrow{s}$ and Definition 5.4

$\Rightarrow q^0 \xrightarrow{s}$ and $q^0 \leqslant c^0$

Theorem 2.17 and $q \xrightarrow{s}$

$\Rightarrow q^0 - I^s \geq 0$

Theorem 2.16

$\Rightarrow q^0 \xrightarrow{s} q^k$

Definition 1.14

$\Rightarrow q^0 \xrightarrow{t^1} q^1, \quad q^1 \xrightarrow{t^2} q^2, \quad \ldots, \quad q^{k-1} \xrightarrow{t^k} q^k$

where $s = t^1, \ldots, t^k$

It will be shown by induction on the transition sequence length that $c^0 \xrightarrow{s} c^k$.

(i.e. that $c^0 \xrightarrow{t^1} c^1,\ c^1 \xrightarrow{t^2} c^2,\ \ldots,\ c^{k-1} \xrightarrow{t^k} c^k$)

BASE

For transition sequence length of 0 it is necessary to show that c^0 is reachable from c^0, which is vacuously true, and that $q^0 \leqslant c^0$, which was shown above.

STEP

Assuming as the induction hypothesis for transition sequence of length i-1 that c^{i-1} is reachable from c^0

i.e. $c^0 \xrightarrow{t^1} c^1, \ldots, c^{i-2} \xrightarrow{t^{i-1}} c^{i-1}$

and that $q^{i-1} \leqslant c^{i-1}$

i.e. that $q^0 \leqslant c^0, \ldots, q^{i-1} \leqslant c^{i-1}$

where $1 \leq i \leq k$, then it must be shown, for transition sequence of length i, that c^i is reachable from c^{i-1}

i.e. that $c^{i-1} \xrightarrow{t^i} c^i$ and that $q^i \leqslant c^i$.

$q^0 \xrightarrow{s} q^k$ and Definition 1.14

$\Rightarrow q^{i-1} \xrightarrow{t^i} q^i$

Definition 1.9

$\Rightarrow q^{i-1} \xrightarrow{t^i}$

Definition 4.1

$\Rightarrow c^{i-1} \xrightarrow{t^i}$

Definition 4.18

$\Rightarrow c^{i-1} \xrightarrow{t^i} c^i$

the induction hypothesis

$\Rightarrow q^{i-1} \leqslant c^{i-1}$

Lemma 4.12

$\Rightarrow q^{i-1} \leqslant \underline{c}^{i-1}(t^i)$

Lemma 4.21

$\Rightarrow q^i \leqslant c^i$

Summarizing the preceding results yields:

<u>THEOREM</u> 5.7

$$c^0 \xrightarrow{s}$$

$$\Leftrightarrow$$

$$c^0 \xrightarrow{s} c^k$$

Proof:

This follows from Lemma 5.5 and Lemma 5.6

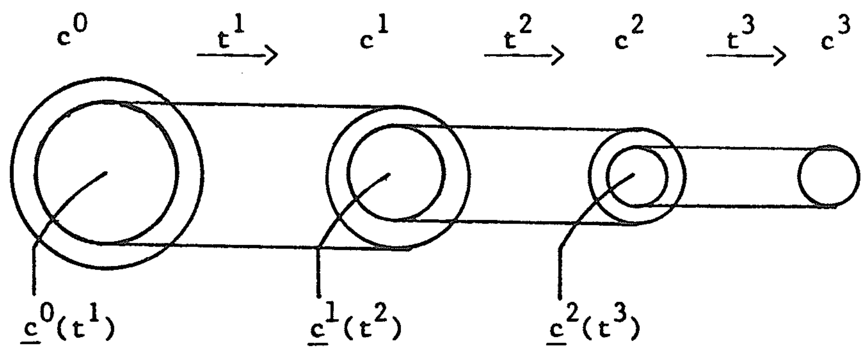
Hence a transition sequence can fire for a given cube iff it is enabled for that cube.

EXAMPLE 5.8

Given 4 cubes, c^0, c^1, c^2 and c^3, and 3 transitions, t^1,t^2 and t^3, such that $c^0 \xrightarrow{s} c^3$ where $s = t^1t^2t^3$. Thus $c^0 \xrightarrow{s} c^3$ which actually represents

$$c^0 \xrightarrow{t^1} c^1 \xrightarrow{t^2} c^2 \xrightarrow{t^3} c^3$$

could be represented graphically as follows, where cubes are circles and subcubes are embedded circles.



In each case the larger circles represent the cubes c^0, c^1, c^2, c^3 respectively. For each of c^0, c^1, c^2 transitions t^1, t^2, t^3, respectively, are enabled, but not necessarily for the entire cube in each case but only for some subset. These subsets of c^0, c^1 and c^2 for which transitions t^1, t^2, t^3, respectively are enabled have been called enabling cubes for transitions (see Definition 4.4) and have been denoted

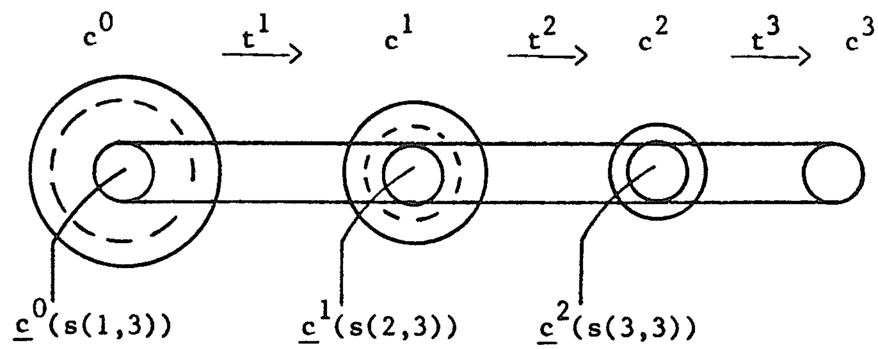
$$\underline{c}^0(t^1),\ \underline{c}^1(t^2) \text{ and } \underline{c}^2(t^3)$$

respectively.

From the diagram it is apparent that any state in c^3 is generated from $\underline{c}^2(t^3)$ by transition t^3. However, the states in $\underline{c}^2(t^3)$ generated by transition t^2 come not only from a subset of c^1 but rather, and more specifically, from a subset of $\underline{c}^1(t^2)$.

This analogy will in fact be shown to be accurate and these subsets will be called enabling cubes for transition subsequences and the subset of $\underline{c}^1(t^2)$ mentioned above will be denoted $\underline{c}^1(s(2,3))$.

The following diagram shows the sequence of subsets of enabling cubes, all the way back to $\underline{c}^0(s(1,3))$, from which c^3 was generated by s.



The subset of c^0 labelled $\underline{c}^0(s(1,3))$ is the subset of c^0 which, via transition sequence s, generated all the states in c^3. Because of its special relationship to transition sequence s and cube c^0, it will also be called an enabling cube, but for transition sequence s, and denoted $\underline{c}^0(s)$. In this example then, because $s = t^1t^2t^3$, $\underline{c}^0(s) = \underline{c}^0(s(1,3))$. Each of the $\underline{c}^i(s(i+1,k))$ in the above diagram is called the enabling cube for the transition subsequence $s(i+1,k)$.

The concept of enabling cubes for transition sequences will be formalized in the next section.

5.3 ENABLING CUBE

The concept of an enabling cube for transitions will be extended to transition sequences.

DEFINITION 5.9 ENABLING CUBE FOR TRANSITION SUBSEQUENCE

Given $s(1,k) = t^1, \ldots, t^k$, the enabling cube $\underline{c}^{i-1}(s(i,k))$ for transition subsequence $s(i,k)$, where $1 \leq i \leq k$, is defined to be

$$\underline{c}^{i-1}(s(i,k)) = \left\langle \underline{c}_j^{i-1}(s(i,k)), \text{ for } j, 1 \leq j \leq n \right\rangle$$

where

$$\begin{aligned} \underline{c}_j^{i-1}(s(i,k)) &= c_j^{i-1} & \text{for } c_j^{i-1} \in N \text{ and} \\ &= [\max(\mathrm{glb}(c_j^{i-1}), I_j^{s(i,k)})]^+ & \text{for } c_j^{i-1} \in N^+ \end{aligned}$$

While Definition 5.9 reads enabling cube for transition subsequences, the fact that the structure so defined is really a cube must be proved. That proof follows.

LEMMA 5.10

$$s = s(i,k) = t^i \ldots t^k$$
$$\Rightarrow$$
$$\underline{c}^{i-1}(s(i,k)) \text{ is a cube}$$
$$\text{for } i\text{: } 1 \leq i \leq k$$

Proof:

It will be shown that each component is a cube component.

For any i

Definition 5.9

$$\begin{aligned} \Rightarrow \underline{c}^{i-1}(s(i,k)) &= c_j^{i-1} & \text{for } c_j^{i-1} \in N \text{ and} \\ &= [\max(\mathrm{glb}(c_j^{i-1}), I_j^{s(i,k)})]^+ & \text{for } c_j^{i-1} \in N^+ \end{aligned}$$

Thus for $c_j^{i-1} \leq N$, $\underline{c}_j^{i-1}(s(i,k))$ certainly has the proper form for a cube component. viz $\underline{c}_j^{i-1}(s(i,k)) \leq N$.

It is necessary only to continue examining the case for $c_j^{i-1} \leq N^+$.

$c_j^{i-1} \leq N^+$ and Definition 3.4

$$\Rightarrow glb(c_j^{i-1}) \leq N$$

Lemma 2.12

$$\Rightarrow I_j^{s(i,k)} \geq 0 \text{ for } j = 1, \ldots, n$$

Definition 2.5

$$\Rightarrow \max(glb(c_j^{i-1}), I_j^{s(i,k)}) \leq N$$

Definition 3.1

$$\Rightarrow [\max(glb(c_j^{i-1}), I_j^{s(i,k)})]^+ \leq N^+$$

Thus for $c_j^{i-1} \leq N^+$, $\underline{c}_j^{i-1}(s(i,k))$ has the proper form for a cube component, viz. $\underline{c}_j^{i-1}(s(i,k)) \leq N^+$.

Hence $\underline{c}^{i-1}(s(i,k))$ is a cube.

COROLLARY 5.11

$$c^0 \xrightarrow{s} c^k$$

$$\text{where } s = s(1,k) = t^1 \ldots t^k$$

$$\Rightarrow$$

$$c_j^{i-1} \leq N \iff \underline{c}_j^{i-1}(s(i,k)) \leq N$$

and

$$c_j^{i-1} \leq N^+ \iff \underline{c}_j^{i-1} \leq N^+$$

Corollary 5.11 is the extension of Corollary 4.6.

The necessary and sufficient conditions for membership of a state in an enabling cube are needed often enough to warrant the following lemma.

LEMMA 5.12

$$q^i \leqslant \underline{c}^i(s(i,k))$$

$$\Longleftrightarrow$$

$$\begin{aligned} q^i_j &= c^i_j && \text{for } c^i_j \leqslant N \text{ and} \\ &\geq \max(\mathrm{glb}(c^i_j), I_j^{s(i+1,k)}) && \text{for } c^i_j \leqslant N^+ \end{aligned}$$

Proof:

This results from application of Definition 3.4 and Definition 3.8 to Definition 5.9

Example 5.8 illustrated the fact that an enabling cube for a transition contains all enabling cubes for transition (sub)sequences starting with that particular transition. The next lemma proves that this is in fact so.

LEMMA 5.13

$$s = s(1,k) = t^1 \ldots t^k$$

$$\Rightarrow$$

$$\underline{c}^{i-1}(s(i,k)) \text{ SUB } \underline{c}^{i-1}(t^i)$$

$$\text{for all } i: 1 \leq i \leq k$$

Proof:

Letting $q^{i-1} \in \underline{c}^{i-1}(s(i,k))$, Lemma 5.12

$$\Rightarrow q_j^{i-1} = c_j^{i-1} \qquad \text{for } c_j^{i-1} \in N \text{ and}$$

$$\geq \max(\text{glb}(c_j^{i-1}), I_j^{s(i,k)}) \text{ for } c_j^{i-1} \in N^+$$

Lemma 2.11

$$\Rightarrow I_j^{s(i,k)} \geq I_j^{t^i}$$

$$\Rightarrow q_j^{i-1} = c_j^{i-1} \qquad \text{for } c_j^{i-1} \in N \text{ and}$$

$$\geq \max(\text{glb}(c_j^{i-1}), I_j^{t^i}) \text{ for } c_j^{i-1} \in N^+$$

Definition 3.8 and Definition 4.4

$$\Rightarrow q^{i-1} \in \underline{c}^{i-1}(t^i)$$

$$\Rightarrow \underline{c}^{i-1}(s(i,k)) \text{ SUB } \underline{c}^{i-1}(t^i)$$

COROLLARY 5.14

$$s = s(1,k) = t^1 \ldots t^k$$

$$\Rightarrow$$

$$\underline{c}^{i-1}(s(i,k)) \text{ SUB } c^{i-1}$$

$$\text{for all } i: 1 \leq i \leq k$$

Corollary 5.14 shows that the enabling cube for a transition (sub)sequence and a given cube is a subcube of the given cube. This is an extension of Lemma 4.8.

Next it will be shown, when a transition sequence consists of a single transition, that the definition of enabling cube for a transition and the definition of enabling cube for a transition sequence yield the same cube.

<u>LEMMA</u> 5.15

$$s = t,\ c \text{ a cube}$$

$$\Rightarrow$$

$$\underline{c}(s) = \underline{c}(t)$$

Proof:

Definition 5.9

$\Rightarrow \underline{c}_j(s) = c_j$ for $c_j \in N$ and

$= [\max(\mathrm{glb}(c_j), I_j^s)]$ for $c_j \in N^+$

Lemma 2.9

$\Rightarrow I^s = I^t$

$\Rightarrow \underline{c}_j(s) = c_j$ for $c_j \in N$ and

$= [\max(\mathrm{glb}(c_j), I_j^t]$ for $c_j \in N^+$

Definition 4.4

$\Rightarrow \underline{c}_j(s) = \underline{c}_j(t)$ for $j = 1, \ldots, n$

$\Rightarrow \underline{c}(s) = \underline{c}(t)$

Next will be shown necessary and sufficient conditions, expressed in terms of $\underline{c}(s)$ and I^s, for s to be enabled for c.

First will be shown a sufficient condition.

LEMMA 5.16

For $s = s(i,k) = t^i \ldots t^k$, and $I^s = I^{s(i,k)}$,

$$\mathrm{MIN}(\underline{c}^{i-1}(s(i,k))) - I^{s(i,k)} \geq 0$$

$$\Rightarrow$$

$$c^{i-1} \xrightarrow{s(i,k)}$$

Proof:

Letting $q = \mathrm{MIN}(\underline{c}^{i-1}(s(i,k)))$, Corollary 5.14

$\Rightarrow q \leq c^{i-1}$

and Lemma 5.12

$$\Rightarrow q_j = c_j^{i-1} \quad \text{for } c_j^{i-1} \in N \text{ and}$$

$$\geq \max(\mathrm{glb}(c_j^{i-1}), I_j^{s(i,k)}) \text{ for } c_j^{i-1} \in N^+$$

$$\Rightarrow q_j - I_j^{s(i,k)} = c_j^{i-1} - I_j^{s(i,k)} \quad \text{for } c_j^{i-1} \in N \text{ and}$$

$$\geq \max(\mathrm{glb}(c_j^{i-1}), I_j^{s(i,k)}) - I_j^{s(i,k)} \text{ for } c_j^{i-1} \in N^+$$

but $\mathrm{MIN}(\underline{c}^{i-1}(s(i,k))) - I^{s(i,k)} \geq 0$ and Definition 5.9

$$\Rightarrow c_j^{i-1} - I_j^{s(i,k)} \geq 0 \quad \text{for } c_j^{i-1} \in N \text{ and}$$

$$\max(\mathrm{glb}(c_j^{i-1}), I_j^{s(i,k)}) - I_j^{s(i,k)} \geq 0 \text{ for } c_j^{i-1} \in N^+$$

$$\Rightarrow q_j - I_j^{s(i,k)} \geq 0 \quad \text{for } j = 1, \ldots, n$$

Theorem 2.17

$\Rightarrow q \xrightarrow{s(i,k)}$

Definition 5.4

$\Rightarrow c^{i-1} \xrightarrow{s(i,k)}$

The next lemma shows a necessary condition.

LEMMA 5.17

For $s(i,k) = t^i \ldots t^k$

$$c^{i-1} \xrightarrow{s(i,k)}$$

$$\Rightarrow$$

$$\text{MIN}(\underline{c}^{i-1}(s(i,k))) - I^{s(i,k)} \geq 0$$

Proof:

Definition 5.9

$$\Rightarrow \underline{c}_j^{i-1}(s(i,k)) = c_j^{i-1} \qquad \text{for } c_j^{i-1} \in N \text{ and}$$
$$= [\max(\text{glb}(c_j^{i-1}), I_j^{s(i,k)})]^+ \quad \text{for } c_j^{i-1} \in N^+$$

Definition 3.4, Definition 3.10 and Definition 3.11

$$\Rightarrow \min(\underline{c}_j^{i-1}(s(i,k))) = c_j^{i-1} \qquad \text{for } c_j^{i-1} \in N \text{ and}$$

$$= \max(\text{glb}(c_j^{i-1}), I_j^{s(i,k)} \quad \text{for } c_j^{i-1} \in N^+$$

$$\Rightarrow \min(\underline{c}_j^{i-1}(s(i,k))) - I_j^{s(i,k)} = c_j^{i-1} - I_j^{s(i,k)} \quad \text{for } c_j^{i-1} \in N \text{ and}$$

$$= \max(\text{glb}(c_j^{i-1}), I_j^{s(i,k)}) - I_j^{s(i,k)} \quad \text{for } c_j^{i-1} \in N^+$$

$c^{i-1} \xrightarrow{s(i,k)}$ and Definition 5.4

$\Rightarrow$ There exists some state q such that

$q \in c^{i-1}$ and $q \xrightarrow{s(i,k)}$.

Theorem 2.17

$$\Rightarrow q - I^{s(i,k)} \geq 0$$

Definition 3.8

$$\Rightarrow c_j^{i-1} - I_j^{s(i,k)} \geq 0 \quad \text{for } c_j^{i-1} \in N$$

Lemma 2.7

$$\Rightarrow \max(\mathrm{glb}(c_j^{i-1}), I_j^{s(i,k)}) - I_j^{s(i,k)} \geq 0 \text{ for } c_j^{i-1} \in N^+$$

The last two results

$$\Rightarrow \mathrm{MIN}(\underline{c}_j^{i-1}(s(i,k))) - I_j^{s(i,k)} \geq 0 \text{ for } j = 1, \ldots, n$$

$$\Rightarrow \mathrm{MIN}(\underline{c}^{i-1}(s(i,k))) - I^{s(i,k)} \geq 0$$

Summarizing the preceding two lemmata yields the following theorem.

THEOREM 5.18

For $s(i,k) = t^i \ldots t^k$,

$$c^{i-1} \xrightarrow{s(i,k)}$$

$$\Longleftrightarrow$$

$$\mathrm{MIN}(\underline{c}^{i-1}(s(i,k))) - I^{s(i,k)} \geq 0$$

Proof:

This is shown in Lemma 5.16 and Lemma 5.17

Next will be shown that, given any transition sequence, if a state is in the enabling cube for that transition sequence, then the state itself is enabled for that transition sequence.

LEMMA 5.19

$$c \xrightarrow{s} \text{ and } q \leqslant \underline{c}(s)$$

$$\Rightarrow$$

$$q \xrightarrow{s}$$

Proof:

$c \xrightarrow{s}$ and Theorem 5.18

$\Rightarrow \mathrm{MIN}(\underline{c}(s)) - I^s \geq 0$

$q \leqslant \underline{c}(s)$, Lemma 5.12 and Definition 3.8

$\Rightarrow q - I^s \geq 0$

It will next be shown that, given any transition sequence, if MIN of a cube is enabled for the given transition sequence, then so is any state in the cube.

<u>LEMMA</u> 5.20

Given a cube c and a transition sequence s, then

$$q = \mathrm{MIN}(c),\ q \xrightarrow{s} \text{ and } q' \leqslant c$$

$$\Rightarrow$$

$$q' \xrightarrow{s}$$

Proof:

$q = \mathrm{MIN}(c)$

$$\Rightarrow q_j = c_j \quad \text{for } c_j \leqslant N \text{ and}$$
$$= \mathrm{glb}(c_j) \text{ for } c_j \leqslant N^+$$

$q' \leqslant c$

$$\Rightarrow q'_j = c_j \quad \text{for } c_j \leqslant N \text{ and}$$
$$\geq \mathrm{glb}(c_j) \text{ for } c_j \leqslant N^+$$
$$\Rightarrow q'_j = q_j \quad \text{for } c_j \leqslant N \text{ and}$$
$$\geq q_j \quad \text{for } c_j \leqslant N^+$$

$q \xrightarrow{s}$

$$\Rightarrow q_j \geq I^s_j \quad \text{for } j = 1, \ldots, n$$
$$\Rightarrow q'_j \geq I^s_j \quad \text{for } j = 1, \ldots, n$$
$$\Rightarrow q' \xrightarrow{s}$$

The following lemma is the transition sequence analogue of Lemma 4.17, cube monotonicity. That is, if a cube is enabled for any transition sequence, then any covering cube is also enabled for that transition sequence.

LEMMA 5.21

$$c \xrightarrow{s}, \ c \ll c'$$
$$\Rightarrow$$
$$c' \xrightarrow{s}$$

Proof:

The proof is analogous to the proof of Lemma 4.17

a) For $c'_j \leqslant N$

Definition 4.14

$\Rightarrow c_j \leqslant N$ and $c_j = c'_j$

$\Rightarrow \underline{c}_j(t) = \underline{c}'_j(t)$

$c \xrightarrow{s}$ and Lemma 5.17

$\Rightarrow \text{MIN}(c_j(s) - I^s \geq 0$

Definition 3.10 and Definition 3.11

$\Rightarrow \underline{c}_j(s) - I_j \geq 0$

$\Rightarrow \underline{c}'_j(s) - I^s_j \geq 0$

b) For $c'_j \leqslant N^+$

Definition 5.9

$\Rightarrow \underline{c}'_j(s) = [\max(\text{glb}(c'_j), I^s_j)]^+$

$\Rightarrow \text{glb}(\underline{c}'_j(s)) \geq I^s_j$

$\Rightarrow \text{glb}(\underline{c}'_j(s)) - I^s_j \geq 0$

a), b) above, Definition 3.10 and Definition 3.11

$\Rightarrow \text{MIN}(\underline{c}'(s)) - I^s \geq 0$

Lemma 5.16

$\Rightarrow c' \xrightarrow{s}$

The following two lemmata show that, as might be expected, the successor cube resulting from the firing of a transition sequence from a given cube is equal to the enabling cube plus the sum of the change vectors which correspond to the transitions constituting the transition sequence.

<u>LEMMA</u> 5.22

$$c^0 \xrightarrow{s} c^k,\ s = s(1,k) = t^1 \dots t^k$$

$$\Rightarrow$$

$$\underline{c}^{k-1}(s(k,k)) = c^k - D^{t^k}$$

Proof:

$c^0 \xrightarrow{s} c^k$ and Definition 5.1

$\Rightarrow c^{k-1} \xrightarrow{t^k} c^k$

Definition 4.18

$\Rightarrow \underline{c}^{k-1}(t^k) = c^k - D^{t^k}$

For $s(k,k) = t^k$, Theorem 5.18

$\Rightarrow \underline{c}^{k-1}(s(k,k)) = \underline{c}^{k-1}(t^k)$

$\Rightarrow c^{k-1}(s(k,k)) = c^k - D^{t^k}$

LEMMA 5.23

$$c^0 \xrightarrow{s} c^k$$

$$\text{where } s = t^1 \ldots t^k = s(1,k)$$

$$\Rightarrow$$

$$\underline{c}^{i-1}(s(i,k)) = c^k - \mathrm{SUM}(b = k,\ i,\ -1)\ D^{t^b}$$

$$\text{for } i:\ 1 \leq i \leq k$$

Proof:

By induction on the transition sequence length.

BASE

For i = k,

Lemma 5.22

$$\Rightarrow \underline{c}^{k-1}(s(k,k)) = c^k - D^{t^k}$$

STEP

Assuming as the induction hypothesis that

$$\underline{c}^{i}(s(i+1,k)) = c^k - \mathrm{SUM}(b = k,\ i+1,\ -1)\ D^{t^b}$$

it will be shown that

$$\underline{c}^{i-1}(s(i,k)) = c^k - \mathrm{SUM}(b = k,\ i,\ -1)\ D^{t^b}$$

i.e. that

$$\underline{c}^{i-1}(s(i,k)) = \underline{c}^{i}(s(i+1,k)) - D^{t^i}$$

$c^{i-1} \xrightarrow{t^i} c^i$ and Definition 4.18

$$\Rightarrow c^i = \underline{c}^{i-1}(t^i) + D^{t^i}$$

Definition 4.4

$$\Rightarrow c_j^i = c_j^{i-1} + D_j^{t^i} \qquad \text{for } c_j^i \leqslant N \text{ and}$$

$$= [\max(\text{glb}(c_j^{i-1}), I_j^{t^i})]^+ + D_j^{t^i} \quad \text{for } c_j^i \leqslant N^+$$

Definition 3.26

$$\Rightarrow c_j^i = c_j^{i-1} + D_j^{t^i} \qquad \text{for } c_j^i \leqslant N \text{ and}$$

$$= [\max(\text{glb}(c_j^{i-1}), I_j^{t^i}) + D_j^{t^i}]^+ \quad \text{for } c_j^i \leqslant N^+$$

Definition 5.9

$$\Rightarrow \underline{c}_j^{i-1}(s(i,k)) = c_j^{i-1} \qquad \text{for } c_j^{i-1} \leqslant N \text{ and}$$

$$= [\max(\text{glb}(c_j^{i-1}), I_j^{s(i,k)})]^+ \quad \text{for } c_j^{i-1} \leqslant N^+$$

Definition 2.8

$$\Rightarrow \underline{c}_j^{i-1}(s(i,k)) = c_j^{i-1} \qquad \text{for } c_j^{i-1} \leqslant N \text{ and}$$

$$= [\max(\text{glb}(c_j^{i-1}), \max(I_j^{t^i}, I_j^{s(i+1,k)} - D_j^{t^i}))]^+ \quad \text{for } c_j^{i-1} \leqslant N^+$$

Definition 5.9

$$\Rightarrow \underline{c}_j^{i}(s(i+1,k)) = c_j^{i} \qquad \text{for } c_j^{i} \leqslant N \text{ and}$$

$$= [\ \max(\text{glb}(c_j^{i}), I_j^{s(i+1,k)})]^+ \quad \text{for } c_j^{i} \leqslant N^+$$

$$\Rightarrow \underline{c}_j^i(s(i+1,k)) = c_j^{i-1} + D_j^{t^i} \qquad \text{for } c_j^{i-1} \leqslant N \text{ and}$$

$$= [\max(\mathrm{glb}([\max(\mathrm{glb}(c_j^{i-1}), I_j^{t^i}) + D_j^{t^i}]^+), I_j^{s(i+1,k)})]^+ \text{ for } c_j^{i-1} \leqslant N^+$$

$$\Rightarrow \underline{c}_j^i(s(i+1,k)) = c_j^{i-1} + D_j^{t^i} \qquad \text{for } c_j^{i-1} \leqslant N \text{ and}$$

$$= [\max(\max(\mathrm{glb}(c_j^{i-1}), I_j^{t^i}) + D_j^{t^i}, I_j^{s(i+1,k)})]^+ \text{ for } c_j^{i-1} \leqslant N^+$$

$$\Rightarrow \underline{c}_j^i(s(i+1,k)) - D_j^{t^i} = c_j^{i-1} + D_j^{t^i} - D_j^{t^i} \qquad \text{for } c_j^{i-1} \leqslant N \text{ and}$$

$$= [\max(\max(\mathrm{glb}(c_j^{i-1}), I_j^{t^i}) + D_j^{t^i}, I_j^{s(i+1,k)})]^+ - D_j^{t^i} \text{ for } c_j^{i-1} \leqslant N^+$$

$$\Rightarrow \underline{c}_j^i(s(i+1,k)) = c_j^{i-1} \qquad \text{for } c_j^{i-1} \leqslant N \text{ and}$$

$$= [\max(\max(\mathrm{glb}(c_j^{i-1}), I_j^{t^i}) + D_j^{t^i}, I_j^{s(i+1,k)}) - D_j^{t^i}]^+ \text{ for } c_j^{i-1} \leqslant N^+$$

$$\Rightarrow \underline{c}_j^i(s(i+1,k)) = c_j^{i-1} \qquad \text{for } c_j^{i-1} \leqslant N \text{ and}$$

$$= [\max(\max(\mathrm{glb}(c_j^{i-1}), I_j^{t^i}) + D_j^{t^i} - D_j^{t^i}, I_j^{s(i+1,k)} - D_j^{t^i})]^+ \text{ for } c_j \leqslant N^+$$

$$\Rightarrow \underline{c}_j^i(s(i+1,k)) = c_j^{i-1} \qquad \text{for } c_j^{i-1} \leqslant N \text{ and}$$

$$= [\max(\max(\mathrm{glb}(c_j^{i-1}), I_j^{t^i}), I_j^{s(i+1,k)} - D_j^{t^i})]^+ \text{ for } c_j^{i-1} \leqslant N^+$$

Lemma 2.6

$$\Rightarrow \underline{c}_j^{i-1}(s(i,k)) = \underline{c}_j^i(s(i+1,k)) - D_j^{t^i}$$

<u>COROLLARY</u> 5.24

$$c^0 \xrightarrow{s} c^k \text{ where } s = t^1 \dots t^k$$

$$\Rightarrow$$

$$\underline{c}^{i-1}(s(i,k)) = \underline{c}^i(s(i+1,k)) - D^{t^i}$$

COROLLARY 5.25

$$c^0 \xrightarrow{s} c^k \text{ where } s = t^1 \ldots t^k$$

$$\Rightarrow$$

$$\underline{c}^0(s) = \underline{c}^k - \text{SUM}(i=1,k,1)\ D^{t^i}$$

$$= c^k - D^s$$

The last result shown in this section will be that for $c^0 \xrightarrow{s} c^k$ and $q^0 \xrightarrow{s} q^k$,

$$q^0 \leqslant \underline{c}(s)$$

$$\Leftrightarrow$$

$$q^k \leqslant c^k$$

The following lemma will be required.

LEMMA 5.26

For $c^0 \xrightarrow{s} c^k$, $q^0 \xrightarrow{s} q^k$ and $s = s(1,k) = t^1 \ldots t^k$,

$$q^0 \leqslant \underline{c}^0(s)$$

$$\Rightarrow$$

$$q^i \leqslant \underline{c}^i(s(i+1,k)) \text{ for all } i\text{: } 1 \leq i < k$$

Proof:

This proof will be by induction upon the length of the transition subsequence $s(i,k)$ for $i = 1,k-1,1$.

BASE

For $i = 1$, $s = s(1,k)$,

$q^0 \leqslant \underline{c}^0(s)$ and Definition 1.13

$\Rightarrow q^0 \leqslant \underline{c}^0(s(1,k))$

STEP

It is assumed as the induction hypothesis that

$$q^{i-1} \leq \underline{c}^{i-1}(s(i,k)).$$

It must be shown that

$$q^{i} \leq \underline{c}^{i}(s(i+1,k)).$$

$q^{i-1} \leq \underline{c}^{i-1}(s(i,k))$, Definition 5.9 and Definition 3.8

$$\Rightarrow q_j^{i-1} = c_j^{i-1} \qquad \text{for } c_j \leq N \text{ and}$$

$$\geq \text{glb}([\max(\text{glb}(c_j^{i-1}), I_j^{s(i,k)})]^+) \text{ for } c_j^{i-1} \leq N^+$$

Lemma 3.6

$$\Rightarrow q_j^{i-1} = c_j^{i-1} \qquad \text{for } c_j \leq N \text{ and}$$

$$\geq \max(\text{glb}(c_j^{i-1}),\ I_j^{s(i,k)}) \text{ for } c_j^{i-1} \leq N^+$$

$q^{i-1} \xrightarrow{t^i}$ and Definition 1.9

$$\Rightarrow q^i = q^{i-1} - I^{t^i} + O^{t^i} = q^{i-1} + D^{t^i}$$

$$\Rightarrow q_j^i = c_j^{i-1} - I_j^{t^i} + O_j^{t^i} \qquad \text{for } c_j^{i-1} \leq N \text{ and}$$

$$\geq \max(\text{glb}(c_j^{i-1}),\ I_j^{s(i,k)}) - I_j^{t^i} + O_j^{t^i} \text{ for } c_j^{i-1} \leq N^+$$

Lemma 2.11

$$\Rightarrow I_j^{s(i,k)} \geq I_j^{t^i}$$

$$\Rightarrow \max(glb(c_j^{i-1}), I_j^{s(i,k)}) - I_j^{t^i} + O_j^{t^i} \geq \max(glb(c_j^{i-1}), I_j^{t^i}) - I_j^{t^i} + O_j^{t^i}$$

$$\Rightarrow q_j^i = c_j^{i-1} - I_j^{t^i} + O_j^{t^i} \quad \text{for } c_j^{i-1} \in N \text{ and}$$

$$\geq \max(glb(c_j^{i-1}), I_j^{t^i}) - I_j^{t^i} + O_j^{t^i} \quad \text{for } c_j^{i-1} \in N^+$$

$q^i \xrightarrow{s(i+1,k)} q^k$ and Lemma 2.14

$$\Rightarrow q^i \geq I^{s(i+1,k)} \quad \text{for } j = 1, \ldots, n$$

$$\Rightarrow q_j^i = c_j^{i-1} - I_j^{t^i} + O_j^{t^i} \quad \text{for } c_j^{i-1} \in N \text{ and}$$

$$\geq \max(\max(glb(c_j^{i-1}), I_j^{t^i}) - I_j^{t^i} + O_j^{t^i}, I_j^{s(i+1,k)}) \quad \text{for } c_j^{i-1} \in N^+$$

But Lemma 4.24

$$\Rightarrow c_j^i = c_j^{i-1} - I_j^{t^i} + O_j^{t^i} \quad \text{for } c_j^{i-1} \in N \text{ and}$$

$$= [\max(glb(c_j^{i-1}), I_j^{t^i}) - I_j^{t^i} + O_j^{t^i}]^+ \quad \text{for } c_j^{i-1} \in N^+$$

$$\Rightarrow q_j^i = c_j^i \quad \text{for } c_j^i \in N \text{ and}$$

$$\geq \max(glb(c_j^i), I_j^{s(i+1,k)}) \quad \text{for } c_j^i \in N^+$$

Thus Definition 5.9

$$\Rightarrow q^i \in \underline{c}^i(s(i+1,k))$$

The sufficient condition can now be shown.

<u>LEMMA</u> 5.27

For $c^0 \xrightarrow{s} c^k$, $q^0 \xrightarrow{s} q^k$,

$$q^0 \leqslant \underline{c}(s)$$

$$\Rightarrow$$

$$q^k \leqslant c^k$$

Proof:

Lemma 5.26

$\Rightarrow q^{k-1} \leqslant \underline{c}^{k-1}(s(k,k))$

Lemma 5.13

$\Rightarrow \underline{c}^{k-1}(s(k,k)) = \underline{c}^{k-1}(t^k)$

$\Rightarrow q^{k-1} \leqslant \underline{c}^{k-1}(t^k)$

Lemma 4.22 or Lemma 4.21

$\Rightarrow q^k \leqslant c^k$

The necessary condition will be shown next.

THe following lemma will be needed to do so.

LEMMA 5.28

For $c^0 \xrightarrow{s} c^k$, $q^0 \xrightarrow{s} q^k$,

$$q^k \leqslant c^k$$

$$\Rightarrow$$

$$q^{k-1} \leqslant \underline{c}^{k-1}(s(k,k))$$

Proof:

$q^{k-1} \xrightarrow{t^k} q^k$, $c^{k-1} \xrightarrow{t^k} c^k$ and Lemma 4.22

$\Rightarrow q^{k-1} \leqslant \underline{c}^{k-1}(t^k)$

Lemma 5.13

$\Rightarrow \underline{c}^{k-1}(t^k) = \underline{c}^{k-1}(s(k,k))$

$\Rightarrow q^{k-1} \leqslant \underline{c}^{k-1}(s(k,k))$

LEMMA 5.29

For $c^0 \xrightarrow{s} c^k$, $q^0 \xrightarrow{s} q^k$,

$$q^k \leqslant c^k$$

$$\Rightarrow$$

$$q^{i-1} \leqslant \underline{c}^{i-1}(s(i,k)) \text{ for all } i: 1 \leq i \leq k$$

Proof:

By induction on the length of the transition subsequence $s(i,k)$ where $i = k, 1, -1$.

BASE

Letting $i = k$,

Lemma 5.28

$$\Rightarrow q^{k-1} \leq \underline{c}^{k-1}(s(k,k))$$

STEP

Assuming as the induction hypothesis that $q^i \leq \underline{c}^i(s(i+1,k))$ it will be shown that $q^{i-1} \leq \underline{c}^{i-1}(s(i,k))$.

$$q^{i-1} \xrightarrow{t^i} q^i$$

$$\Rightarrow q^i = q^{i-1} - I^{t^i} + O^{t^i}$$

$$\Rightarrow q^{i-1} = q^i + I^{t^i} - O^{t^i}$$

$q^i \leq \underline{c}^i(s(i+1,k))$ and Definition 5.9

$$\Rightarrow q^i_j = c^i_j \qquad \text{for } c^i_j \in N \text{ and}$$

$$\geq \max(glb(c^i_j), I^{s(i+1,k)}_j) \qquad \text{for } c^i_j \in N^+$$

$$\Rightarrow q^i_j = c^i_j + I^{t^i}_j - O^{t^i}_j \qquad \text{for } c^i_j \in N \text{ and}$$

$$\geq \max(glb(c^i_j), I^{s(i+1,k)}_j) + I^{t^i}_j - O^{t^i}_j \qquad \text{for } c^i_j \in N^+$$

$c^{i-1} \xrightarrow{t^i} c^i$ and Lemma 4.24

$$\Rightarrow c_j^i = c_j^{i-1} - I_j^{t^i} + O_j^{t^i} \qquad \text{for } c_j^{i-1} \in N \text{ and}$$

$$= [\max(\text{glb}(c_j^{i-1}), I_j^{t^i}) - I_j^{t^i} + O_j^{t^i}]^+ \text{ for } c_j^{i-1} \in N^+$$

$$\Rightarrow q_j^{i-1} = c_j^{i-1} - I_j^{t^i} + O_j^{t^i} + I_j^{t^i} - O_j^{t^i} \qquad \text{for } c_j^{i-1} \in N \text{ and}$$

$$\geq \max(\max(\text{glb}(c_j^{i-1}), I_j^{t^i}) - I_j^{t^i} + O_j^{t^i},\ \ I_j^{s(i+1,k)}) \ + I_j^{t^i} - O_j^{t^i}$$

for $c_j^{i-1} \in N^+$

$$\Rightarrow q_j^{i-1} = c_j^{i-1} \qquad \text{for } c_j^{i-1} \in N \text{ and}$$

$$\geq \max(\max(\text{glb}(c_j^{i-1}), I_j^{t^i}),\ \ I_j^{s(i+1,k)} + I_j^{t^i} - O_j^{t^i}) \text{ for } c_j^{i-1} \in N^+$$

$$\Rightarrow q_j^{i-1} = c_j^{i-1} \qquad \text{for } c_j^{i-1} \in N \text{ and}$$

$$\geq \max(\text{glb}(c_j^{i-1}), I_j^{t^i}) \text{ for } c_j^{i-1} \in N^+$$

$$\Rightarrow q_j^{i-1} = c_j^{i-1} \qquad \text{for } c_j^{i-1} \in N \text{ and}$$

$$\geq \text{glb}(c_j^{i-1}) \qquad \text{for } c_j^{i-1} \in N^+$$

$q^{i-1} \xrightarrow{s(i,k)} q^k$ and Theorem 2.16

$$\Rightarrow q_j^{i-1} \geq I_j^{s(i,k)} \text{ for } j = 1, \ldots, n$$

$$\Rightarrow q_j^{i-1} = c_j^{i-1} \qquad \text{for } c_j^{i-1} \in N \text{ and}$$

$$\geq \max(\text{glb}(c_j^{i-1}), I_j^{s(i,k)}) \text{ for } c_j^{i-1} \in N^+$$

Definition 5.9

$$\Rightarrow q^{i-1} \in \underline{c}^{i-1}(s(i,k))$$

Summarizing the preceding two lemmata yields the following theorem.

<u>THEOREM</u> 5.30

For $c^0 \xrightarrow{s} c^k$, $q^0 \xrightarrow{s} q^k$,

$$q^0 \leqslant \underline{c}^0(s)$$

$$\Leftrightarrow$$

$$q^k \leqslant c^k$$

Proof:

This follows from Lemma 5.27 and Lemma 5.29

Chapter VI

REACHABILITY TREE

6.1 INTRODUCTION

Karp and Miller [Karp & Miller 1968] developed an algorithm which shows explicitly which places are bounded and which are not for a given Petri Net and some given initial state. Hack [Hack 1975] modified the notation somewhat and called the construct produced by this algorithm a Coverability Tree. He also showed that it could be used to determine whether some arbitrary state can be covered by a reachable state. This algorithm cannot however in general show whether this arbitrary state is itself reachable.

The cube notation introduced in the preceding chapter permits this Coverability Tree construction algorithm to be modified so that the reachability of a given state from the initial state can be determined for some Vector Replacement Systems. The resulting construct is called the Reachability Tree. The Reachability Tree for a given Vector Replacement System, VRS, is denoted RT_{VRS}.

It will be shown that a state is in a cube or node in the Reachability Tree if and only if it is in the Reachability Set, provided only that the Reachability Tree construction algorithm terminates normally.

A comparison with the Coverability Tree Construction Algorithm is provided in the next chapter.

6.2 REACHABILITY TREE CONSTRUCTION

The Reachability Tree construction algorithm creates a tree where each node is a cube. The root node is the cube containing only the initial state.

The nodes may be labelled as follows:

a) All nodes are labelled frontier nodes when generated. Upon further processing, nodes will be relabelled as either internal nodes, loop nodes or null nodes. If the algorithm terminates normally, all frontier nodes will have been so relabelled.

b) A node which is not enabled for any transition will have no successors. Such a node will be called a null node.

c) A node which is a subcube of an antecedent on the same branch of the tree will be called a loop node. It will have no successors but will have a pointer to its containing antecedent.

d) All other nodes will be called internal nodes. These nodes all have successors.

The term leaf node is sometimes used as the opposite of internal node and stands for any node with no successors, i.e. both loop and null nodes.

Processing starts with the initial or root node, i.e. the cube consisting only of the initial state, and continues, one frontier node at a time, as long as there are frontier nodes left.

If at any time, the embedded conditions are violated, the algorithm terminates abnormally and the Reachability Tree cannot be constructed. Hence Reachability cannot be determined by this method in that event.

These embedded conditions have been selected so as to make possible the proof that the Reachability Set equals the set of states represented by the Reachability Tree.

Each frontier node is processed in turn. If there exists no transition which is enabled for it then it is labelled a null node.

If it is not null, then it is compared to its antecedents to determine if it is a subcube of one or more of them. If so it is labelled a loop node. In order to record the particular antecedent which contains it, a loop backpointer is established from the new loop node back to this antecedent.

If the node is not a leaf node, that is neither loop nor null, then its successors will be generated.

The first step in generating its successor is to determine its immediate successor. i.e. the cube resulting from the firing of the enabled transition. If the current node is c^k then its immediate successor will be denoted c' and the succesor of c^k will be denoted c^{k+1}. c^{k+1} is initially set equal to c'.

The concept of potential $^+$-component will now be introduced before proceeding.

DEFINITION 6.1 POTENTIAL $^{+}$-COMPONENT

For some $c^a, c' \in RT_{VRS}$ such that c^a is an antecedent of c' and

$$c^a << c'$$

and there exists c^a_j and c'_j such that c^a_j, $c'_j \in N$ and $c'_j > c^a_j$

then c'_j is called a potential $^{+}$-component with respect to c^a.

All antecedents of c' are checked to see if any components of c' are potential $^{+}$-components with respect to any antecedent. If there are none, then processing of this particular successor of c^k is finished.

If there do exist potential $^{+}$-components with respect to one or more antecedents then c^{k+1} is processed further with respect to each such antecedent.

If for a given antecedent all potential $^{+}$-components in c' have already become $^{+}$-components in c^{k+1} then no further action is required with respect to the given antecedent. Otherwise an attempt is made to generate a new $^{+}$-component in c^{k+1} corresponding to the given antecedent.

Before this can be done, it is necessary to find a transition sequence that will change the particular potential $^{+}$-component by 1 every time it is fired. The transition sequence which takes some state in the antecedent to the state MIN(c') proves to be sufficient for this purpose. Its change vector is computed and then examined.

If any of the conditions required of the change vector do not hold, then the algorithm terminates abnormally.

If they do hold then a $^+$-backpointer is established linking c^{k+1} with the given antecedent, and the given component of c^{k+1} is changed to a $^+$-component whose greatest lower bound is its former value. Processing then proceeds with the next potential $^+$-component.

This algorithm is now presented formally, in modular, structured pseudo-code.

<u>DEFINITION</u> 6.2 REACHABILITY TREE

Given a vector replacement system, VRS = $[T, q^{init}]$, its <u>reachability tree</u> is defined as the set of cubes (the nodes in the tree) constructed as follows:

<u>let</u> the root node in the tree, c^{root}, contain only the initial state q^{init}.

i.e. <u>let</u> $c^{root} = \{q^{init}\}$

<u>label</u> c^{root} as a <u>frontier</u> <u>node</u> for the present.

<u>begin</u>

    <u>while</u> there exist any nodes in the reachability tree currently labelled as frontier nodes <u>do</u>

        <u>perform</u> PROCESS A FRONTIER NODE

    <u>end</u> <u>while</u>

    tree construction terminates normally

<u>end</u>

begin PROCESS A FRONTIER NODE

select a node currently labelled as a frontier node, say c^k

if there exists some transition t such that $c^k \xrightarrow{t}$, then do

if there exists some antecedent of c^k, say c^a, such that c^k SUB c^a then do

label c^k as a loop node

establish a loop back-pointer from c^k to c^a and label it L.

else do

for each t such that $c^k \xrightarrow{t}$ do

perform GENERATE SUCCESSOR NODE

end for

end if

else do

label c^k as a null node (i.e. a dead end)

end if

end PROCESS A FRONTIER NODE

begin GENERATE SUCCESSOR NODE

compute the enabling cube $\underline{c}^k(t)$

label the immediate successor of c^k via transition t to be c'. Thus $c^k \xrightarrow{t} c'$.

let $c' := \underline{c}^k(t) - I^t + 0^t$.

let $c^{k+1} := c'$ for the time being. c^{k+1} will become the successor node of c^k in the reachability tree.

let $h := 0$ (h will accumulate the number of $^+$-backpointers created for c^{k+1}.)

for each $c^a \in RT_{VRS}$ such that c^a is an antecedent of c^{k+1} for a going from k to 0 in steps of -1 (i.e. regressing along the path from c^{root} to c^k starting at c^k) do

if $c^a \ll c'$ and there exists at least one j such that $c'_j \in N$, $c^a_j \in N$ and $c'_j > c^a_j$ (such c'_j are called potential $^+$-components) then do

perform TEST NODE FOR FURTHER ACTION

end if

end for

let $b^{k+1} := h$

end GENERATE SUCCESSOR NODE

begin TEST NODE FOR FURTHER ACTION

  if for all c'_j which are potential $^+$-components with respect to the current c^a, the corresponding c_j^{k+1} are already $\leqslant N^+$,

  then do

    no further action is required

  else do

    perform GENERATE NEW $^+$-COMPONENT

  end if

end TEST NODE FOR FURTHER ACTION

begin GENERATE NEW $^+$-COMPONENT

  perform CALCULATE $\hat{D}^{a^h,k+1}$

  if for $j = 1, \ldots, n$

    a) there is only one j, say $j = m_h$ such that

      $c_j^{a^h} \leqslant N$, $c'_j \leqslant N$ and $\hat{D}^{a^h,k+1} = 1$ and

    b) for all other j where $c_j^{a^h}$, $c'_j \leqslant N$, $\hat{D}_j^{a^h,k+1} = 0$

    c) for all other j (i.e. where $c'_j \leqslant N^+$), $\hat{D}_j^{a^h,k+1} \leq 0$

  then do

    establish a $^+$-back-pointer from c^{k+1} to c^{a^h}

    let h := h + 1

      $c_{m_h}^{k+1} := [c'_{m_h}]^+$

    (c_j^{k+1} remains unchanged for all other j: $j \neq m_h$)

  else do

    the algorithm terminates abnormally

  end if

end GENERATE NEW $^+$-COMPONENT

begin CALCULATE $\hat{D}^{a^h,k+1}$

let $q^{k+1} := \mathrm{MIN}(c^{k+1}) = \mathrm{MIN}(c')$

let $q^{k,0} := q^{k+1} - D^{t^{k+1}}$

$s^{k,0} := t^{k+1}$

$i := k$

while $i \geq a + 1$ do

let b' := number of $^+$-backpointers associated with c^i.

if $b' > 0$ then do

let $c^i_{m_1}, \ldots, c^i_{m_{b'}}$ be the $^+$-components associated with each backpointer.

for $h' := 1,b',1$ do

It is now known that $c^i_{m_{h'}}$ is the $^+$-component corresponding to the h'th $^+$-backpointer. Letting $c^{a^{h'}}$ be the cube to which this $^+$-backpointer points implies that

$\hat{s}^{a^{h'},i}$ is the corresponding transition sequence

and

$\hat{D}^{a^{h'},i}$ is the corresponding change vector.

let $R := q^{i,h'-1}_{m_{h'}} - \mathrm{glb}(c^i_{m_{h'}})$

$q^{i,h'} := q^{i,h'-1} - R\hat{D}^{a^{h'},i}$

$s^{i,h'} := R\hat{s}^{a^{h'},i} s^{i,h'-1}$

let $h' := h' + 1$

end for

end if

$\underline{\text{let}}\ q^{i} := q^{i,b'}$

$s^{i} := s^{i,b'}$

$\underline{\text{let}}\ q^{i-1} := q^{i} - D^{t^{a^{i}}}$

$s^{i-1} := t^{i}s^{i}$

$\underline{\text{let}}\ i := i-1$

$\underline{\text{end}}\ \underline{\text{while}}$

$\underline{\text{let}}\ \hat{s}^{a^{h},k+1} := s^{a}$

$\hat{D}^{a^{h},k+1} := q^{k+1} - q^{a}$

$\underline{\text{end}}$ CALCULATE $\hat{s}^{a^{h},k+1}$ and $\hat{D}^{a^{h},k+1}$

The following notation is introduced for convenience.

$\underline{\text{DEFINITION}}$ 6.3 $c^{i-1} \xrightarrow{t^{i}} c^{i}$

For c^{i-1}, $c^{i} \in RT_{VRS}$, the generation by Definition 6.2 of c^{i} from c^{i-1} via transition t^{i} is denoted

$$c^{i-1} \xrightarrow{t^{i}} c^{i},$$

and for $s = s(1,k) = t^{1} \ldots t^{k}$

$$c^{0} \xrightarrow{t^{1}} c^{1} \xrightarrow{t^{2}} \ldots \xrightarrow{t^{k}} c^{k}$$

is denoted $c^{0} \xrightarrow{s} c^{k}$.

Thus if c^{0} is the root node in the reachability tree then $c^{0} \xrightarrow{s} c^{k}$ means that $c^{k} \in RT_{VRS}$. It should be noted that no assumption is made as to reachability of c^{k}, despite the resemblence to the various reachability notations.

It should also be noted that $c^{0} \xrightarrow{s} c^{0}$ where s is the null transition sequence will be required later.

6.3 FINITENESS OF REACHABILITY TREE

For the reachability tree to be useful in determining reachability it must be finite (i.e. the algorithm must terminate). The approach taken here to show this is essentially that of [Hack 1975] extended again to cubes.

First will be defined non-decreasing sequences of integers, $^+$-components and cubes. Then it will be shown that every infinite subsequence of integers, $^+$-components and cubes contains a non-decreasing infinite subsequence of integers, $^+$-components and cubes respectively.

König's Infinity Lemma, for infinite rooted trees will be stated and then used to show that the Reachability Tree is finite and hence can be constructed.

DEFINITION 6.4 NON-DECREASING SEQUENCE OF INTEGERS

A sequence of integers

$$z^1, z^2, \ldots, z^i, z^{i+1}, \ldots$$

is said to be non-decreasing if $z^{i+1} \geq z^i$ for all i.

DEFINITION 6.5 NON-DECREASING SEQUENCE OF $^+$-COMPONENTS

A sequence of $^+$-components,

$$[k^1]^+, [k^2]^+, \ldots, [k^i]^+, [k^{i+1}]^+, \ldots$$

is said to be non-decreasing if the sequence of integers

$$\mathrm{glb}([k^1]^+), \mathrm{glb}([k^2]^+), \ldots, \mathrm{glb}([k^i]^+), \mathrm{glb}([k^{i+1}]^+), \ldots$$

is non-decreasing.

DEFINITION 6.6 NON-DECREASING SEQUENCE OF CUBES

A sequence of cubes is said to be non-decreasing if, for all j:

a) for all k, $k \leqslant N$ and a given j, all $c_j^k \leqslant N$ or all $c_j^k \leqslant N^+$ and

b) for c^k, $c^{k'}$ any two elements of the sequence such that c^k precedes $c^{k'}$:

$$\text{for } c_j^k, c_j^{k'} \leqslant N, \qquad c_j^{k'} \geq c_j^k$$
$$\text{for } c_j^k, c_j^{k'} \leqslant N^+, \qquad \mathrm{glb}(c_j^{k'}) \geq \mathrm{glb}(c_j^k)$$

LEMMA 6.7 [Hack 1975]

Every infinite sequence of non-negative integers contains an infinite non-decreasing subsequence.

LEMMA 6.8

Every infinite sequence of $^+$-components (cube components $\leqslant N^+$) contains an infinite non-decreasing sub-sequence.

Proof:

A sequence of $^+$-components (cube components $\leqslant N^+$),

$$[k^1]^+, [k^2]^+, \ldots, [k^i]^+, [k^{i+1}]^+, \ldots$$

has corresponding to it a sequence of non-negative integers,

$$\mathrm{glb}([k^1]^+), \mathrm{glb}([k^2]^+), \ldots, \mathrm{glb}([k^i]^+), \mathrm{glb}([k^{i+1}]^+), \ldots$$

This sequence then has an infinite non-decreasing subsequence by Lemma 6.7. Selecting the subsequence of cube components from the original sequence of cube components corresponding to this subsequence of integers yields, by Definition 6.5 the desired non-decreasing subsequence.

LEMMA 6.9

Every infinite sequence of cubes contains an infinite non-decreasing subsequence.

Proof:

Consider the first component. If there are infinitely many cubes whose first component is $\leqslant N$ then by Lemma 6.7 there exists an infinite subsequence of cubes whose first component is non-decreasing.

Otherwise there must be infinitely many cubes whose first components are $\leqslant N^+$. By Lemma 6.8 there then exists an infinite subsequence of cubes whose first component is non-decreasing.

This infinite subsequence now contains another infinite subsequence non-decreasing in its second component (again by Lemma 6.7 or Lemma 6.8) and so on to the n^{th} component.

Thus there exists an infinite subsequence non-decreasing in each component.

It will now be shown that

a) The reachability tree, if it is to be infinite, must have an infinite path leading away from the origin (c^{root}).

and that

b) the tree construction precludes such a path.

Therefore the tree is finite and can be constructed.

LEMMA 6.10 KÖNIG'S INFINITY LEMMA [König 1936, Hack 1975]

An infinite rooted tree wherein each node has only a finite number of immediate successors must have an infinite path.

THEOREM 6.11

The Reachability Tree is finite and hence can be constructed.

Proof:

Suppose the reachability tree RT_{VRS} for some vector replacement system is infinite.

By construction every node has at most as many immediate successors as there are transitions. Hence by König's Infinity Lemma for a rooted tree (Lemma 6.10) there must be an infinite path in the tree. That is a path which does not eventually end in a leaf node (a null node or a loop node).

By Lemma 6.9 there must be an infinite non-decreasing subsequence of the sequence of cubes along this infinite path.

Thus for all c^k in such an infinite, non-decreasing subsequence,

if some $c_j^k \in N$ then all $c_j^k \in N$ for that j, over all k and
if some $c_j^k \in N^+$ then all $c_j^k \in N^+$ for that j, over all k.

For any two cubes c^k and $c^{k'}$ in the infinite, non-decreasing subsequence such that $c^{k'}$ comes after c^k,

$$c_j^{k'} \geq c_j^k \qquad \text{for } c_j^k, c_j^{k'} \in N \text{ or}$$
$$\text{glb}(c_j^{k'}) \geq \text{glb}(c_j^k) \qquad \text{for } c_j^k, c_j^{k'} \in N^+.$$

There are three possibilities:

a) there exist no $c_j^{k'}$, c_j^k for c_j^k, $c_j^{k'} \in N$ and

$glb(c_j^{k'}) \geq glb(c^k)$ for c_j^k, $c_j^{k'} \in N^+$

b) all $c_j^{k'} = c_j^k$ for c_j^k, $c_j^{k'} \in N$ and

$glb(c_j^{k'}) \geq glb(c^k)$ for c_j^k, $c_j^{k'} \in N^+$

c) at least one $c_j^{k'} > c_j^k$, other $c_j^{k'} \geq c_j^k$ } for c_j^k, $c_j^{k'} \in N$

$glb(c_j^{k'}) \geq glb(c_j^k)$ for c_j^k, $c_j^{k'} \in N^+$

a) and b)

=> $c^{k'}$ SUB c^k

=> $c^{k'}$ is a loop node with no successors (by construction).

=> $c^{k'}$ cannot be part of the infinite path.

=> $c^{k'}$ cannot be part of the infinite subsequence.

c)

=> The j^{th} component of the resulting cube would, by construction, be $[c_j^{k'}]^+$ and not $c_j^{k'}$. Thus this cube cannot occur in the infinite path. Hence it cannot occur in the infinite subsequence.

Thus there can be no elements of such an infinite subsequence, and hence the reachability tree is finite.

6.4 EQUALITY OF REACHABILITY TREE AND REACHABILITY SET

The reachability tree was defined to be a set (tree) of cubes. Of interest here however are the states contained in the cubes in the Reachability Tree. Since the Reachability Tree can be said to represent the set of states contained in all the cubes in it, it will in fact for convenience, be said to contain those states which it represents.

DEFINITION 6.12

Given $RT_{VRS} = \{ c^0, c^1, \ldots c^k \}$ then

$$q \in RT_{VRS} \text{ iff } q \in c^i \text{ and } c^i \in RT_{VRS} \text{ for } i\text{: } 1 \leq i \leq k$$

and

$$RT_{VRS} = c^0 \cup c^1 \cup \ldots \cup c^k$$

In order to show that the Reachability Set and the Reachability Tree are equal (i.e. represent the same set of states) it will be shown that:

a) $q \in RS_{VRS} \Rightarrow q \in c \in RT_{VRS}$

and

b) $q \in c \in RT_{VRS} \Rightarrow q \in RS_{VRS}$

6.4.1 Reachability Tree Contains Reachability Set

In order to show that the Reachability Set for a given Vector Replacement System is a subset of the Reachability Tree for the same Vector Replacement System, it will be shown that any state in the Reachability Set is also in some cube which is a node in the Reachability Tree.

This proof is based on the analogous proof for the Coverability Tree in [Hack 1975], modified for cubes and extended to the Reachability Tree algorithm described here.

Before this can be done, it is necessary to prove a lemma about the relationship of c' and c^{k+1}, as defined in Definition 6.2

LEMMA 6.13 $c^k \xrightarrow{t} c^{k+1}$ and $c^k \xrightarrow{t} c' \Rightarrow c'$ SUB c^{k+1}

For $c^k, c^{k+1} \in RT_{VRS}$ such that c^{k+1} is generated from c^k via transition t and $c^k \xrightarrow{t} c'$ then

$$c' \text{ SUB } c^{k+1}$$

Proof:

By construction c^{k+1} differs from c' only in that for some j, $1 \leq j \leq n$,

$$c_j^{k+1} = [c_j']^+ \text{ for } c_j' \in N \text{ and } c_j^{k+1} \in N^+,$$

while for all others, $c_j^{k+1} = c_j'$.

But coupled with the subcube attributes (Theorem 3.22)

$$\Rightarrow c' \text{ SUB } c^{k+1}$$

That RS_{VRS} SUB RT_{VRS} will be shown as follows.

LEMMA 6.14

For a given VRS = [T, q^{init}]

$$q \in RS_{VRS}$$
$$=>$$

there exists a $c \in RT_{VRS}$ such that $q \in c$.

Proof:

By induction on the length of the transition sequence from q^{init} to q.

BASE

For transition sequence length of 0, $q^{init} \in RS_{VRS}$. By construction $c^{root} = \{q^{init}\}$

Thus $q^{init} \in RS_{VRS}$ => $q^{init} \in c^{root} \in RT_{VRS}$.

STEP

It must be shown that if

$$q^0 \xrightarrow{t^1} q^1 \xrightarrow{t^2} q^2 \ldots q^{k-1} \xrightarrow{t^k} q^k$$

with $q^i \in RS_{VRS}$ for all i: $1 \leq i \leq k$ and each $q^i \in$ some cube (say c^i) $\in RT_{VRS}$ and if

$$q^k \xrightarrow{t^{k+1}} q^{k+1},$$

and

$$q^{k+1} = q^k - I^{t^{k+1}} + O^{t^{k+1}}$$
$$(\text{i.e. } q^{k+1} \in RS_{VRS})$$

that $q^{k+1} \in$ some c, say c^{k+1}, such that $c^{k+1} \in RT_{VRS}$.

$q^k \xrightarrow{t^{k+1}}$, $q^k \in c^k$

$=> \quad c^k \xrightarrow{t^{k+1}}$

Therefore c^k is not a null node.

If c^k is a loop node then there is a loop-back-pointer to the antecedent, say $c^{k'}$, in the tree which includes c^k. If $q^k \leqslant c^k$ and c^k SUB $c^{k'}$ then $q^k \leqslant c^{k'}$.

Also $c^k \xrightarrow{t^{k+1}}$, c^k SUB $c^{k'}$ => $c^{k'} \xrightarrow{t^{k+1}}$.

Therefore it is possible to relabel $c^{k'}$ as c^k and use it in the ensuing argument.

If c^k is not a loop node then no such relabelling is required.

In either case there is now a c^k such that it contains q^k and that transition t^{k+1} is enabled for it. Thus some cube, call it c' is immediately reachable from c^k.

It is now known that:

$$c^k \xrightarrow{t} c', \; q^k \leqslant c^k, \; q^k \xrightarrow{t^{k+1}}, \; q^k \xrightarrow{t^{k+1}} q^{k+1}$$

Hence Lemma 4.21

$$\Rightarrow q^{k+1} \leqslant c'.$$

But if c^{k+1} is the cube generated by the reachability tree algorithm from c^k via transition t^{k+1} then

Lemma 6.13

$$\Rightarrow c' \text{ SUB } c^{k+1}.$$

Therefore $q^{k+1} \leqslant c^{k+1}$.

Thus RS_{VRS} SUB RT_{VRS}.

6.4.2 General Cube Reachability

One last definition of reachability, that for general cube reachability, will now be introduced along with its notation.

DEFINITION 6.15 GENERAL CUBE REACHABILITY

Given any two cubes, c and c′, c′ is said to be general cube reachable from c iff for any state $q' \leqslant c'$, there exists a $q \leqslant c$ and a transition sequence s such that $q \xrightarrow{s} q'$.

This is denoted c ~~~> c′.

It should be noted that by Definition 6.15, any cube is general cube reachable from itself via the null transition sequence. This is denoted c ~~~> c.

The following lemma shows the relationship between immediate and general cube reachability.

LEMMA 6.16

$$c \xrightarrow{s} c'$$

$$\Rightarrow$$

c ~~~> c′

Proof:

This follows directly from Lemma 5.3 and Definition 6.15.

It should be noted that the converse is not necessarily true.

The following two lemmata will show that general cube reachability is transitive.

LEMMA 6.17

$$c^1 \rightsquigarrow c^2 \text{ and } c^2 \rightsquigarrow c^3$$
$$\Rightarrow$$
$$c^1 \rightsquigarrow c^3$$

Proof:

This too follows directly from Definition 6.15.

LEMMA 6.18

$$c^0 \rightsquigarrow c,\ q \in c$$
$$\Rightarrow$$
$$q \in RS_{VRS}$$

Proof:

$c^0 \rightsquigarrow c$ and Definition 6.15

$\Rightarrow$ For any state $q \in c$, there exists a $q^0 \in c^0$ and a transition sequence s such that $q^0 \xrightarrow{s} q$.

Definition 1.18

$\Rightarrow q \in RS_{VRS}$

6.4.3 Reachability Set Contains Reachability Tree

The proof that the Reachability Tree is a subset of the Reachability Set is far more involved than was the proof of the converse. This proof represents a major part of the second main result of this thesis, namely the Reachability Tree Construction Algorithm.

Since states in RT_{VRS} are grouped in cubes it must be shown that if any c is in RT_{VRS} (i.e. c^{root} ---> c for $c^{root} = \{q^{init}\}$), then any $q \in c$ is also $\in RS_{VRS}$.

i.e. it must be shown that

$$c^{root} \dashrightarrow c$$

$$\Rightarrow$$

$$c^{root} \rightsquigarrow c$$

To begin the following lemma extracts trom Definition 6.2 the most often-used properties of the cubes in the Reachability Tree.

LEMMA 6.19

Given

$$c^0 \dashrightarrow c^{a^b} \dashrightarrow c^{a^{b-1}} \dashrightarrow \ldots \dashrightarrow c^{a^2} \dashrightarrow c^{a^1} \dashrightarrow c^{i^e-1} \dashrightarrow c^{i^e}$$

and

b $^+$-backpointers associated with c^{i^e},

each pointing to c^{a^h}, h: $1 \leq h \leq b$.

$\Rightarrow$

a) $c^{a^h} \ll c^{i^e}$ for h, $1 \leq h \leq b$

b) There exists c' and t^{i^e} such that:

$c^{i^e-1} \xrightarrow{t^{i^e}} c'$

$c' = \mathrm{MIN}(\underline{c}^{i^e-1}(t^{i^e})) + D^{t^{i^e}}$

$c' \ll c^{i^e}$

c' SUB c^{i^e}

c) For each $^+$-backpointer, for j: $1 \leq j \leq n$

i) There is only one j, say $j = m_h$, such that $c^{a^h}_{m_h} \leqslant N$ and $c'_{m_h} \leqslant N$ and $\hat{D}^{a^h,k+1,h} = 1$

ii) for all j such that $c^{a^h}_j$, $c'_j \leqslant N$ and $j \neq m_h$, $\hat{D}^{a^h,k+1,h} = 0$

iii) For all other j, (i.e. $c'_j \leqslant N^+$), $\hat{D}^{a^h,k+1,h} \leq 0$

d) $c^{e^i}_j = c'_j$ for $j \neq m_h$, for all h: $1 \leq h \leq b$

$= [c'_j]^+$ for $j = m_h$, for all h: $1 \leq h \leq b$

Proof:

This follows by construction from Definition 6.2

The next theorem is the major result in this chapter. It shows that the method of grouping states as cubes and generating cubes with additional $^+$-components, as described in Definition 6.2 quarantees the presence, in the reachability set, of all states in the newly generated cube.

This theorem shows that any state in a cube with a newly generated $^+$-component is reachable from the initial cube and hence is in the reachability set.

<u>THEOREM</u> 6.20

$$c^0 \dashrightarrow c^{i^1} \dashrightarrow c^{i^2} \dashrightarrow \ldots \dashrightarrow c^{i^g}$$

where the c^{i^e}, e: $1 \leq e \leq g$ are those cubes in a particular branch of the reachability tree RT_{VRS} which have $^+$-backpointers associated with them. (It should be noted that $c^{root} = c^0$ in this case.)

$$\Rightarrow$$

$$c^0 \rightsquigarrow c^{i^g}$$

i.e. for any state $q^g \in c^{i^g}$, there exists some state q^0, $q^0 \in c^0$, and some transition sequence s such that $q^0 \xrightarrow{s} q^g$.

Proof:

This multiply nested inductive proof has been placed in Appendix I because of its length and complexity.

It now remains to extend the result of Theorem 6.20 from cubes in the reachability tree with new $^+$-components to all cubes in the reachability tree.

THEOREM 6.21

$$c^0 \dashrightarrow c$$

$$\Rightarrow$$

$$c^0 \rightsquigarrow c$$

Proof:

If c is a cube with a new $^+$-component, then Theorem 6.20 applies directly and the proof is complete.

Assuming c does not have a new $^+$-component yields two possibilities.

If c has antecedents with new $^+$-components, then letting the one closest to c be c^a, then Theorem 6.20

$\Rightarrow c^0 \rightsquigarrow c^a$

and Definition 6.2

$\Rightarrow c^a \longrightarrow c$

Definition 6.15

$\Rightarrow c^a \rightsquigarrow c$

and Lemma 6.17

$\Rightarrow c^0 \rightsquigarrow c$

If c has no antecedents with new $^+$-components, then Definition 6.2

$\Rightarrow c^0 \longrightarrow c$

and Definition 6.15

$\Rightarrow c^0 \rightsquigarrow c$

Rephrasing this result yields:

<u>THEOREM</u> 6.22

$$\text{There exists a } c \in RT_{VRS} \text{ such that } q \in c$$

$$\Rightarrow$$

$$q \in RS_{VRS}$$

Proof:

Definition 6.3 and $c \in RT_{VRS}$

$\Rightarrow c^0 \longrightarrow c$

Theorem 6.21

$\Rightarrow c^0 \rightsquigarrow c$

Lemma 6.18

$\Rightarrow q \in RS_{VRS}$

6.4.4 Conclusion

Summarizing the results of this chapter yields:

THEOREM 6.23

For a given Vector Replacement System, VRS, and some initial state q^0,

$$RS_{VRS} = RT_{VRS}$$

Proof:

This is shown by Lemma 6.14 and Theorem 6.22.

Thus it has been shown, for any Vector Replacement System (or equivalent Petri Net) for which the Reachability Tree construction algorithm terminates normally, that reachability of an arbitrary state from the given initial state can be determined by inspection of the reachability tree.

That is, a state is reachable from the initial state if and only if it is a member of a cube in the reachability tree, provided only that the reachability tree construction algorithm terminates normally.

6.5 CUBE SIMPLIFICATION

Once a cube representation for a given Reachability Set has been obtained via the Reachability Tree construction algorithm, it is often possible to simplify the cube representation somewhat, viz to reduce the number of cubes needed to specify it.

One technique used is called cube compression. It involves taking two cubes and replacing them with a single equivalent cube containing exactly those states found in the supplanted cubes.

For example the cubes

$$\langle 0\ 1 \rangle \text{ and } \langle 1^+ 1 \rangle$$

can be replaced by

$$\langle 0^+\ 1 \rangle$$

The conditions under which this can be done are shown in the following lemma.

<u>LEMMA</u> 6.24 CUBE COMPRESSION

Given two cubes, c^1 and c^2 where there exists exactly one component say the m^{th}, such that

$$c_j^1 = c_j^2 \qquad \text{for all } j: j \neq m \text{ and}$$
$$= glb(c_j^2) - 1 \text{ for } j : j = m$$
$$\Rightarrow$$
$$c^1 \cup c^2 = c^3$$
$$\text{where } c_j^3 = c_j^1 = c_j^2 \text{ for all } j: j \neq m \text{ and}$$
$$= [c_j^1]^+ \quad \text{for h: } j = m$$

Proof:

It must be shown that

$$q \leqslant c^1 \cup c^2 \iff q \leqslant c^3$$

$q \leqslant c^1 \cup c^2$

$$\iff q \leqslant c^1 \quad \text{or} \quad q \leqslant c^2$$

$$\iff \begin{aligned} q_j &= c_j^1 && \text{for } c_j^1 \leqslant N \text{ and} \\ &\geq glb(c_j^1) && \text{for } c_j^1 \leqslant N^+ \end{aligned}$$

or

$$\begin{aligned} q_j &= c_j^2 && \text{for } c_j^2 \leqslant N \text{ and} \\ &\geq glb(c_j^2) && \text{for } c_j^2 \leqslant N^+ \end{aligned}$$

the relationship between c^1 and c^2

$$\iff \begin{aligned} q_j &= c_j^1 = c_j^2 \text{ for } c_j^1,\ c_j^2 \leqslant N \\ &\geq glb(c_j^1) \text{ for } c_j^1,\ c_j^2 \leqslant N^+ \\ &= c_j^1 \text{ or } \geq glb(c_j^2) \text{ for } c_j^1 \leqslant N,\ c_j^2 \leqslant N^+ \end{aligned}$$

$c_j^1 = glb(c_j^2) - 1$

$$\iff \begin{aligned} q_j &= c_j^1 = c_j^2 \text{ for } c_j^1,\ c_j^2 \leqslant N \\ &\geq glb(c_j^1) \text{ for } c_j^1,\ c_j^2 \leqslant N^+ \\ &\geq c_j^1 \quad \text{for } c_j^1 \leqslant N,\ c_j^2 \leqslant N^+ \end{aligned}$$

definition of c^3

$$\iff \begin{aligned} q_j &= c_j^3 && \text{for } c_j^3 \leqslant N \text{ and} \\ &\geq glb(c_j^3) && \text{for } c_j^3 \leqslant N^+ \end{aligned}$$

$$\iff q \leqslant c^3$$

The other major simplification technique simply involves removing, from the explicit cube representation of the Reachability Set, any cubes which are subcubes of other cubes in the set. This can be done without any loss of information.

EXAMPLE 6.25

The infinite Reachability Set for Example 1.22, expressed as a finite set of cubes is:

$$\langle 0\ 1\ \rangle\ \langle 0\ 2\ \rangle\ \langle 0^+1\ \rangle\ \langle 0^+2^+\rangle\ \langle 0^+3^+\rangle$$
$$\langle 0^+4^+\rangle\ \langle 1\ 1\ \rangle\ \langle 1^+1\ \rangle\ \langle 1^+2^+\rangle\ \langle 1^+3^+\rangle$$
$$\langle 2^+1\ \rangle\ \langle 2^+2^+\rangle\ \langle 3^+1\ \rangle$$

where, for example the infinite set of states represented by the leftmost branch of the tree, { $\langle 1,1\rangle$, $\langle 2,1\rangle$, $\langle 3,1\rangle$, ... } can be represented by the single cube $\langle 1^+,1\ \rangle$.

The cubes $\langle 0\ 2\ \rangle$ $\langle 0^+3^+\rangle$ $\langle 0^+4^+\rangle$ $\langle 1^+2^+\rangle$ $\langle 1^+3^+\rangle$ and $\langle 2^+2^+\rangle$ are all subcubes of $\langle 0^+2^+\rangle$. Similarly, $\langle 0\ 1\ \rangle$ $\langle 1\ 1\ \rangle$ $\langle 1^+1\ \rangle$ $\langle 2^+1\ \rangle$ and $\langle 3^+1\ \rangle$ are all subcubes of $\langle 0^+1\ \rangle$. Eliminating them means the entire Reachability Set can be simplified to:

$$\langle 0^+1\ \rangle\ \langle 0^+2^+\rangle.$$

Applying Lemma 6.24 allows this to be further simplified to:

$$\langle 0^+1^+\rangle.$$

Thus a single cube suffices to represent the entire Reachability Set for this example.

6.6 IMPLEMENTATION

A computer program implementation of an earlier version of the Reachability Tree Construction Algorithm was written to assist in developing the results presented here.

This program is written in Fortran (Watfiv-S) and runs interactively. It accepts input, in the form of a Vector Replacement System description, from an on-line terminal or from a previously stored file.

It can produce the Karp and Miller Coverability Tree and/or the Reachability Tree presented here and print out the states in the resulting tree.

The program can also apply the simplification rules presented in the preceding section and print out a simplified version of the Reachability Set.

It can also analyze the Vector Replacement System and determine its membership in one of several subclasses.

Because the program was intended solely as a design tool and not a production program it was not modified to coincide precisely with the latest version of the Reachability Tree Construction Algorithm. Sample output for the Producer/Consumer example is given in Appendix II.

Chapter VII

DISCUSSION OF RESULTS

7.1 COMPARISON WITH KARP & MILLER COVERABILITY TREE

The Karp and Miller Reachability Tree construction algorithm [Karp & Miller 1968], here called the Karp and Miller Coverability Tree construction algorithm, also folds infinite sets of states into single nodes in the Coverability Tree. When a component becomes unbounded it is represented by w, unfortunately thereby causing a loss of information. For example all of the following sets of states are represented by $\langle 1\ w \rangle$:

$$\{ \langle 1\ 0 \rangle, \langle 1\ 1 \rangle, \langle 1\ 2 \rangle, \langle 1\ 3 \rangle, \ldots \}$$

$$\{ \langle 1\ 0 \rangle, \langle 1\ 2 \rangle, \langle 1\ 4 \rangle, \langle 1\ 6 \rangle, \ldots \}$$

$$\{ \langle 1\ 5 \rangle, \langle 1\ 8 \rangle, \langle 1\ 11 \rangle, \langle 1\ 14 \rangle, \ldots \}$$

and the following sets of states can be represented by $\langle w\ w \rangle$:

$$\{ \langle 1\ 1 \rangle, \langle 1\ 2 \rangle, \ldots \langle 2\ 1 \rangle, \langle 2\ 2 \rangle, \ldots \}$$

$$\{ \langle 1\ 1 \rangle, \langle 2\ 2 \rangle, \langle 3\ 3 \rangle, \ldots \}$$

$$\{ \langle 2\ 3 \rangle, \langle 3\ 5 \rangle, \langle 4\ 7 \rangle, \ldots \}.$$

Thus it is not known

a) What minimum value w can take

b) What increment from one state to another is represented by w,

c) Whether two or more w-components are dependent upon each other.

For this reason the Coverability Tree is not sufficient to determine reachability of a given state.

The Reachability Tree construction algorithm presented here uses the Cube Notation and modifications to the Karp and Miller Coverability Tree construction algorithm to overcome these difficulties.

The Cube Notation permits representation of the minimum value (greatest lower bound or glb) for a component, and all integral values greater than or equal to it.

Furthermore, the Reachability Tree construction algorithm ensures that:

a) all components which become unbounded do so with an increment of 1 and

b) only one component at a time may become unbounded via a given transition sequence.

Any Vector Replacement System or Petri Net whose syntax, or structure, and dynamics cause the violation of any of these conditions are rejected. This implies that the Reachability Tree construction algorithm defines a non-syntactic subclass of Vector Replacement Systems and Petri Nets.

These restrictions also add considerable complexity to the Reachability Tree construction algorithm and even more complexity to the proof of its correctness. In particular this applies to that portion of the proof which shows that any state in the Reachability Tree is also in the Reachability Set.

One further change to the Coverability Tree Construction Algorithm involves loop nodes. Karp and Miller required equality with an antecedent node before calling a new node a loop node. Here only inclusion of all states in a new node in an antecedent is required.

7.2 SYNTACTIC AND NON-SYNTACTIC SUBCLASSES

Because some analysis questions about Petri Nets are difficult to answer, or in the case of the Reachability Problem not yet known to be decidable, and because simpler constructions are sometimes adequate, many subclasses of Vector Replacement Sytems and Petri Nets have been defined.

Syntactic subclasses are those which permit a Vector Replacement System's, or Petri Net's, membership in that subclass to be determined by examining their respective structures.

Non-syntactic subclasses on the other hand are those for which such structural examination is insufficient. In these cases, membership in such subclasses is dependent upon dynamic properties, i.e. behaviour, of the Vector Replacement System or Petri Net.

7.3 RT-SUBCLASS

The Reachability Tree construction algorithm defines a non-syntactic subclass of Vector Replacement Systems or Petri Nets.

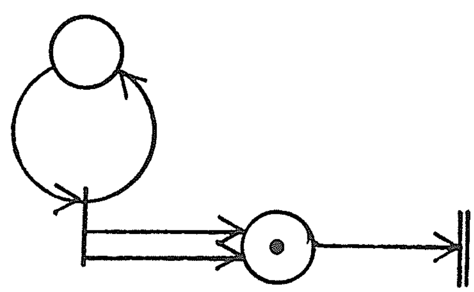
DEFINITION 7.1

The RT-subclass consists of those Vector Replacement Systems or Petri Nets for which the Reachability Tree construction algorithm terminates normally.

Membership in the RT-subclass cannot be determined merely by examining the structure of the Petri Net or Vector Replacement System, but rather depends upon the successful completion of the Reachability Tree construction algorithm. Hence it is a non-syntactic subclass.

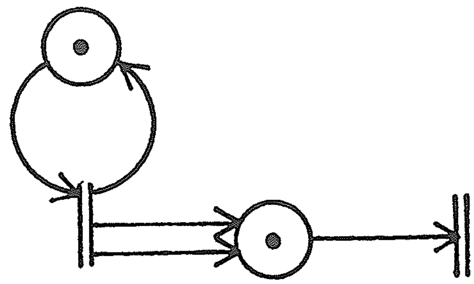
In fact, it is possible for a given Vector Replacement Sytem or Petri Net with a particular initial state or marking to be in the RT-subclass and for the same net with a different initial marking to be excluded.

A trivial example suffices to illustrate this.



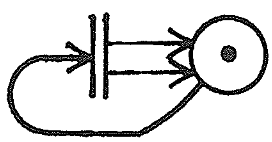
is in the RT-subclass.

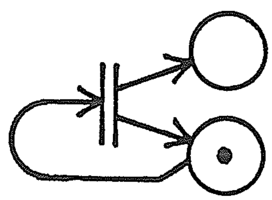
On the other hand,

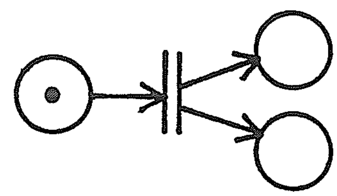


is not in the RT-subclass because a place can become unbounded with an increment other than one, in this case two.

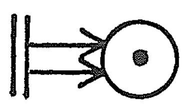
Further examples which are members of the RT-subclass are:

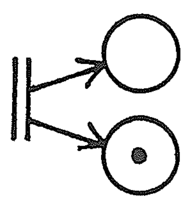


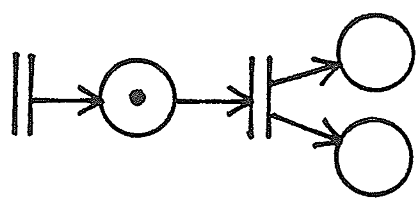




while the following, respectively syntactically similar examples are not:







The absence of a simple, syntactic test to determine RT-subclass membership restricts the usefulness of this subclass. The Reachability Tree construction algorithm must be attempted and allowed to terminate, either normally or abnormally, to determine membership and non-membership, respectively.

7.4 COMPARISON TO OTHER SUBCLASSES

In order to put the RT-subclass into proper perspective, several examples of syntactic and non-syntactic subclasses will be introduced briefly and then related to the non-syntactic subclass defined by the Reachability Tree construction algorithm, the RT-subclass.

The common syntactic subclasses to be explored are Marked Graphs, State Machines, Conflict Free Nets, and Free Choice Nets.

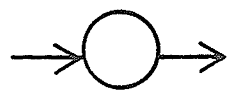
In addition, several extensions to State Machines and Marked Graphs will be introduced here for the first time, namely Extended Marked Graphs, Extended State Machines and Further Extended State Machines.

The common non-syntactic subclasses which will be discussed are Safe Nets, Bounded Nets, and Persistent Nets.

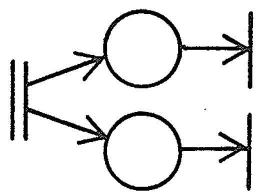
7.4.1 Marked Graphs

Marked Graphs are Petri Nets where each place has exactly one input arc and exactly one output arc.

i.e.

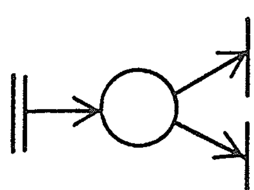


The following Petri Net:



is a Marked Graph but is not in the RT-subclass because a single transition can simultaneously increase the marking of two places an unbounded number of times. Thus the two places do not become unbounded independently.

The following Petri Net:



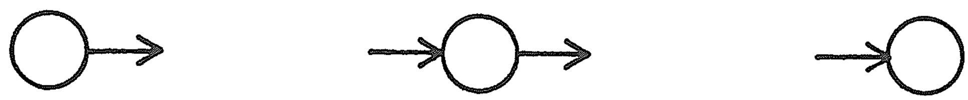
is certainly in the RT-subclass, yet it is not a Marked Graph because a place has more than one output arc.

Thus Marked Graphs neither contain, nor are contained by, the RT-subclass.

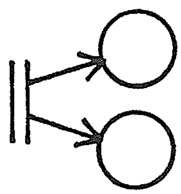
7.4.2 Extended Marked Graphs

Extended Marked Graphs are defined to be Petri Nets where each place has at most one input arc and at most one output arc.

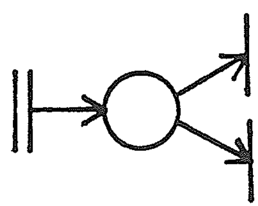
i.e.



The following Petri Net:



is an Extended Marked Graph but it is not in the RT-subclass because a single transition can simultaneously increase the marking of two places an unbounded number of times, whereas:



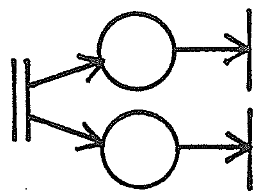
is certainly in the RT-subclass, yet it is not an Extended Marked Graph because a place has more than one output arc.

Thus Extended Marked Graphs neither contain, nor are contained by, the RT-subclass.

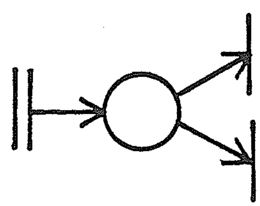
7.4.3 Conflict Free Nets

Conflict Free Nets are Petri Nets where each place has at most one output arc. There are no restrictions on the number of input arcs.

The following Petri Net:



is a Conflict Free Net but it is not in the RT-subclass because the two places do not become unbounded independently whereas:



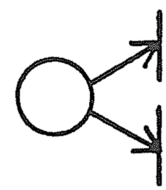
is certainly in the RT-subclass, yet it is not a Conflict Free Net again because a place has more than one output arc.

Thus Conflict Free Nets neither contain, nor are contained by, the RT-subclass.

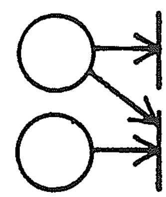
7.4.4 Free Choice Nets

Free Choice Nets are Petri Nets where, if a place is an input to several transitions, then it is the only input place for these transitions.

i.e.

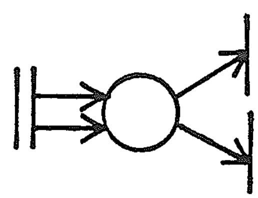


is a Free Choice Net but

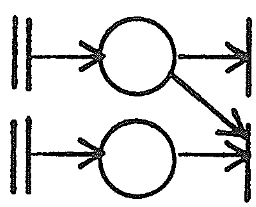


is not because the top place is an input place to both transitions but it is not the only input place to the bottom transition.

The following Petri Net:



is a Free Choice Net but it is not in the RT-subclass because it contains a place which becomes unbounded by an increment other than one, whereas:



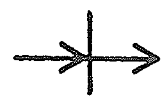
is certainly in the RT-subclass, yet it is not a Free Choice Net as was seen above.

Thus Free Choice Nets neither contain, nor are contained by, the RT-subclass.

7.4.5 State Machines

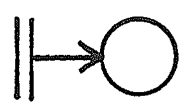
State Machines are Petri Nets where each transition has exactly one input arc and exactly one output arc.

i.e.



It is known [Peterson 1981] that State Machines are finite. This means that no places (components) become unbounded and hence State Machines are contained in the RT-subclass.

On the other hand, the following Petri Net:

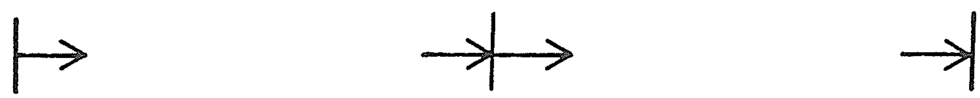


is certainly in the RT-subclass, yet it is not a State Machine because the transition has no input arc.

Thus State Machines are contained in, but do not contain, the RT-subclass.

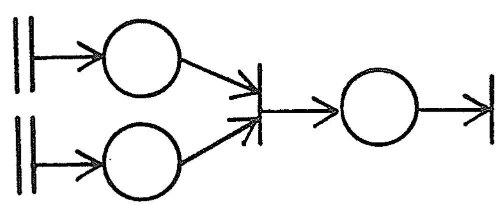
7.4.6 Extended State Machines

<u>Extended State Machines</u> are defined to be Petri Nets where each transition has at most one input arc and at most one output arc.

i.e. 

Extended State Machines are not finite since a transition with no input arc and one output arc leading to a place can create arbitrarily many tokens in that place. This is however the only way in which unboundedness of places in the net can be initiated. Thus such unboundedness, whether occurring in a place immediately preceded by a transition with no inputs, or occurring at some other location in the Petri Net, occurs one token at a time ($D_j = 1$) and for only one place at a time (all other $D_j \leq 0$). Furthermore, it can be shown that the $\hat{s}$ of Definition 6.2 consist of single transitions. Thus Extended State Machines are contained in the RT-subclass.

On the other hand, the following Petri Net:



is certainly in the RT-subclass, yet it is not an Extended State Machine because a transition has more than one input arc.

Thus Extended State Machines are contained in, but do not contain, the RT-subclass.

7.4.7 Further Extended State Machines

Further Extended State Machines are defined to be Petri Nets where each transition has at most one output arc. It may have any number of input arcs. (Incidentally, Further Extended State Machines are duals of Conflict Free Nets.)

Arguments similar to those for Extended State Machines lead to the conclusion that Further Extended State Machines are also contained in the RT-subclass.

The counterexample used for Extended State Machines also serves to show that Further Extended State Machines do not contain the RT-subclass.

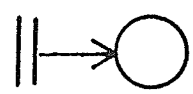
Thus Further Extended State Machines are contained in, but do not contain, the RT-subclass.

7.4.8 Safe Nets and Bounded Nets

Bounded Nets are Petri Nets where each place is bounded. Safe Nets are Bounded Nets where the bound for each place is 1.

Since there are no unbounded places in such a net the Reachability Tree construction algorithm will always terminate normally. Hence Bounded Nets and Safe Nets are contained in the RT-subclass.

A simple net which is in the RT-subclass but is not bounded is:

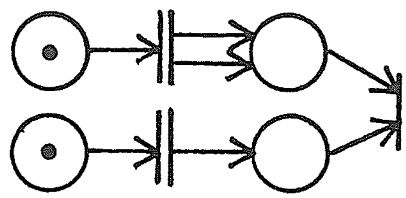


Thus Bounded and Safe Nets are contained in, but do not contain, the RT-subclass.

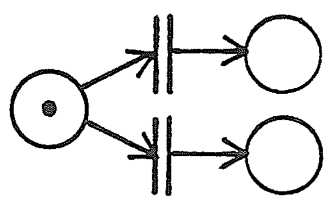
7.4.9 Persistent Nets

<u>Persistent Nets</u> are Petri Nets where, if two transitions are concurrently enabled, the firing of one transition cannot disable the other.

The Petri Net:



is a Persistent Net but is not in the RT-subclass because a place becomes unbounded by an increment other than one, namely two. The net



is in the RT-subclass but is not a Persistent Net because the firing of either transition disables the other.

Thus Persistent Nets neither contain, nor are contained in, the RT-subclass.

7.5 SIGNIFICANCE OF SUBCLASS

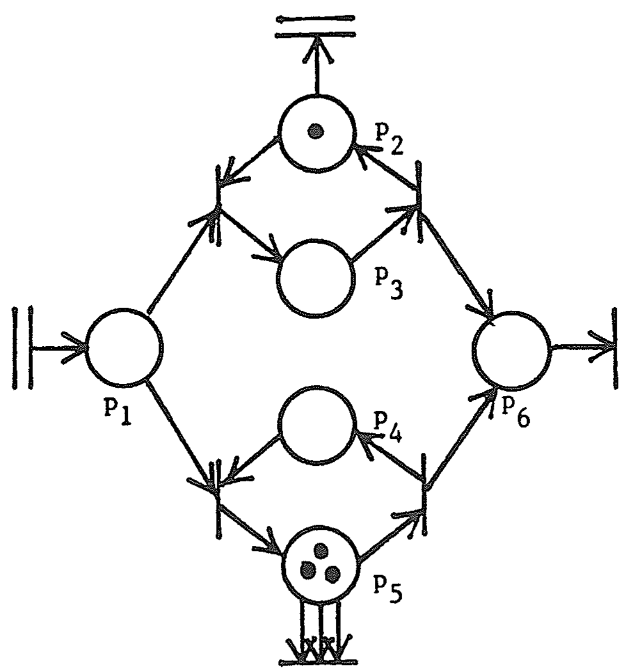
The RT-subclass, i.e. the non-syntactic subclass of Vector Replacement Systems and Petri Nets defined by the Reachability Tree construction algorithm, has been shown to be non-trivial.

It contains, properly, the syntactic subclasses State Machines, Extended State Machines and Further Extended State Machines and the non-syntactic subclasses, Safe Nets and Bounded Nets.

Thus decideability of reachability, and an algorithm for determining reachability, have been shown for a heretofor unexplored and non-trivial subclass of Petri Nets and Vector Replacement Systems.

The following Net is an example:

EXAMPLE 7.2



where $q^0 = \langle 0\ 1\ 0\ 0\ 3\ 0 \rangle$.

The Reachability Set can be expressed as:

$$< 0^+ \; 1 \; 0 \; 0 \; 3 \; 0^+ >$$

$$< 0^+ \; 1 \; 0 \; 1 \; 2 \; 0^+ >$$

$$< 0^+ \; 1 \; 0 \; 2 \; 1 \; 0^+ >$$

$$< 0^+ \; 1 \; 0 \; 3 \; 0 \; 0^+ >$$

$$< 0^+ \; 0 \; 1 \; 0 \; 3 \; 0^+ >$$

$$< 0^+ \; 0 \; 1 \; 1 \; 2 \; 0^+ >$$

$$< 0^+ \; 0 \; 1 \; 2 \; 1 \; 0^+ >$$

$$< 0^+ \; 0 \; 1 \; 3 \; 0 \; 0^+ >$$

$$< 0^+ \; 1 \; 0 \; 0 \; 0 \; 0^+ >$$

$$< 0^+ \; 0 \; 1 \; 0 \; 0 \; 0^+ >$$

$$< 0^+ \; 0 \; 0 \; 0 \; 0 \; 0^+ >$$

$$< 0^+ \; 0 \; 0 \; 0 \; 3 \; 0^+ >$$

$$< 0^+ \; 0 \; 0 \; 1 \; 2 \; 0^+ >$$

$$< 0^+ \; 0 \; 0 \; 2 \; 1 \; 0^+ >$$

$$< 0^+ \; 0 \; 0 \; 3 \; 0 \; 0^+ >$$

Chapter VIII

CONCLUSION

8.1 SUMMARY OF RESULTS

This thesis has introduced a new, non-syntactic subclass of Vector Replacement Systems and Petri Nets called the RT-subclass. This subclass is characterized as containing precisely those Vector Replacement Systems or Petri Nets for which the Reachability Tree Construction Algorithm, also introduced here, terminates normally, i.e. permits determination of reachability.

The RT-subclass is called non-syntactic because membership in it is defined not in terms of the structure or syntax, rather in terms of the behaviour, or dynamics, of the Vector Replacement System or Petri Net.

In order to develop this algorithm, a notation called "Cube Notation" was introduced which permits folding certain infinite sets of states, such as $\{ \langle 0, 1, 2 \rangle, \langle 0, 1, 3 \rangle, \langle 0, 1, 4 \rangle, \ldots \}$ into finite sets of cubes, such as $\langle 0, 1, 2^{+} \rangle$.

It has been shown, for any Vector Replacement System or Petri Net in the RT-subclass, that Reachability can be determined by inspection of the cubes in the Reachability Tree produced by the Reachability Tree Construction Algorithm.

The Reachability Tree can also be simplified using the cube simplification rules to produce a simplified Reachability Set, also consisting of cubes.

Three new syntactic subclasses were also introduced, Extended State Machines, Further Extended State Machines and Extended Marked Graphs.

It was then shown that the RT-subclass is non-trivial, that State machines, Extended State Machines, Further Extended State Machines, Safe Nets and Bounded Nets are all properly contained in the RT-subclass, and that Marked Graphs, Extended Marked Graphs, Conflict Free Nets, Free Choice Nets and Persistent Nets are incomparable with the RT-subclass.

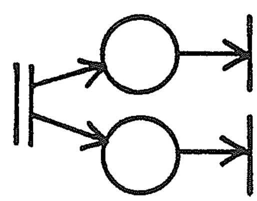
8.2 FURTHER WORK

There are a great many areas in which further work can be done. Several of these are outlined below.

More easily applied necessary and sufficient conditions for RT-subclass membership are desirable. Purely syntactic conditions, being the simplest to test would therefore also be the most desirable, but simpler non-syntactic conditions would also be useful.

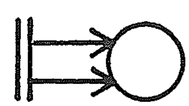
It may also be possible to extend this work to a somewhat less restrictive subclass. This may require encoding of more information in the notation or the use of more complex expressions.

For example, it would be desirable to include constructs such as:



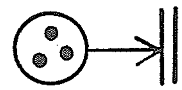
which are currently excluded by the Reachability Tree construction algorithm conditions but whose Reachability Set is obviously = $< 0^+ \; 0^+ >$.

Similarly:



has { < 0 >, < 2 >, < 4 >, ... } as its Reachability Set. Perhaps some encoding such as $< 0^{+2} >$ might have some merit.

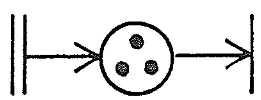
Another possibility is seen in situations like:



where the Reachability Set = $\{ \langle 3 \rangle, \langle 2 \rangle, \langle 1 \rangle, \langle 0 \rangle \}$. This might be representable as $\langle 3_{-} \rangle$.

(i.e. a $_{-}$-component is generated whenever a new node (cube) is covered by an antecedent.)

If this $_{-}$-notation for cubes were coupled with the $^{+}$-notation for cubes, the resulting $^{+}_{-}$-notation might permit representing the Reachability Set of:



as $\{ \langle 3^{+}_{-} \rangle \}$ which could be simplified to $\{ \langle 0^{+} \rangle \}$. This would serve to eliminate a (potentially) large number of nodes from some Reachability Trees.

The application of these techniques is in general of course far more complex than these simple illustrations show. It is possible however that one or other of them may produce useful results.

Another area worthy of further attention is that of simplifying or pruning the Reachability Tree produced by the Reachability Tree construction algorithm.

One approach would be the elimination from further consideration not only of those nodes which are contained in an antecedent in the tree, but also the elimination from further consideration of those nodes which are contained in any other non-terminal node in the tree. It should then be necessary only to develop one subtree for the largest containing node. This approach would tend to destroy the tree structure though, in the sense that the loop backpointers would not only go "up" in the tree but might also point "laterally" to some other subtree. The resulting construct would therefore more appropriately be called a Reachability Graph. This approach would of course further complicate the proof.

A first attempt at developing a Reachability Graph construction algorithm has been made and a computer program implemented to perform the construction. Sample output for the Producer/Consumer example is given in Appendix II.

Results with numerous examples are encouraging. The algorithm has in all cases attempted so far yielded correct results and has required less memory and execution time than the Reachability Tree construction algorithm. No attempt has as yet been made to prove its correctness. This must await further refinement of the algorithm.

Lastly, the results presented here should be compared to other related work. This would include

a) corresponding subclasses (if any) of related models

b) Petri Net and other Languages

c) analogues of the Reachability Problem in other areas.

REFERENCES

[Agerwala 1979] Agerwala, T., "Putting Petri Nets to Work", Computer, I.E.E.E., December 1979

[Crespi-Reghizzi & Mandrioli 1975] Crespi-Reghizzi, S. and Mandrioli, D., "A Decidability Theorem For a Class of Vector Addition Systems", Information Processing Letters, Volume 3, Number 3,January 1975.

[Ellis & Nutt 1980] Ellis, C.A. and Nutt, G.J., "Office Information Systems and Computer Science", Computing Surveys, Volume 12, Number 1, Association for Computing Machinery, New York, March 1980.

[Hack 1974] Hack, M., "Analysis of Production Schemata by Petri Nets", MSc Thesis, Department of Electrical Engineering, Massachusetts Institute of Technology, Cambridge, Massachusetts, February 1972. Also in Technical Report 94, Project MAC, Massachusetts Institute of Technology, Cambridge, Massachusetts, February 1972 Errata: "Corrections to Analysis of Production Schemata by Petri Nets", Computation Structures Group Note 17, Project MAC, Massachusetts Institute of Technology, Cambridge, Massachusetts, June 1974.

[Hack 1975] Hack, M., "Decision Problems for Petri Nets and Vector Addition Systems", Technical Memo 59, Project MAC, Massachusetts Institute of Technology, Cambridge, Massachusetts, March 1975.

[Hopcroft & Pansiot 1976] Hopcroft, J. and Pansiot, J-J., "On the Reachability Problem for 5-Dimensional Vector Addition Systems", Technical Report, Department of Computer Science, Cornell University, Ithaca, New York, June 1976.

[Karp & Miller 1968] Karp, R. and Miller, R., "Parallel Program Schemata", RC-2053, IBM T. J. Watson Research Center, Yorkton Heights, New York, April 1968. Also in Journal of Computer and Systems Science, Volume 3, Number 4, May 1969.

[Keller 1972] Keller, R., "Vector Replacement Systems: A Formalism for Modelling Asynchronous Systems", Technical Report 117, Computer Science Laboratory, Princeton University, Princeton, New Jersey, December 1972, Revised January 1974.

[König 1936] König, D., "Theorie der Endlichen und Unendlichen Graphen", Akademische Verlagsgesellschaft, Leipzig, 1936.

[Laucht 1979] Laucht, C. M., "Cube Notation for Vector Replacement Systems", Proceedings of the Ninth Manitoba Conference on Numerical Mathematics and Computing, Winnipeg, Manitoba, Canada, September 1979.

[Lipton et al 1974] Lipton, R and Snyder, L. and Zalcstein, Y., "A Comparative Study of Models of Parallel Computation", Proceedings of the 15th Annual Symposium on Switching and Automata, IEEE, New York, 1974

[Meldman 1977] Meldman, J., "New Techniques for Modelling the Behaviour of Man-Machine Information Systems", Sloan Management Review, Volume 18, Number 3, Spring 1977.

[Meldman and Holt 1971] Meldman, J. and Holt, A., "Petri Nets and Legal Systems", Jurimetrics, Volume 12, Number 2, Dec 1971.

[Misunas 1973] Misunas, D., "Petri Nets and Speed Independent Design", Communications of the ACM, Volume 16, Number 8, Association for Computing Machinery, New York, August 1973.

[Noe 1971] Noe, J., "A Petri Net Model of the CDC 6400", Report 71-04-03, Computer Science Department, University of Washington, Seattle, Washington, 1971. Also in Proceedings of the ACM SIGOPS Workshop on System Performance Evaluation, ACM, New York, April 1971.

[Patil & Dennis 1972] Patil, S. and Dennis, J., "The Description and Realization of Digital Systems", Computation Structures Group Memo 71 Project MAC, Massachusetts Institute of Technology, Cambridge, Massachusetts, October 1972. Also in Proceedings of the 6th Annual IEEE Computer Society International Conference Digest of Papers, COMPCON 72, IEEE, New York, 1972.

[Peterson & Bredt 1974] Peterson, J. and Bredt, T., "A Comparison of Models of Parallel Computation", Information Processing 74, Proceedings of the 1974 IFIP Congress, North-Holland, Amsterdam, August 1974.

[Peterson 1977] Peterson, J., "Petri Nets", Computing Surveys, Volume 9, Number 3, ACM, New York, September 1977.

[Peterson 1981] Peterson, J., "Petri Net Theory and the Modelling of Systems", Prentice-Hall, Englewood Cliffs, N.J., 1981.

[Roth 1956] Roth, J.P., "Algebraic Topological Methods for the Synthesis of Switching Systems I", Transactions of the American Mathematical Society, July 1958. Also in ECP 56-02, Institute for Advanced Study, April, 1956.

[Sacerdote & Tenney 1977] Sacerdote, S. and Tenney, R., "The Decidability of the Reachability Problem for Vector Addition Systems", Proceedings of the 9th Annual ACM Symposium on the Theory of Computing, Association for Computing Machinery, New York, 1977. Notice of retraction of proof given in Proceedings of the 10th Annual ACM Symposium on the Theory of Computing, Association for Computing Machinery, New York, 1978.

[Van Leeuwen 1974] Van Leeuwen, J., "A Partial Solution to the Reachability Problem for Vector Addition Systems", Proceedings of the 6th Annual ACM Symposium on the Theory of Computing, Association for Computing Machinery, New York, 1974.

APPENDIX I

THEOREM 6.20

$$c^{i^0} \dashrightarrow c^{i^1} \dashrightarrow c^{i^2} \dashrightarrow \ldots \dashrightarrow c^{i^g}$$

where the c^{i^e}, e: $1 \le e \le g$ are those cubes in a particular branch of the reachability tree RT_{VRS} which have $^+$-backpointers associated with them and c^{i^0} is the root node c^{root}.

$$\Rightarrow$$

$$c^{i^0} \rightsquigarrow c^{i^g}$$

i.e. for any state $q^g \in c^{i^g}$, there exists some state q^0, $q^0 \in c^{i^0}$, and some transition sequence s such that $q^0 \xrightarrow{s} q^g$.

Proof:

The proof will be by induction upon e: $0 \le e \le g$.

begin INDUCTION 1

begin BASE 1

For e = 0, it follows from Definition 6.3 that

$c^{i^0} \rightsquigarrow c^{i^e}$

Definition 6.2

$\Rightarrow c^{i^0} = c^{i^e}$ has no $^+$-backpointers.

Letting b^e be the number of $^+$-backpointers associated with c^{i^e}

$\Rightarrow b^e = b^0 = 0$

end BASE 1

begin STEP 1

Assuming as the induction hypothesis that,

a) For e = 1

$$c^{i^0} \rightsquigarrow c^{i^{e-1}}$$

and

$$b^0 = 0$$

i.e. c^{i^0} has no new $^+$-components

b) For e > 1,

for all c^{i^z}, z: $1 \leq z \leq e - 1$

$$c^{i^0} \rightsquigarrow c^{i^z}$$

and

$g^{a^{h^z}, i^z}$ as defined in Definition 6.2 exists for each $c^{a^{h^z}}$ such that there is a $^+$-backpointer from c^{i^z} to $c^{a^{h^z}}$, h^z: $1 \leq h^z \leq b^z$.

and

for any $q^z \lessdot c^{i^z}$ and c'^z corresponding to c^{i^z} as per Definition 6.2, there exists a $q'^z \lessdot c'^z$ such that $q'^z \xrightarrow{s'^z} q^z$

where

$$q'^z = q^z - R^{b^z}\hat{D}^{a^{b^z}, i^z} - \ldots - R^1\hat{D}^{a^1, i^z}$$

and

$$s'^z = R^{b^z} g^{a^{b^z}, i^z} \ldots R^1 g^{a^1, i^z}$$

Then for any e,

it will be shown that

$$c^{i^0} \sim\sim\sim> c^{i^e}$$

and

$\hat{g}^{a^{h^e}, i^e}$ as defined in Definition 6.2 exists for each $c^{a^{h^e}}$ such that there is a $^+$-backpointer from c^{i^e} to $c^{a^{h^e}}$, h^e: $1 \leq h^e \leq b^e$.

and

for any $q^e \lessdot c^{i^e}$ and c'^e corresponding to c^{i^e} as per Definition 6.2, there exists a $q'^e \lessdot c'^e$ such that $q'^e \xrightarrow{s'^e} q^e$

where

$$q'^e = q^e - R^{b^e}\hat{D}^{a^{b^e}, i^e} - \ldots - R^1\hat{D}^{a^1, i^e}$$

and

$$s'^e = R^{b^e}\hat{g}^{a^{b^e}, i^e} \ldots R^1\hat{g}^{a^1, i^e}$$

Definition 6.2

$$\Rightarrow c^{i^0} \dashrightarrow c^{a^{b^e}} \dashrightarrow c^{a^{b^e-1}} \dashrightarrow \ldots \dashrightarrow c^{a^1} \dashrightarrow c^{i^e}$$

Lemma 6.19

$\Rightarrow$ a) $c^{a^h} \ll c^{i^e}$ for h: $1 \leq h \leq b^e$

where b^e is the number of new $^+$-components associated with c^{i^e}.

b) There exists c^{i^e-1}, c'^e and t^{i^e} such that

$c^{i^0} \dashrightarrow c^{i^e-1} \dashrightarrow c^{i^e}$,

$c^{i^e-1} \xrightarrow{t^{i^e}} c'^e$,

$c'^e = \mathrm{MIN}(\underline{c}^{i^e-1}(t^{i^e})) + D^{t^{i^e}}$,

$c'^e \ll c^{i^e}$ and c'^e SUB c^{i^e}.

c) For each $^+$-backpointer (i.e. for each h: $1 \leq h \leq b^e$), for j = 1, ..., n

i) There is only one j: $j = m_h$ such that $c^{a^h}_{m_h}$, $c'^e_{m_h} \nless N$ and $\hat{D}^{a^h,i^e}_{m_h} = 1$

ii) For all other $c^{a^h}_{m_h}$, $c'^e_{m_h} \nless N$ where j: $j \neq m_h$,

$$\hat{D}^{a^h,i^e}_{j} = 0$$

iii) For all other cases (ie $c'^e_j \nless N^+$ and j: $j \neq m_h$),

$\hat{D}^{a^h,i^e}_{j} \leq 0$

d) $c_j^{i^e} = c_j'^{e}$ for $j: j \neq m_h$

$= [c_j'^{e}]^+$ for $j: j = m_h$ } for all $h : 1 \leq h \leq b^e$

Defintion of c'^{e} and c^{i^e}

$\Rightarrow MIN(c'^{e}) = MIN(c^{i^e})$

Since c^{i^e} is the next cube in RT_{VRS} after $c^{i^{e-1}}$ to have any $^+$-backpointers, each intervening cube is immediately reachable from its immediate antecedent.

$\Rightarrow c^{i^{e-1}} \longrightarrow c'^{e}$

Lemma 6.16

$\Rightarrow c^{i^{e-1}} \sim\sim\sim> c'^{e}$

$c^{i^0} \sim\sim\sim> c^{i^z}$ for all $z: 1 \leq z \leq e-1$

$\Rightarrow c^{i^0} \sim\sim\sim> c^{i^{e-1}}$

Lemma 6.17

$\Rightarrow c^{i^0} \sim\sim\sim> c'^{e}$

Thus to show $c^{i^0} \sim\sim\sim> c^{i^e}$, it is sufficient to show $c'^{e} \sim\sim\sim> c^{i^e}$.

First it will be shown

for $q^{i^e} = MIN(c'^{e}) = MIN(c^{i^e})$,

and c^{a^h}, any antecedent of c^{i^e} to which a $^+$-backpointer points from c^{i^e}

for $h: 1 \leq h \leq b^e$ as defined in

that there exists a

$q^{a^h} \in c^{a^h}$ and δ^{a^h, i^e} Definition 6.2

such that

$q^{a^h} \xrightarrow{\delta^{a^h, i^e}} q^{i^e}$

begin INDUCTION 2

This will be shown by induction upon k where k goes from i^e to $a^h + 1$ in steps of -1. s^{k-1} will be the transition sequence leading from q^{k-1} to q^{i^e}.

begin BASE 2

For $k = i^e$, $q^{i^e} = \mathrm{MIN}(c'^{e})$ and Definition 6.2

$\Rightarrow$ There exists $q^{i^e-1} \in c^{i^e-1}$ such that $q^{i^e-1} \xrightarrow{t^{i^e}} q^{i^e}$

where $q^{i^e-1} = q^{i^e} - D^{t^{i^e}}$

$\Rightarrow s^{i^e-1} = t^{i^e}$

$\Rightarrow q^{i^e-1} \xrightarrow{s^{i^e-1}} q^{i^e}$

end BASE 2

begin STEP 2

For $k \leq i^e - 1$,

assuming as the induction hypothesis that

$$q^k \xrightarrow{s^k} q^{i^e}$$

it will be shown that there exists a $q^{k-1} \in c^{k-1}$

such that

$$q^{k-1} \xrightarrow{s^{k-1}} q^{i^e}$$

and furthermore that

$$s^{k-1} = t^k R^{b^k} s^{a^{b^k},k} \ldots R^1 s^{a^1,k} s^k \quad \text{for } b^k \neq 0 \text{ and}$$

$$= t^k s^k \quad \text{for } b^k = 0$$

where b^k is the number of $^+$-backpointers associated with c^k.

(It should be noted that $b^k \neq 0$ only for a c^k which is one of the c^{i^z}, z: $1 \leq z \leq e$. All other intervening c^k do not have any additional $^+$-backpointers, and hence have $b^k = 0$.)

If $b^k \neq 0$ then $c^k = c^{i^z}$ for some z, $1 \leq z \leq e-1$ and there exist b^k $^+$-backpointers. The induction hypothesis from STEP 1

=> For any $q^k \leqslant c^k$,

there exists a $q'^k \leqslant c'^k$ such that $q'^k \xrightarrow{s'^k} q^k$

where

$$q'^k = q^k - R^{b^k} \hat{D}^{a^{b^k},k} - \ldots - R^1 \hat{D}^{a^1,k}$$

and

$$s'^k = R^{b^k} \mathcal{S}^{a^{b^k},k} \ldots R^1 \mathcal{S}^{a^1,k}$$

Definition 6.2 and $q'^k \leqslant c'^k$

=> There exists a $q^{k-1} \leqslant c^{k-1}$ such that

$q^{k-1} \xrightarrow{t^k} q'^k$ where $q^{k-1} = q'^k - D^{t^k}$

=> $q^{k-1} = q^k \; R^{b^k} \hat{D}^{a^{b^k},k} - \ldots - R^1 \hat{D}^{a^1,k}$

Thus $q^{k-1} \xrightarrow{t^k} q'^k$, $q'^k \xrightarrow{s'^k} q^k$ and $q^k \xrightarrow{s^k} q^{i^e}$

=> $q^{k-1} \xrightarrow{s^{k-1}} q^{i^e}$

where $s^{k-1} = t^k s'^k s^k$

$= t^k R^{b^k} \mathcal{S}^{a^{b^k},k} \ldots R^1 \mathcal{S}^{a^1,k} s^k$

If $b^k = 0$, then there exist no new $^+$-backpointers and Definition 6.2

=> There exists a $q^{k-1} \leqslant c^{k-1}$ such that

$q^{k-1} \xrightarrow{t^k} q^k$ where $q^{k-1} = q^k - D^{t^k}$

Thus $q^{k-1} \xrightarrow{t^k} q^k$, and $q^k \xrightarrow{s^k} q^{i^e}$

=> $q^{k-1} \xrightarrow{s^{k-1}} q^{i^e}$

where $s^{k-1} = t^k s^k$

Thus for any b^k,

$$q^{k-1} \xrightarrow{s^{k-1}} q^{i^e}$$

where

$$s^{k-1} = t^k R^{b^k} \mathfrak{s}^{a^{b^k},k} \ldots R^1 \mathfrak{s}^{a^1,k} s^k \quad \text{for } b^k \neq 0 \text{ and}$$

$$= t^k s^k \quad \text{for } b^k = 0$$

<u>end</u> STEP 2

Letting $k = a^h + 1$

$$\Rightarrow q^{a^h} \leq c^{a^h} \text{ and } q^{a^h} \xrightarrow{s^{a^h}} q^{i^e} \text{ where}$$

$$s^{a^h} = t^{a^h + 1} R^{b^{a^h + 1}} \mathfrak{s}^{a^{b^{a^h + 1}},a^h + 1} \ldots R^1 \mathfrak{s}^{a^1,a^h + 1} s^{a^h + 1}$$

$$\text{for } b^{a^h + 1} \neq 0 \text{ and}$$

$$= t^{a^h + 1} s^{a^h + 1} \quad \text{for } b^{a^h + 1} = 0$$

$$\text{and } q^{i^e} = \mathrm{MIN}(c'^e) = \mathrm{MIN}(c^{i^e})$$

But Definition 6.2

$$\Rightarrow s^{a^h} = \mathfrak{s}^{a^h,i^e} \text{ by construction}$$

$$\Rightarrow q^{a^h} \xrightarrow{\mathfrak{s}^{a^h,i^e}} q^{i^e} \text{ where } \hat{D}^{a^h,i^e} = q^{i^e} - q^{a^h}$$

<u>end</u> INDUCTION 2

$$q^{a^h} \leq c^{a^h},\ q^{a^h} \xrightarrow{\mathfrak{s}^{a^h,i^e}}$$

$$\Rightarrow c^{a^h} \xrightarrow{\mathfrak{s}^{a^h,i^e}}$$

$c^{a^h} \ll c'^e$ and Lemma 5.21

$$\Rightarrow c'^e \xrightarrow{\mathfrak{s}^{a^h,i^e}}$$

Next it will be shown, by induction upon h: $0 \leq h \leq b^e$, that

$$c'^e \rightsquigarrow c^{i^e}$$

and further that for any $q^e \in c^{i^e}$ there exists a $q'^e \in c'^e$ such that

$$q'^e \xrightarrow{s^1} q^e$$

where

$$q'^e = q^e - R^{b^e}\hat{D}^{a^{b^e},i^e} - \dots - R^1\hat{D}^{a^1,i^e}$$

and

$$s'^e = R^{b^e}\hat{S}^{a^{b^e},i^e} \dots R^1\hat{S}^{a^1,i^e}$$

<u>begin</u> INDUCTION 3

For convenience, $c^{i^e,0} = c'^e$.

Each $c^{i^e,h}$ for h: $1 \leq h \leq b^e$ will represent the cube defined from $c^{i^e,h-1}$ by adding the h^{th} +-backpointer (and the corresponding +-component).

$$\text{i.e. } c_j^{i^e,h} = c_j^{i^e,h-1} \quad \text{for all } j: j \neq m_h \text{ and}$$
$$= [c_j^{i^e,h-1}]^+ \quad \text{for some } j: j = m_h$$

Then $c^{i^e,b^e} = c^{i^e}$.

<u>begin</u> <u>BASE</u> <u>3</u>

For h = 0, it follows from Definition 6.15 that

$$c^{i^e,0} \rightsquigarrow c^{i^e,h}$$

i.e. that for any $q^0 \in c^{i^e,0}$, there exists a $q^h \in c^{i^e,h}$ and some transition sequence, say s^2 = null transition sequence, such that $q^0 \xrightarrow{s^2} q^h$

i.e. $q^0 = q^h$ and

s^2 = null transition sequence

<u>end</u> BASE 3

begin STEP 3

The induction hypothesis will be that:

$$c^{i^e,0} \rightsquigarrow c^{i^e,h-1}$$

i.e. that for any $q^{h-1} \leqslant c^{i^e,h-1}$, there exists a $q^0 \leqslant c^{i^e,0}$ and some transition sequence, say s^2 such that $q^0 \xrightarrow{s^2} q^{h-1}$

and furthermore

if h = 1 that

$$q^0 = q^{h-1}$$

and

$$s^2 = \text{null transition sequence}$$

and if h > 1 that

$$q^0 = q^{h-1} - R^{h-1}\hat{D}a^{h-1,i^e} - \ldots - R^1\hat{D}a^{1,i^e}$$

and

$$s^2 = R^{h-1}\underset{-}{s}a^{h-1,i^e} \ldots R^1\underset{-}{s}a^{1,i^e}$$

and

$$R^k = q^k_{m_k} - \text{glb}(c^{i^e,k}_{m_k}) = q^k_{m_k} - c^{i^e,0}_{m_k} \text{ for } k\text{: } 1 \leq k \leq h-1$$

It will be shown that

$$c^{i^e,0} \leadsto c^{i^e,h}$$

i.e. that for any $q^h \lessdot c^{i^e,h}$, there exists a $q^0 \lessdot c^{i^e,0}$ and some transition sequence, say s^3 such that $q^0 \xrightarrow{s^3} q^h$

and furthermore that

$$q^0 = q^h - R^h\hat{D}a^{h,i^e} - \dots - R^1\hat{D}a^{1,i^e}$$

and

$$s^3 = R^h\hat{S}a^{h,i^e} \dots R^1\hat{S}a^{1,i^e}$$

and

$$R^k = q^k_{m_k} - \text{glb}(c^{i^e,k}_{m_k}) = q^k_{m_k} - c^{i^e,0}_{m_k} \text{ for } k\text{: } 1 \leq k \leq h$$

The induction hypothesis

$$\Rightarrow c^{i^e,0} \leadsto c^{i^e,h-1}$$

Lemma 6.17

$$\Rightarrow \text{to show } c^{i^e,0} \leadsto c^{i^e,h},$$

it is sufficient to show $c^{i^e,h-1} \leadsto c^{i^e,h}$

i.e. that for any $q^h \lessdot c^{i^e,h}$, there exists a $q^{h-1} \lessdot c^{i^e,h-1}$ and some transition sequence, say s^4 such that $q^{h-1} \xrightarrow{s^3} q^h$

It will in fact be shown that

$$q^{h-1} = q^h - R^h\hat{D}a^{h,i^e} \text{ and}$$

$$s^4 = r^h\hat{S}a^{h,i^e}$$

are the required state and transition sequence.

It will first be shown that for any $q^h \leqslant c^{i^e,h}$,

q^{h-1} defined as above is $\leqslant c^{i^e,h-1}$.

Letting q^h be any state such that $q^h \leqslant c^{i^e,h}$

$$\Rightarrow q_j^h = c_j^{i^e,h} \quad \text{for } c_j^{i^e,h} \leqslant N \text{ and}$$
$$\geq \text{glb}(c_j^{i^e,h}) \text{ for } c_j^{i^e,h} \leqslant N^+$$

definition of $c^{i^e,h}$ in terms of $c^{i^e,h-1}$

$$\Rightarrow c_j^{i^e,h} = c_j^{i^e,h-1} \quad \text{for all } j: j \neq m_h \text{ and}$$

$$= [c_j^{i^e,h-1}]^+ \text{ for one } j: j = m_h$$

$$\Rightarrow q_j^h = c_j^{i^e,h-1} \quad \text{for } c_j^{i^e,h-1} \leqslant N \text{ and all } j: j \neq m_h$$

$$\geq c_j^{i^e,h-1} \quad \text{for } c_j^{i^e,h-1} \leqslant N \text{ and some } j: j = m_h$$

$$\geq \text{glb}(c_j^{i^e,h-1}) \text{ for } c_j^{i^e,h-1} \leqslant N^+$$

Letting $R^h = q_j^h - c_j^{i^e,h-1}$

$\Rightarrow$ There are 2 cases to examine

a) $R^h = 0$ (i.e. $q_j^h = c_j^{i^e,h-1}$)

b) $R^h \neq 0$ (i.e. $q_j^h \neq c_j^{i^e,h-1}$)

$R^h = 0$ and the above definitions for q^{h-1} and s^4

$\Rightarrow q^{h-1} = q^h$ and s^4 = null transition sequence

which gives the desired result.

$R^h \neq 0$

$$\Rightarrow q_j^h = c_j^{i^e,h-1} \quad \text{for } c_j^{i^e,h-1} \in N \text{ and all } j: j \neq m_h$$

$$\geq c_j^{i^e,h-1} + R^h \text{ for } c_j^{i^e,h-1} \in N \text{ and some } j: j = m_h$$

$$\geq \mathrm{glb}(c_j^{i^e,h-1}) \text{ for } c_j^{i^e,h-1} \in N^+$$

$$\Rightarrow R^h = q_j^h - c_j^{i^e,h-1} = q_j^h - \mathrm{glb}(c_j^{i^e,h}) = q_j^h - c_j^{i^e,0}$$

For convenience, let

$\hat{s}$ stand for $\hat{s}^{a^h,i^e}$,

$\hat{D}$ stand for $\hat{D}^{a^h,i^e}$ and

$\hat{I}$ stand for $\hat{I}^{a^h,i^e}$

Letting $q^{h-1} = q^h - R^h\hat{D}$

$$\Rightarrow q_j^{h-1} = c_j^{i^e,h-1} - R^h\hat{D} \qquad \text{for } c_j^{i^e,h-1} \in N \text{ and all } j: j \neq m_h$$

$$= c_j^{i^e,h-1} + R^h - R^h\hat{D} \quad \text{for } c_j^{i^e,h-1} \in N \text{ and one } j: j = m_h$$

$$\geq \mathrm{glb}(c_j^{i^e,h-1}) - R^h\hat{D} \quad \text{for } c_j^{i^e,h-1} \in N^+$$

Expressing $c^{i^e,h-1}$ in terms of $c^{i^e,0}$

$$\Rightarrow c_j^{i^e,h-1} = [c_j^{i^e,0}]^+ \text{ for } c_j^{i^e,0} \in N,\ c_j^{i^e,h-1} \in N^+$$

$$= c_j^{i^e,0} \quad \text{for all other cases}$$

but $\hat{D}_j = 0$ for $c_j^{i^e,0} \in N$ and all j: $j \neq m_h$

$= 1$ for $c_j^{i^e,0} \in N$ and one j: $j = m_h$

≤ 0 for $c_j^{i^e,0} \in N^+$

$\Rightarrow \hat{D}_j = 0$ for $c_j^{i^e,0} \in N$ and $c_j^{i^e,h-1} \in N^+$

$= 0$ for $c_j^{i^e,0} \in N$, $c_j^{i^e,h-1} \in N$ and $j \neq m_h$

$= 1$ for $c_j^{i^e,0} \in N$, $c_j^{i^e,h-1} \in N$ and $j = m_h$

≤ 0 for $c_j^{i^e,0} \in N^+$ and $c_j^{i^e,h-1} \in N^+$

$\Rightarrow \hat{D}_j = 0$ for $c_j^{i^e,h-1} \in N$ and all j: $j \neq m_h$

$= 1$ for $c_j^{i^e,h-1} \in N$ and one j: $j = m_h$

≤ 0 for $c_j^{i^e,h-1} \in N^+$

$\Rightarrow q_j^{h-1} = c_j^{i^e,h-1} - 0$ for $c_j^{i^e,h-1} \in N$ and all j: $j \neq m_h$

$= c_j^{i^e,h-1} + R^h - R^h$ for $c_j^{i^1,0} \in N$ and one j: $j = m_h$

$\geq \mathrm{glb}(c_j^{i^e,h-1}) - R^h\hat{D}$ for $c_j^{i^e,h-1} \in N^+$

$\Rightarrow q_j^{h-1} = c_j^{i^e,h-1}$ for $c_j^{i^e,h-1} \in N$

$\geq \mathrm{glb}(c_j^{i^e,h-1})$ for $c_j^{i^e,h-1} \in N^+$

$\Rightarrow q^{h-1} \in c^{i^e,h-1}$

Thus it has been shown that for any $q^h \in c^{i^e,h}$, q^{h-1} as defined is $\in c^{i^e,h-1}$.

It will now be shown that $q^{h-1} \xrightarrow{\hat{s}^{a^h, i^e}}$

For $c_j^{i^e,h-1} \in N$,

$q^{h-1} \in c^{i^e,h-1}$

$\Rightarrow q_j^{h-1} = c_j^{i^e,h-1}$

$c^{i^e,h-1} \xrightarrow{\hat{s}}$

$\Rightarrow c_j^{i^e,h-1} - \hat{I}_j \geq 0$

$\Rightarrow q_j^{h-1} - \hat{I}_j \geq 0$

For $c_j^{i^e,h-1} \in N^+$,

$q^{h-1} \in c^{i^e,h-1}$

$\Rightarrow q_j^{h-1} \geq \mathrm{glb}(c_j^{i^e,h-1})$

definition of $c^{i^e,h-1}$

$\Rightarrow c^{i^e,0} << c^{i^e,h-1}$

$\Rightarrow q_j^{h-1} \geq \mathrm{glb}(c_j^{i^e,0})$

$q^{a^h} \xrightarrow{\hat{s}} \mathrm{MIN}(c^{i^e,0})$

$\Rightarrow \mathrm{glb}(c_j^{i^e,0}) = q_j^{a^h} + \hat{D}_j$ and $q_j^{a^h} \geq \hat{I}_j$

definition of $c^{i^e,h}$ and $c^{i^e,h-1}$

$$\Rightarrow MIN(c^{i^e,h-1}) = MIN(c^{i^e,0})$$

$$\Rightarrow glb(c_j^{i^e,h-1}) = q_j^{a^h} + \hat{D}_j$$

$$q^{h-1} = q^h - R^h\hat{D}$$

$$\Rightarrow q^h = q^{h-1} + R^h\hat{D}$$

$$q^h \in c^{i^e,h}$$

$$\Rightarrow q^h \geq glb(c_j^{i^e,h})$$

$$\Rightarrow q_j^{h-1} + R^h\hat{D}_j \geq glb(c_j^{i^e,h})$$

but $c^{i^e,h-1} \in N^+$ and definition of $c^{i^e,h}$

$$\Rightarrow c_j^{i^e,h-1} = c_j^{i^e,h}$$

$$\Rightarrow q_j^{h-1} + R^h\hat{D}_j \geq glb(c_j^{i^e,h-1})$$

$$\Rightarrow q_j^{h-1} + R^h\hat{D}_j \geq q_j^{a^h} + \hat{D}_j$$

$$\Rightarrow q_j^{h-1} + R^h\hat{D}_j - \hat{D}_j \geq q_j^{a^h}$$

$$q_j^{a^h} \geq \hat{I}_j$$

$$\Rightarrow q_j^{h-1} + (R^h-1)\hat{D}_j \geq \hat{I}_j$$

For $c_j^{i^e,h-1} \in N^+$, either the corresponding $c_j^{i^e,0} \in N$ or $\in N^+$

$c_j^{i^e,0} \in N$,

$$\Rightarrow \hat{D}_j = 0$$

$c_j^{i^e,0} \in N^+$,

$$\Rightarrow \hat{D}_j \leq 0$$

$$\Rightarrow \hat{D}_j \leq 0 \text{ for } c_j^{i^e,0} \in N^+$$

$R^h \neq 0$

$$\Rightarrow (R^h - 1) \geq 0$$

$\hat{D}_j \leq 0$

$$\Rightarrow (R^h - 1)\hat{D}_j \leq 0$$

$$\Rightarrow q_j^{h-1} \geq \hat{I}_j$$

Thus $q_j^{h-1} \geq \hat{I}_j$ for all $c_j^{i^e,h-1}$ for j: $1 \leq j \leq n$

$$\Rightarrow q^{h-1} \xrightarrow{\hat{s}}$$

It now remains to be shown that

$$q^{h-1} = q^h - R^h \hat{D} a^{h,i^e} \text{ and } s^4 = R^h \hat{s} a^{h,i^e}$$

are such that

$$q^{h-1} \xrightarrow{s^4} q^h.$$

<u>begin</u> INDUCTION 4

Letting $\hat{q}^0 = q^{h-1}$, it will be shown, by induction upon r: $1 \leq r \leq R^h$ that for any r

$$\hat{q}^0 \xrightarrow{r\hat{s}} \hat{q}^r \text{ where } \hat{q}^r = \hat{q}^0 + r\,\hat{D}^{a^h, i^e}$$

<u>begin</u> BASE 4

For r = 1 it will be shown that

a) $\hat{q}^0 \xrightarrow{\hat{s}}$ and

b) there exist a $\hat{q}^1$ such that $\hat{q}^0 \xrightarrow{\hat{s}} \hat{q}^1$ where $\hat{q}^1 = \hat{q}^0 + \hat{D}^{a^h, i^e}$.

$\hat{q}^0 = q^{h-1}$ and $q^{h-1} \xrightarrow{\hat{s}}$

$\Rightarrow \hat{q}^0 \xrightarrow{\hat{s}}$

Definition 2.1 and Lemma 2.4

$\Rightarrow$ There exists a $\hat{q}^1$ such that $\hat{q}^0 \xrightarrow{\hat{s}} q^1$ where $\hat{q}^1 = \hat{q}^0 + \hat{D}$

<u>end</u> BASE 4

<u>begin</u> STEP 4

Assuming as the induction hypothesis for r-1 that

a) $\hat{q}^0 \xrightarrow{(r-1)\hat{s}} \hat{q}^{r-1}$ and

b) $\hat{q}^{r-1} = \hat{q}^0 + (r-1)\hat{D}$

it will be shown that

a) $\hat{q}^0 \xrightarrow{r\hat{s}} \hat{q}^r$ and $\hat{q}^{r-1} \xrightarrow{\hat{s}} \hat{q}^r$

b) $\hat{q}^r = \hat{q}^0 + r\hat{D} = \hat{q}^{r-1} + \hat{D}$

For $c_j^{i^e,0} \in N$,

$\hat{q}_j^{r-1} = \hat{q}_j^0 + (r-1)\hat{D}_j$

$\Rightarrow \hat{q}_j^0 = \hat{q}_j^{r-1} - (r-1)\hat{D}$

$\hat{q}^0 \xrightarrow{\hat{s}}$

$\Rightarrow \hat{q}_j^0 - \hat{I}_j \geq 0$

$\Rightarrow \hat{q}_j^{r-1} - (r-1)\hat{D}_j - \hat{I}_j \geq 0$

$\Rightarrow \hat{q}_j^{r-1} - \hat{I}_j \geq (r-1)\hat{D}_j$

but $\hat{D}_j = 1$ for $c_j^{i^e,0} \in N$ and some j: $j = m_h$ and

$= 0$ for $c_j^{i^e,0} \in N$ and all j: $j \neq m_h$

and $1 \leq r \leq R^h$

$\Rightarrow (r-1)\hat{D}_j \geq 0$

$\Rightarrow \hat{q}_j^{r-1} - \hat{I}_j \geq 0$ for $c_j^{i^e,0} \in N$

For $c_j^{i^e,0} \in N^+$,

definition of $c^{i^e,h}$

$\Rightarrow c_j^{i^e,h} = c_j^{i^e,0}$

$\Rightarrow glb(c_j^{i^e,h}) = glb\ (c_j^{i^e,0})$

$q^h \leqslant c^{i^e,h}$

$\Rightarrow q_j^h \geq glb(c_j^{i^e,h})$

$\Rightarrow q_j^h \geq glb(c_j^{i^e,0})$

$MIN(c') \xrightarrow{\hat{s}}$ and $c' = c^{i^e,0}$

$\Rightarrow glb(c_j^{i^e,0}) - \hat{I}_j \geq 0$

$\Rightarrow q_j^h - \hat{I}_j \geq 0$

$\hat{q}^0 = q^{h-1}$ and $q^{h-1} = q^h - R^h\hat{D}$

$\Rightarrow q^h = \hat{q}^0 + R^h\hat{D}$

$\Rightarrow q_j^0 + R^h\hat{D}_j - \hat{I}_j \geq 0$

$\hat{q}^{r-1} = \hat{q}^0 + (r-1)\hat{D}$

$\Rightarrow \hat{q}^0 = \hat{q}^{r-1} - (r-1)\hat{D}$

$\Rightarrow \hat{q}_j^{r-1} - (r-1)\hat{D}_j + R^h\hat{D}_j - \hat{I}_j \geq 0$

$\Rightarrow \hat{q}_j^{r-1} - (r-1-R^h)\hat{D}_j - \hat{I}_j \geq 0$

$\Rightarrow \hat{q}_j^{r-1} - \hat{I}_j \geq (r-1-R^h)$

$1 \leq r \leq R^h$

$\Rightarrow (r - 1 - R^h) \leq 0$

$c_j^{i^e,0} \leqslant N^+$

$\Rightarrow \hat{D}_j \leq 0$

$\Rightarrow (r - 1 - R^h)\hat{D}_j \geq 0$

$\Rightarrow \hat{q}_j^{r-1} - \hat{I}_j \geq 0$ for j: $c_j^{i^e,0} \leqslant N^+$

Thus $\hat{q}_j^{r-1} - \hat{I}_j \geq 0$ for $j = 1, \ldots, n$

$\Rightarrow \hat{q}^{r-1} \xrightarrow{\hat{s}}$

Definition 2.1 and Lemma 2.4

$\Rightarrow$ there exists some state, say $\hat{q}^r$, such that

$$\hat{q}^{r-1} \xrightarrow{\hat{s}} \hat{q}^r \text{ where } \hat{q}^r = \hat{q}^{r-1} + \hat{D}$$

but the induction hypothesis

$$\Rightarrow \hat{q}^0 \xrightarrow{(r-1)\hat{s}} \hat{q}^{r-1} \text{ where } \hat{q}^{r-1} = \hat{q}^0 + (r-1)\hat{D}$$

$$\Rightarrow \hat{q}^0 \xrightarrow{r\hat{s}} \hat{q}^r \text{ where } \hat{q}^r = \hat{q}^0 + r\,\hat{D}$$

Thus it has been shown that

a) $\hat{q}^0 \xrightarrow{r\hat{s}} \hat{q}^r$ and $\hat{q}^{r-1} \xrightarrow{\hat{s}} \hat{q}^r$

b) $\hat{q}^r = \hat{q}^0 + r\hat{D} = \hat{q}^{r-1} + \hat{D}$

<u>end</u> STEP 4

Thus for any r: $1 \leq r \leq R^h$

$$\hat{q}^0 \xrightarrow{r\hat{s}} \hat{q}^r \text{ where } \hat{q}^r = \hat{q}^0 + r\hat{D}$$

<u>end</u> INDUCTION 4

Letting $r = R^h$

$$\Rightarrow \hat{q}^0 \xrightarrow{R^h\hat{s}} \hat{q}^{R^h} \text{ where } \hat{q}^{R^h} = \hat{q}^0 + R^h\hat{D}$$

$\hat{q}^0 = q^h - R^h\hat{D}$

$$\Rightarrow \hat{q}^{R^h} = q^h - R^h\hat{D} + R^h\hat{D}$$

$$= q^h$$

$$\Rightarrow \hat{q}^0 \xrightarrow{R^h\hat{s}} q^h \text{ where } q^h = \hat{q}^0 + r^h\hat{D}$$

But $\hat{q}^0 = q^{h-1}$

$$\Rightarrow q^{h-1} \xrightarrow{R^h\hat{s}} q^h \text{ where } q^h = q^{h-1} + R^h\,\hat{D}$$

but q^h was any state $\prec c^{i^e,h}$ and q^{h-1} was shown to be $\prec c^{i^e,h-1}$,

thus Definition 6.15

$$\Rightarrow c^{i^e,h-1} \leadsto c^{i^e,h}$$

But the induction hypothesis

$$\Rightarrow c^{i^e,0} \rightsquigarrow c^{i^e,h-1}$$

Lemma 6.17

$$\Rightarrow c^{i^e,0} \rightsquigarrow c^{i^e,h}$$

i.e. that for any $q^h \leqslant c^{i^e,h}$, there exists a $q^0 \leqslant c^{i^e,0}$ and some transition sequence, say s^3, such that $q^0 \xrightarrow{s^3} q^h$.

Furthermore for h = 1, the induction hypothesis

$$\Rightarrow q^0 = q^{h-1} \text{ and } s^2 = \text{null transition sequence}$$

$$q^h = q^{h-1} + R^h\hat{D} \text{ and } q^{h-1} \xrightarrow{R^h\hat{s}} q^h$$

$$\Rightarrow q^1 = q^0 + R^1\hat{D}a^{1,i^e}$$

$$\text{and } s^3 = R^1\hat{s}a^{1,i^e}$$

$$\text{where } R^1 = q^1_{m_1} - \text{glb}(c^{i^e,1}_{m_1}) = q^1_{m_1} - c^{i^e,0}_{m_1}$$

and for $h \neq 1$, the induction hypothesis

$$\Rightarrow q^0 = q^{h-1} - R^{h-1}\hat{D}a^{h-1,i^e} - \ldots - R^1\hat{D}a^{1,i^e}$$

and

$$s^2 = R^{h-1}\hat{s}a^{h-1,i^e} \ldots R^1\hat{s}a^{1,i^e}$$

and

$$R^k = q^k_{m_k} - \text{glb}(c^{i^e,k}_{m_k}) = q^k_{m_k} - c^{i^e,0}_{m_k} \text{ for } k\text{: } 1 \leq k \leq h-1$$

$q^h = q^{h-1} + R^h\hat{D}$ and $q^{h-1} \xrightarrow{R^h \mathcal{S}} q^h$

$$\Rightarrow q^0 = q^h - R^h\hat{D}a^{h,i^e} - \dots - R^1\hat{D}a^{1,i^e}$$

and

$$s^3 = R^h \mathcal{S} a^{h,i^e} \dots R^1 \mathcal{S} a^{1,i^e}$$

and

$$R^k = q^k_{m_k} - \mathrm{glb}(c^{i^e,k}_{m_k}) = q^k_{m_k} - c^{i^e,0}_{m_k} \text{ for } k\text{: } 1 \leq k \leq h$$

Thus it has been shown that

$$c^{i^e,0} \rightsquigarrow c^{i^1,h}$$

i.e. that for any $q^h \in c^{i^e,h}$, there exists a $q^0 \in c^{i^e,0}$ and some transition sequence, say s^3 such that $q^0 \xrightarrow{s^3} q^h$

and furthermore that for any h,

$$q^0 = q^h - R^h\hat{D}a^{h,i^e} - \dots - R^1\hat{D}a^{1,i^e}$$

and

$$s^3 = R^h \mathcal{S} a^{h,i^e} \dots R^1 \mathcal{S} a^{1,i^e}$$

and

$$R^k = q^k_{m_k} - \mathrm{glb}(c^{i^e,k}_{m_k}) = q^k_{m_k} - c^{i^e,0}_{m_k} \text{ for } k\text{: } 1 \leq k \leq h$$

<u>end</u> STEP 3

Letting $h = b^e$

$\Rightarrow q^0 \xrightarrow{s^3} q^e$ where

$$q^0 = q^e - R^{b^e}\hat{D}^{a^{b^e},i^e} - \ldots - R^1\hat{D}^{a^1,i^e}$$

and

$$s^3 = R^{b^e}g^{a^{b^e},i^e} \ldots R^1 g^{a^1,i^e}$$

and

$$R^k = q^k_{m_k} - glb(c^{i^e,k}_{m_k}) = q^k_{m_k} - c^{i^e,0}_{m_k} \text{ for } k:\ 1 \leq k \leq b^e$$

and

$$c^{i^e,0} \rightsquigarrow c^{i^e,b^e}$$

But $c^{i^e,0} = c'^e$ and $c^{i^e,b^e} = c^{i^e}$

$\Rightarrow c'^e \rightsquigarrow c^{i^e}$

Thus it has been shown that for any $q^e \leqslant c^{i^e}$ there exists a $q'^e \leqslant c'^e$

such that

$$q'^e \xrightarrow{s'^e} q^e$$

where

$$q'^e = q^e - R^{b^e}\hat{D}^{a^{b^e},i^e} - \ldots - R^1\hat{D}^{a^1,i^e}$$

and

$$s'^e = R^{b^e}g^{a^{b^e},i^e} \ldots R^1 g^{a^1,i^e}$$

<u>end</u> INDUCTION 3

But it has already been shown that

$$c^{i^0} \rightsquigarrow c'^e$$

Therefore $c'^e \rightsquigarrow c^{i^e}$ and Lemma 6.17

$\Rightarrow c^{i^0} \rightsquigarrow c^{i^e}$

Thus it has been shown that for any e,

$$c^{i^0} \rightsquigarrow c^{i^e}$$

and

$g^{a^{h^e}, i^e}$ as defined in Definition 6.2 exists for each $c^{a^{h^e}}$ such that there is a $^+$-backpointer from c^{i^e} to $c^{a^{h^e}}$, h^e: $1 \leq h^e \leq b^e$.

and

for any $q^e \leqslant c^{i^e}$ and c'^e corresponding to c^{i^e} as per Definition 6.2, there exists a $q'^e \leqslant c'^e$ such that $q'^e \xrightarrow{s'^e} q^e$

where

$$q'^e = q^e - R^{b^e}\hat{D}^{a^{b^e}, i^e} - \ldots - R^1\hat{D}^{a^1, i^e}$$

and

$$s'^e = R^{b^e} g^{a^{b^e}, i^e} \ldots R^1 g^{a^1, i^e}$$

<u>end</u> STEP 1

Letting e = g

$$\Rightarrow c^{i^0} \rightsquigarrow c^{i^g}$$

<u>end</u> INDUCTION 1

APPENDIX II

PROGRAM OUTPUT FOR PRODUCER/CONSUMER EXAMPLE

```
NO OF TRANSITIONS=    4    NO OF PLACES=    5
                           PLACES --->
           TRANSITION #    1 2 3 4 5
                       1 I 1 0 0 0 0
                         0 0 1 0 0 0
                       2 I 0 1 0 0 0
                         0 1 0 1 0 0
                       3 I 0 0 1 1 0
                         0 0 0 0 0 1
                       4 I 0 0 0 0 1
                         0 0 0 0 1 0
          INITIAL MARKING 1 0 0 1 0
?
t1    (produce Reachability Tree)
          25 STATE(S):           25 ACTUAL AND     0 DUMMIES
PRINT STATE TABLE? - Y OR N
y
 S       T    R   TR          TREE - LAUCHT
 T   LL  RL  LI   RI   LB T   VERSION DATE:791121
 A   IE  AE  IG   AG   IA Y PLACES --->
 TN  NF  NF  NH   NH   NC P
 EO  KT  ST  KT   ST   KK E 1 2 3 4 5
  1   2   1   0    0    0 I 1 0 0 1 0
  2   3   2   0    0    1 I 0 1 0 1 0
  3   4   1   5    3    2 I 1 0 1+1 0
  4  18   2  19    3    3 I 0 1 1+1 0
  5   6   1   7    4    3 I 1 0 0+0 1
  6  14   2  15    4    5 I 0 1 0+0 1
  7   8   1   9    3    5 I 1 0 0+1 0
  8  10   2  11    3    7 I 0 1 0+1 0
  9   5   0   0    0    7 L 1 0 0+0 1
 10   7   0   0    0    8 L 1 0 1+1 0
 11  12   2  13    4    8 I 0 1 0+0 1
 12   5   0   0    0   11 L 1 0 1+0 1
 13   8   0   0    0   11 L 0 1 0+1 0
 14   5   0   0    0    6 L 1 0 1+0 1
 15  16   2  17    3    6 I 0 1 0+1 0
 16   3   0   0    0   15 L 1 0 1+1 0
 17   6   0   0    0   15 L 0 1 0+0 1
 18   3   0   0    0    4 L 1 0 2+1 0
 19  20   2  21    4    4 I 0 1 0+0 1
 20  24   1  25    4   19 I 1 0 1+0 1
 21  22   2  23    3   19 I 0 1 0+1 0
 22   3   0   0    0   21 L 1 0 1+1 0
 23  19   0   0    0   21 L 0 1 0+0 1
 24  19   0   0    0   20 L 0 1 1+0 1
 25   3   0   0    0   20 L 1 0 1+1 0
          25 STATE(S):           25 ACTUAL AND     0 DUMMIES
```

```
a    (analyze)
 MG SM EMG ESM CFN FCN SELF-LOOP MULTI-ARC
  T  F  T   F   T   T     F         F
?
rs    (simplify to yield Reachability Set)
           4 STATE(S)
PRINT STATE TABLE? - Y OR N
y
TREE - LAUCHT                   REACHABILITY SET - LAUCHT
                              VERSION DATE:791121
                           PLACES --->
                           1 2 3 4 5
                         1 1 0 0+0 1
                         2 0 1 0+0 1
                         3 1 0 0+1 0
                         4 0 1 0+1 0
?
gl    (produce Reachability Graph instead of Tree)
          13 STATE(S):          13 ACTUAL AND    0 DUMMIES
PRINT STATE TABLE? - Y OR N
y
 S       T    R   TR          GRAPH - LAUCHT
 T   LL  RL  LI  RI  LB T     VERSION DATE:791121
 A   IE  AE  IG  AG  IA Y PLACES --->
 TN  NF  NF  NH  NH  NC P
 EO  KT  ST  KT  ST  KK E 1 2 3 4 5
  1   2   1   0   0   0 I 1 0 0 1 0
  2   3   2   0   0   1 I 0 1 0 1 0
  3   4   1   5   3   2 I 1 0 1+1 0
  4   8   0   0   0   3 L 0 1 1+1 0
  5   6   1   7   4   3 I 1 0 0+0 1
  6  11   0   0   0   5 L 0 1 0+0 1
  7   8   1   9   3   5 I 1 0 0+1 0
  8  10   2  11   3   7 I 0 1 0+1 0
  9   5   0   0   0   7 L 1 0 0+0 1
 10   3   0   0   0   8 L 1 0 1+1 0
 11  12   2  13   4   8 I 0 1 0+0 1
 12   5   0   0   0  11 L 1 0 1+0 1
 13   8   0   0   0  11 L 0 1 0+1 0
          13 STATE(S):          13 ACTUAL AND    0 DUMMIES
?
rs    (simplify to yield Reachability Set)
           4 STATE(S)
PRINT STATE TABLE? - Y OR N
y
GRAPH - LAUCHT                  REACHABILITY SET - LAUCHT
                              VERSION DATE:791121
                           PLACES --->
                           1 2 3 4 5
                         1 1 0 0+0 1
                         2 1 0 0+1 0
                         3 0 1 0+1 0
                         4 0 1 0+0 1
```