CAMBRIDGE STUDIES IN
ADVANCED MATHEMATICS 1

Algebraic automata theory

Algebraic automata theory

W.M.L.HOLCOMBE
Department of Pure Mathematics, The Queen's University of Belfast

CAMBRIDGE UNIVERSITY PRESS
CAMBRIDGE
LONDON NEW YORK NEW ROCHELLE
MELBOURNE SYDNEY

PUBLISHED BY THE PRESS SYNDICATE OF THE UNIVERSITY OF CAMBRIDGE
The Pitt Building, Trumpington Street, Cambridge, United Kingdom

CAMBRIDGE UNIVERSITY PRESS
The Edinburgh Building, Cambridge CB2 2RU, UK
40 West 20th Street, New York NY 10011–4211, USA
477 Williamstown Road, Port Melbourne, VIC 3207, Australia
Ruiz de Alarcón 13, 28014 Madrid, Spain
Dock House, The Waterfront, Cape Town 8001, South Africa

http://www.cambridge.org

First published 1982
First paperback edition 2004

A catalogue record for this book is available from the British Library

Library of Congress catalogue card number: 81-18169

ISBN 0 521 23196 5 hardback
ISBN 0 521 60492 3 paperback

To Jill, Lucy, and my mother, and in fond memory of my father and grandfather

Contents

Introduction

In recent years there has been a growing awareness that many complex processes can be regarded as behaving rather like machines. The theory of machines that has developed in the last twenty or so years has had a considerable influence, not only on the development of computer systems and their associated languages and software, but also in biology, psychology, biochemistry, etc. The so-called 'cybernetic view' has been of tremendous value in fundamental research in many different areas. Underlying all this work is the mathematical theory of various types of machine. It is this subject that we will be studying here, along with examples of its applications in theoretical biology, etc.

The area of mathematics that is of most use to us is that which is known as modern (or abstract) algebra. For a hundred years or more, algebra has developed enormously in many different directions. These all had origins in difficult problems in the theory of equations, number theory, geometry, etc. but in many areas the subject has taken on its own momentum, the problems arising from within the subject, and as a result there has been a general feeling that much of abstract algebra is of little practical value. The advent of the theory of machines, however, has provided us with new motivation for the development of algebra since it raises very real practical problems that can be examined using many of the abstract tools that have been developed in algebra. This, to me, is the most exciting aspect of the subject, the ability of using algebra in a useful and meaningful way to tackle some of the fundamental questions facing us today: What can machines do? How do we think and speak? How do cells repair themselves? How do biochemical systems function? How do organisms grow and develop? etc. We will not be able to answer these questions here, that is neither possible nor the aim

of this book. What we will be doing is to lay the foundations for the algebraic study of machines, by looking at various types of machines, their properties, and ways in which complex machines can be simulated by simpler machines joined together in some way. This will then provide a theoretical framework for the more detailed analysis of the applications of machine theory in these subjects, with the ultimate aim of explaining many natural and artificial phenomena. Perhaps a later work devoted to the applications of machine theory would make use of the developments outlined here.

We begin with some elementary material concerning the theory of semigroups. This is presented as concisely as possible; it may be omitted by those readers familiar with the material. Others could easily start by reading the first few pages of chapter 2, which introduces the state machine, before returning (hopefully better motivated) to the details of chapter 1.

The second chapter examines many elementary properties of the state machine, the ways in which it can be connected together with others, and finishes with some applications. I have tried to include as wide a variety as possible and I have not treated them in great depth because the required background knowledge in biology, biochemistry, etc. may not be available. For those interested, the references provide sources of further reading.

Chapter 3 develops the idea of a covering, by which state machines can be simulated by other, perhaps simpler, state machines in various configurations. This area represents a major change in philosophy in algebra since we do not attempt to describe the machine exactly but rather what it can do. There are some general results that enable us to start with an arbitrary state machine and simulate it with simpler machines constructed from finite simple groups and elementary 'two-state' machines connected up suitably. The best known method for doing this, the holonomy decomposition, is examined in chapter 4. However, this process leads to simulating machines that can be very large and thus relatively inefficient. In specific situations it is possible to develop much better simulators and a variety of techniques for doing this are examined in chapter 3.

The theory of recognizers is intimately connected with the theory of state machines and is of considerable importance in the theory of computers. This area is studied in chapter 5.

Finally we end with a more practical and realistic type of machine and apply the previous results to this situation in chapter 6.

Some of this material would be suitable for an advanced undergraduate course on applied algebra or automata theory and I have indeed given such a course for some years at Queen's University, Belfast. The more advanced material (chapters 4, 6) would be suitable for a graduate course.

I hope that this book can help forge links between pure mathematicians, computer scientists and theoretical biologists. There are great benefits in a dialogue between practitioners of these subjects and although I realize that the approach here is rather mathematical, I hope that it will not prevent others from making use of the material. With this in mind I have included as an appendix a computer program for evaluating the semigroup of a state machine. This has been developed for me by Dr A. W. Wickstead (Pure Mathematics, Q.U.B.) and I would like to take this opportunity of thanking him for his help. The program is suitable for use on a microcomputer, something that is becoming readily available these days.

My other thanks go to many of my colleagues at Queen's who have helped me with various questions. As usual, though, I have to take responsibility for any errors that may occur in this work.

Sheila O'Brien (Q.U.B.) made an excellent job of typing my manuscript and I would like to record my gratitude here.

I must also thank Dr E. Dilger (Tübingen) for reading the manuscript and Dr B. McMaster for helping me with the proofs.

Michael Holcombe (*July 1981*)

1

Semigroups and their relatives

We may as well begin at the beginning and this will involve us in a brief excursion through some of the fundamental concepts essential for any algebraic subject. It will also enable us to become acquainted with the notation used, although the experienced reader could easily skip through this chapter. We will assume that the reader has a knowledge of elementary set theory.

1.1 Relations

One of the fundamental concepts in mathematics is that of a relation. It can be introduced in a variety of ways but the most useful one for us is the following abstract approach.

Let A be a non-empty set. A *relation*, $\mathcal{R}$, on A is a subset $\mathcal{R} \subseteq A \times A$. If $(a, a') \in A \times A$ and $(a, a') \in \mathcal{R}$ we say that a is *$\mathcal{R}$-related* to a'. Sometimes a natural notation is used in mathematics to express this relationship between two elements of a set, for example if $A = \mathbb{Z}$, the set of all integers, then there is a relation $\leq$ that can be defined on $\mathbb{Z}$. We write $a \leq a'$ if the number $a' - a$ is not negative and the set $\mathcal{R} \subseteq \mathbb{Z} \times \mathbb{Z}$ defining this relation consists of all ordered pairs $(a, a') \in \mathbb{Z} \times \mathbb{Z}$ such that $a \leq a'$.

A relation $\mathcal{R}$ on the set A is an *equivalence relation* if:

(i) $(a, a) \in \mathcal{R}$ for all $a \in A$
(ii) $(a, a') \in \mathcal{R} \Rightarrow (a', a) \in \mathcal{R}$ for $a, a' \in A$
(iii) $(a, a') \in \mathcal{R}$ and $(a', a'') \in \mathcal{R} \Rightarrow (a, a'') \in \mathcal{R}$ for $a, a', a'' \in A$.

The existence of an equivalence relation is a very useful fact because it means that we can partition the set A into a disjoint union of subsets in a natural way.

Let $\mathcal{R}$ be an equivalence relation on the set A, for each $a \in A$ define the set $[a]_{\mathcal{R}} = \{a' \in A | (a, a') \in \mathcal{R}\}$, so $[a]_{\mathcal{R}}$ is the subset of all elements of A that are related to a under $\mathcal{R}$. It is called the *equivalence class* defined by a. If the relation $\mathcal{R}$ is understood we just write $[a]$.

Consider now the collection of all the *distinct* subsets of the form $[a]$ where $a \in A$. If we denote this collection by the notation $A/\mathcal{R}$ we can establish the following:

Theorem 1.1.1

Let $\mathcal{R}$ be an equivalence relation on the set A. The set $A/\mathcal{R}$ consists of a collection of pairwise disjoint subsets of A that cover A. By this we mean that if the collection $A/\mathcal{R}$ is indexed by some set I so that

$$A/\mathcal{R} = \{H_i \mid i \in I\} \quad \text{where each } H_i \text{ is of the form } [a]$$

for some $a \in A$ then

(i) $\bigcup_{i \in I} H_i = A$

(ii) $H_i \cap H_j = \varnothing \quad \text{if } i \neq j \ (i, j \in I)$.

Proof Let $a \in A$, then $(a, a) \in \mathcal{R}$ and so $a \in [a]$ and thus there is $i \in I$ such that $[a] = H_i$. Hence $a \in H_i$ for some $i \in I$. This proves (i). Now suppose that $a \in H_i$ for some $i \in I$, then there exists $a' \in A$ such that $a \in [a']$ where $H_i = [a']$. Let $b \in [a]$, then $(a, b) \in \mathcal{R}$, but $(a', a') \in \mathcal{R}$ and so $(a', b) \in \mathcal{R}$, which means $b \in [a']$. Hence $[a] \subseteq [a']$. Let $c \in [a']$, then $(a', c) \in \mathcal{R}$, however $(a', a) \in \mathcal{R}$ implies $(a, a') \in \mathcal{R}$ and then $(a, c) \in \mathcal{R}$ and $c \in [a]$. Hence $[a] = [a']$. Finally choose $d \in H_i \cap H_j$ where $H_i = [a']$ and $H_j = [a'']$. Since $[a'] = [d]$ and $[a''] = [d]$ from the above it is clear that $H_i = H_j$ and so $i = j$. This proves (ii). □

The set $A/\mathcal{R}$ is called the *quotient set of A with respect to $\mathcal{R}$*.

If A is a set and $\pi = \{H_i \mid i \in I\}$ is a collection of subsets of A satisfying (i) $\bigcup_{i \in I} H_i = A$ and (ii) $H_i \cap H_j \doteq \varnothing$ if $i \neq j$, $(i, j \in I)$; then π is called a *partition* of A. We call the subsets H_i $(i \in I)$, the *π-blocks*. Clearly an equivalence relation defines a partition based upon the distinct equivalence classes. Conversely, given a partition $\pi = \{H_i \mid i \in I\}$ we can define an equivalence relation $\mathcal{R}$ in the following way:

$$a \mathcal{R} a' \Leftrightarrow \text{there exists } i \in I \text{ such that } a \in H_i \text{ and } a' \in H_i.$$

So two elements are equivalent precisely when they belong to the same π-block. It will sometimes be convenient to identify an equivalence relation with its partition.

Associated with a general relation $\mathcal{R}$ on a set A are two subsets of A defined as follows:

$$\mathfrak{D}(\mathcal{R}) = \{a \in A \mid (a, b) \in \mathcal{R} \text{ for some } b \in A\}$$
$$\mathfrak{R}(\mathcal{R}) = \{a \in A \mid (b, a) \in \mathcal{R} \text{ for some } b \in A\}.$$

We call $\mathfrak{D}(\mathcal{R})$ the *domain* of $\mathcal{R}$ and $\mathfrak{R}(\mathcal{R})$ the *range* of $\mathcal{R}$. If $\mathcal{R}$ is an equivalence relation then $\mathfrak{D}(\mathcal{R}) = \mathfrak{R}(\mathcal{R}) = A$.

One of the benefits of taking this approach to relations is the ease of generalizing ideas to relations *between* sets.

Let X and Y be sets, a *relation $\mathcal{R}$ from X to Y* is a subset $\mathcal{R} \subseteq X \times Y$. As before we will say that elements (x, y) belonging to $\mathcal{R}$ are *$\mathcal{R}$-related*, $(x \in X, y \in Y)$. Let us denote this relation by the symbols $\mathcal{R} : X \rightsquigarrow Y$.

There are certain 'extreme' situations that we will briefly examine now. We no longer prevent the sets X and Y from being empty. If either X or Y or both are empty then $X \times Y$ is *defined* to be the empty set. So that we can certainly define a relation from the empty set $\varnothing$ to a set Y, or from a set X to the empty set $\varnothing$, in both cases $\mathcal{R} = \varnothing$ is the only possible relation. In fact the *empty relation* $\mathcal{R} = \varnothing$ can be defined from any set X to any set Y, we write it as $\boldsymbol{\theta} : X \to Y$.

As before we can define the concepts of domain and range, thus if $\mathcal{R} : X \rightsquigarrow Y$ is a relation then

$$\mathfrak{D}(\mathcal{R}) = \{x \in X \mid (x, y) \in \mathcal{R} \text{ for some } y \in Y\}$$
$$\mathfrak{R}(\mathcal{R}) = \{y \in Y \mid (x, y) \in \mathcal{R} \text{ for some } x \in X\}$$

Clearly $\mathfrak{D}(\mathcal{R}) \subseteq X$, $\mathfrak{R}(\mathcal{R}) \subseteq Y$ and one or both may be $\varnothing$.

Suppose, now, that $\mathcal{R} : X \rightsquigarrow Y$ is a relation, so that $\mathcal{R} \subseteq X \times Y$. Define a relation $\mathcal{R}^{-1} : Y \rightsquigarrow X$ as follows:

$$\mathcal{R}^{-1} = \{(y, x) \in Y \times X \mid (x, y) \in \mathcal{R}\} \subseteq Y \times X.$$

We call $\mathcal{R}^{-1}$ the *inverse relation* of $\mathcal{R}$. Then $\mathfrak{D}(\mathcal{R}^{-1}) = \mathfrak{R}(\mathcal{R})$ and $\mathfrak{R}(\mathcal{R}^{-1}) = \mathfrak{D}(\mathcal{R})$.

A function (or mapping) is a special type of relation. Let $\mathcal{R} : X \rightsquigarrow Y$ be a relation such that

$$\text{if } (x, y) \in \mathcal{R} \text{ and } (x, y') \in \mathcal{R} \text{ then } y = y',$$

where $x \in X$; $y, y' \in Y$. We call $\mathcal{R}$ a *partial function* and write it as $\mathcal{R} : X \to Y$. A relation is thus a partial function if each element of the domain $\mathfrak{D}(\mathcal{R})$ is related to exactly one element of the range $\mathfrak{R}(\mathcal{R})$.

A *function* is a partial function $\mathcal{R} : X \to Y$ such that $\mathfrak{D}(\mathcal{R}) = X$.

Example 1.1

Let $X=\{a, b, c\}$, $Y=\{w, x, y, z\}$. If $\mathscr{R}_1: X \rightsquigarrow Y$ is defined by $\mathscr{R}_1=\{(a, w), (a, x), (c, w)\}$ then $\mathscr{R}_1$ is a relation, it is not a partial function since $(a, w) \in \mathscr{R}_1$ and $(a, x) \in \mathscr{R}_1$. Also note that

$$\mathscr{D}(\mathscr{R}_1)=\{a, c\}, \quad \mathscr{R}(\mathscr{R}_1)=\{w, x\}$$

$\mathscr{R}_1^{-1}: Y \rightsquigarrow X$ is given by $\mathscr{R}_1^{-1}=\{(w, a), (x, a), (w, c)\}$.

Now let $R_2: X \rightsquigarrow Y$ be defined by $\mathscr{R}_2=\{(a, w), (b, w)\}$ then $\mathscr{R}_2$ is a partial function, $\mathscr{D}(\mathscr{R}_2)=\{a, b\} \neq X$, $\mathscr{R}(\mathscr{R}_2)=\{w\} \neq Y$ and $\mathscr{R}_2^{-1}: Y \rightsquigarrow X$ is given by $\mathscr{R}_2^{-1}=\{(w, a), (w, b)\}$. Note that $\mathscr{R}_2^{-1}$ is not a partial function.

Finally define $\mathscr{R}_3: X \to Y$ by $\mathscr{R}_3=\{(a, w), (b, w), (c, x)\}$, then $\mathscr{R}_3$ is a function. $\mathscr{D}(\mathscr{R}_3)=X$, $\mathscr{R}(\mathscr{R}_3)=\{x, w\} \neq Y$. The relation $\mathscr{R}_3^{-1}: Y \rightsquigarrow X$ is not a partial function.

Suppose that $\mathscr{R}: X \rightsquigarrow Y$ is a relation. We say that $\mathscr{R}$ is *surjective* if $\mathscr{R}(\mathscr{R})=Y$, and $\mathscr{R}$ is called *injective* if, given

$$(x, y) \in \mathscr{R} \text{ and } (x', y) \in \mathscr{R} \text{ then } x=x'.$$

Theorem 1.1.2

Let $\mathscr{R}: X \rightsquigarrow Y$ be a relation.

(i) If $\mathscr{R}$ is injective then $\mathscr{R}^{-1}$ is a partial function.

(ii) If $\mathscr{R}$ is injective and surjective then $\mathscr{R}^{-1}$ is a function.

Proof (i) We must show that $\mathscr{R}^{-1}: Y \rightsquigarrow X$ is a partial function, so that if

$$(y, x) \in \mathscr{R}^{-1} \text{ and } (y, x') \in \mathscr{R}^{-1} \text{ then } x=x'$$

but $(y, x) \in \mathscr{R}^{-1}$ is equivalent to $(x, y) \in \mathscr{R}$ and similarly $(y, x') \in \mathscr{R}^{-1}$ yields $(x', y) \in \mathscr{R}$. The injectivity of $\mathscr{R}$ gives us $x=x'$ and so $\mathscr{R}^{-1}$ is a partial function.

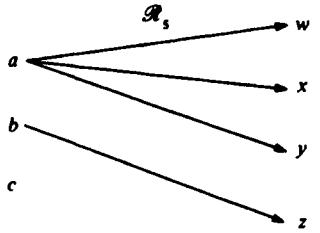
(ii) Since $\mathscr{D}(\mathscr{R}^{-1})=\mathscr{R}(\mathscr{R})=Y$ we see that $\mathscr{R}^{-1}: Y \to X$ is now a function. □

Examples 1.2

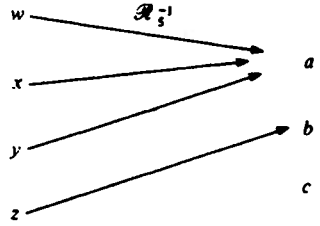
Let $X=\{a, b, c\}$, $Y=\{w, x, y, z\}$. Define $\mathscr{R}_4: X \rightsquigarrow Y$ by $\mathscr{R}_4=\{(a, x), (a, y), (b, z)\}$ then $\mathscr{R}_4$ is injective and $\mathscr{R}_4^{-1}=\{(x, a), (y, a), (z, b)\}$ is a partial function $\mathscr{R}_4^{-1}: Y \to X$.

If $\mathscr{R}_5: X \rightsquigarrow Y$ is given by $\mathscr{R}_5=\{(a, x), (a, y), (b, z), (a, w)\}$ then $\mathscr{R}_5^{-1}: Y \to X$ becomes $\mathscr{R}_5^{-1}=\{(w, a), (x, a), (y, a), (z, b)\}$ which is a function.

In many cases we will indicate the definition of a relation by using a diagram, for example the relation $\mathscr{R}_5$ is represented by the arrows in the following:



and $\mathscr{R}_5^{-1}$ is given by:



Theorem 1.1.3

Let $\mathscr{R}: X \to Y$ be a function, then $\mathscr{R}^{-1}: Y \to X$ is a function if and only if $\mathscr{R}$ is surjective and injective.

Proof If $\mathscr{R}$ is an injective, surjective function then $\mathscr{R}^{-1}$ is a function by 1.1.2. If $\mathscr{R}^{-1}$ is a function and $\mathscr{R}$ is a function then $\mathscr{D}(\mathscr{R}^{-1})=Y=\mathscr{R}(\mathscr{R})$ and so $\mathscr{R}$ is surjective. Suppose that $(x, y) \in \mathscr{R}$ and $(x', y) \in \mathscr{R}$ for some $x, x' \in X$, $y \in Y$, then $(y, x) \in \mathscr{R}^{-1}$ and $(y, x') \in \mathscr{R}^{-1}$. But $\mathscr{R}^{-1}$ is a function and so $x=x'$, which means that $\mathscr{R}$ is injective. □

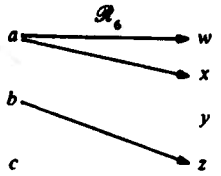
There is a natural concept of inclusion that can be defined between relations. If $\mathscr{R}: X \rightsquigarrow Y$ and $\mathscr{S}: X \rightsquigarrow Y$ are relations then $\mathscr{R} \subseteq X \times Y$ and $\mathscr{S} \subseteq X \times Y$. Suppose that $\mathscr{R} \subseteq \mathscr{S}$ then we see that

$$\text{given } (x, y) \in \mathscr{R} \text{ then } (x, y) \in \mathscr{S}.$$

This inclusion of relations may also be applied to partial functions in the natural way.

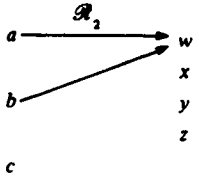
Example 1.3

The relation $\mathscr{R}_6 : X \rightsquigarrow Y$ given by

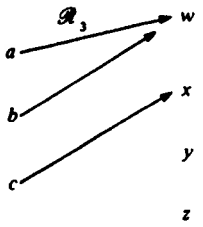


is such that $\mathscr{R}_6 \subseteq \mathscr{R}_5$ where $\mathscr{R}_5$ is defined in example 1.2.

The partial function $\mathscr{R}_2 : X \to Y$ defined in example 1.1 is



and $\mathscr{R}_3 : X \to Y$ is the function



and so $\mathscr{R}_2 \subseteq \mathscr{R}_3$.

Finally we examine the problem of defining functions between empty sets. We have already noted that $\varnothing \times Y$ and $X \times \varnothing$ both equal the empty set $\varnothing$, consequently there are relations $\varnothing : \varnothing \rightsquigarrow Y$ and $\varnothing : X \rightsquigarrow \varnothing$, where X and Y are sets. Both of these relations are in fact partial functions since the condition for a partial function is satisfied vacuously. However $\varnothing : \varnothing \to Y$ is a function whereas $\varnothing : X \to \varnothing$ is not if $X \neq \varnothing$, since $\mathfrak{D}(\varnothing) = \varnothing \neq X$ in this case.

(Notice that a relation $\mathscr{R} : X \rightsquigarrow Y$ will *not* be a partial function if we can find $x \in X$, $y, y' \in Y$ such that $(x, y) \in \mathscr{R}$ and $(x, y') \in \mathscr{R}$ and $y \neq y'$. In neither of these last two cases can this be done and so both $\varnothing : \varnothing \rightsquigarrow Y$ and $\varnothing : X \to \varnothing$ are partial functions. The relations $\varnothing : \varnothing \to \varnothing$, $\theta : X \to Y$ are in fact partial functions.)

The relation notation used here is sometimes a little cumbersome in practice and we propose to adjust it slightly especially when we are dealing with functions and partial functions.

Let $\mathscr{R} : X \to Y$ be a partial function and suppose that $x \in X$, then either there exists a $y \in Y$ such that $(x, y) \in \mathscr{R}$ or no such y exists. In the first case we will write $y = \mathscr{R}(x)$ and in the second case $\varnothing = \mathscr{R}(x)$. By a natural extension of this we will use the notation $\mathscr{R}(X)$ for $\mathfrak{R}(\mathscr{R})$, the range of $\mathscr{R}$. Generally if $X' \subseteq X$ then we write

$$\mathscr{R}(X') = \{y \in Y \mid (x', y) \in \mathscr{R} \text{ for some } x' \in X'\}.$$

Now let X_i $(i \in I)$ be a family of subsets of X. Then

$$\begin{aligned} \mathscr{R}\left(\bigcup_{i \in I} X_i\right) &= \left\{y \in Y \mid (x', y) \in \mathscr{R} \text{ for some } x' \in \bigcup_{i \in I} X_i\right\} \\ &= \bigcup_{i \in I} \{y \in Y \mid (x_i, y) \in \mathscr{R} \text{ for some } x_i \in X_i\} \\ &= \bigcup_{i \in I} \mathscr{R}(X_i). \end{aligned}$$

We describe this situation by saying that the relation $\mathscr{R}$ is *completely additive*.

If $\mathscr{R} : X \rightsquigarrow Y$ is a relation and $x \in X$ then we define

$$\mathscr{R}(x) = \{y \in Y \mid (x, y) \in \mathscr{R}\}.$$

In some circumstances $\mathscr{R}(x)$ is a singleton (for example when $\mathscr{R}$ is a function) and it is convenient to identify this singleton subset with the element it contains and in this way we may establish a coherent notation. So that if $\mathscr{R} : X \rightsquigarrow Y$ and $\mathscr{S} : X \rightsquigarrow Y$ are relations and $x, x' \in X$ then the phrase $\mathscr{R}(x) \subseteq \mathscr{S}(x')$ will be meaningful whether $\mathscr{R}$ and $\mathscr{S}$ are functions or not.

From example 1.3 we have $\mathscr{R}_2(a) \subseteq \mathscr{R}_3(a)$ meaning $\mathscr{R}_2(a) = \mathscr{R}_3(a)$ and $\mathscr{R}_2(c) \subseteq \mathscr{R}_3(c)$ which means $\varnothing \subseteq \{x\}$.

If $\mathscr{R} : X \rightsquigarrow Y$ and $\mathscr{S} : Y \rightsquigarrow Z$ are relations then we define the *composition* or *product* relation $\mathscr{S} \circ \mathscr{R} : X \rightsquigarrow Z$ by $(x, z) \in \mathscr{S} \circ \mathscr{R}$ if and only if there exists $y \in Y$ such that $(x, y) \in \mathscr{R}$ and $(y, z) \in \mathscr{S}$. Clearly if $\mathfrak{R}(\mathscr{R}) \cap \mathfrak{D}(\mathscr{S}) = \varnothing$ then $\mathscr{S} \circ \mathscr{R} = \varnothing$. When $\mathscr{R} : X \to Y$ and $\mathscr{S} : Y \to Z$ are functions then $\mathscr{S} \circ \mathscr{R} : X \to Z$ is also a function (except in the case when $\mathscr{S} = \varnothing$ and $X \neq \varnothing$).

In all cases $(\mathscr{S} \circ \mathscr{R})(x) = \mathscr{S}(\mathscr{R}(x))$ where $\mathscr{S}(\mathscr{R}(x))$ is defined to be $\cup\{\mathscr{S}(y) \mid y \in \mathscr{R}(x)\}$.

Since relations are defined as subsets it is possible to consider the intersection of two relations. So that if $\mathscr{R} : X \rightsquigarrow Y$ and $\mathscr{S} : X \rightsquigarrow Y$ are relations then $\mathscr{R} \subseteq X \times Y$ and $\mathscr{S} \subseteq X \times Y$. The *intersection relation*

$\mathcal{R} \cap \mathcal{S} : X \rightsquigarrow Y$ is then defined to be the subset $\mathcal{R} \cap \mathcal{S} \subseteq X \times Y$. One particular example of interest is the case where $X = Y$ and $\mathcal{R}$ and $\mathcal{S}$ are equivalence relations. If $a \in X$ then $[a]_{\mathcal{R}\cap\mathcal{S}} = [a]_{\mathcal{R}} \cap [a]_{\mathcal{S}}$, where $[a]_{\mathcal{R}\cap\mathcal{S}}$ means the equivalence class of a with respect to the equivalence relation $\mathcal{R} \cap \mathcal{S}$.

1.2 Semigroups and homomorphisms

On many occasions we will be dealing with a set S which has some additional structure. This will often take the form of a rule for 'combining' certain elements of S to 'produce' a *unique* element of S. We shall refer to such processes as multiplications on S.

If s and s_1 are elements of S and they can be combined to form a new element of S we would write this as $s \cdot s_1 = s_2$ where $s_2 \in S$. This process is somewhat more precisely stated if we use the concept of a relation. Then we are considering a relation $\mathcal{R} : S \times S \rightsquigarrow S$ defined by $((s, s_1), s_2) \in \mathcal{R}$ if and only if $s \cdot s_1 = s_2$. In fact $\mathcal{R}$ is a partial function since we want s_2 to be a *unique* element. The domain $\mathcal{D}(\mathcal{R})$ may be a proper subset of $S \times S$ in which case we cannot multiply an arbitrary pair of elements of S. $\mathcal{R}$ is said to define a *closed partial binary operation* on S. If $\mathcal{D}(\mathcal{R}) = S \times S$ then $\mathcal{R}$ is a *closed binary operation* on S, in this case we can multiply any two elements of S to obtain an element of S. We will usually drop the multiplication symbol when dealing with products of elements in S, writing ss_1 in place of $s \cdot s_1$.

If S has a closed binary operation satisfying the *associativity* condition $a(bc) = (ab)c$ for every $a, b, c \in S$ we call S a *semigroup*. If we need to specify the operation involved we will write '$(S, \cdot)$ is a semigroup'. Semigroups are very common in many branches of mathematics and they certainly play a central role in the theory of automata so we had better look at some of their more important properties. First of all we will examine some examples.

Examples 1.4

(i) The set of natural numbers $\{1, 2, \ldots\}$ is a semigroup with respect to both the binary operations of addition and multiplication.

(ii) The set of all integers is a semigroup with respect to the binary operations of addition and multiplication.

(iii) Let A be a set and S the set of all relations that can be defined on A. Suppose that $\mathcal{R} : A \rightsquigarrow A$ and $\mathcal{R}' : A \rightsquigarrow A$ are both members of S, as before we define a new relation $\mathcal{R} \circ \mathcal{R}' : A \rightsquigarrow A$ by

$$(a, b) \in \mathcal{R} \circ \mathcal{R}' \Leftrightarrow \exists c \in A \text{ such that } (a, c) \in \mathcal{R}' \text{ and } (c, b) \in \mathcal{R}.$$

This process defines a closed binary operation on the set S which is associative and so S is a semigroup.

(iv) If A is a set and S is the set of all functions from A to A then we can define a closed binary operation on S in the same way as in (iii). So if $\mathcal{R} : A \to A$ and $\mathcal{R}' : A \to A$ are functions we define a new function $\mathcal{R} \circ \mathcal{R}' : A \to A$ by:

$$\mathcal{R} \circ \mathcal{R}'(a) = b \Leftrightarrow \exists c \in A \text{ such that } \mathcal{R}'(a) = c \text{ and } \mathcal{R}(c) = b.$$

$$\text{i.e. } \mathcal{R} \circ \mathcal{R}'(a) = b \Leftrightarrow \mathcal{R}(\mathcal{R}'(a)) = b.$$

Therefore the binary operation is just function composition.

The set of all partial functions from A to A is a semigroup under this operation, the partial function $\theta : A \to A$ belongs to the set; we denote it by $\boldsymbol{PF}(A)$.

(v) Let Σ be any non-empty set. A *string* or *word* from Σ is any finite sequence of elements from Σ. Define a closed binary operation on the set Σ^+ of all words from Σ as follows. Let $\sigma_1 \ldots \sigma_n, \sigma'_1 \ldots \sigma'_m \in \Sigma^+$, then $\sigma_1 \ldots \sigma_n \cdot \sigma'_1 \ldots \sigma'_m$ is called their *concatenation* and is clearly a word from Σ. It is easy to see that Σ^+ is a semigroup with respect to this operation. We shall call Σ^+ the *free semigroup generated by the set* Σ. We regard Σ as being embedded in Σ^+.

(vi) The empty set is a semigroup, the binary operation is the empty function $\varnothing : \varnothing \times \varnothing \to \varnothing$ which is associative – a set S with a closed binary operation is a semigroup *unless* we can find elements $a, b, c \in S$ such that

$$a(bc) \neq (ab)c.$$

If $(S, \cdot)$ is a semigroup and $A \subseteq S$, $B \subseteq S$ are subsets we define

$$AB = \{s \in S \mid s = ab \text{ for some } a \in A, b \in B\}.$$

Closely associated with the idea of a semigroup is the concept of a monoid. For this we need another definition.

Let S be a semigroup, an element $e \in S$ is called a *unit element* of S if $ae = ea = a$ for every $a \in S$. A simple exercise shows that if S possesses a unit element then that element is unique.

A semigroup possessing a unit element is called a *monoid*.

Examples 1.5

(i) Both semigroups in example 1.4(i) are monoids, in fact all the semigroups in that example with the exception of 1.4(v) and 1.4(vi) are monoids.

(ii) Let Σ be a non-empty set. We have already examined the set Σ^+ consisting of all words from Σ. Let us adjoin an extra element, called

the *null word* and denoted by Λ, this is just the empty sequence and has the following formal properties:
if

$$a \in \Sigma^+$$

then

$$\Lambda a = a\Lambda = a$$

and so Λ acts like a unit. If we define $\Sigma^* = \Sigma^+ \cup \{\Lambda\}$ then Σ^* is a monoid. It is called the *free monoid generated by the set* Σ.

This procedure gives us a general method for transforming a semigroup into a monoid.

Let S be a semigroup. Suppose that S is not a monoid, choose any element $e \notin S$ and form the set $S^\cdot = S \cup \{e\}$, we define a multiplication denoted by $*$ on $S^\cdot$ by extending the multiplication already on S so that

$$a * b = \begin{cases} ab & \text{if } a, b \in S \\ b & \text{if } a = e \\ a & \text{if } b = e \end{cases}$$

where $a, b \in S^\cdot$.

Clearly $S^\cdot$ is a monoid with unit element e. If S is already a monoid we will define $S^\cdot = S$.

Let $(S, \cdot)$ and $(T, *)$ be semigroups and $f: S \to T$ be a mapping. We call f a *semigroup homomorphism* if

$$f(a) * f(b) = f(ab) \quad \text{for all } a, b \in S.$$

If $(S, \cdot)$ and $(T, *)$ are monoids with identities e and e' respectively and $f: S \to T$ is a semigroup homomorphism such that

$$f(e) = e'$$

then f is called a *monoid homomorphism.*

Semigroup homomorphisms forge a strong link between the semigroup structures concerned and are of great importance in the development of the theory. Sometimes, however, it is necessary to consider slightly more general relationships between semigroups. If $\mathcal{R}: S \rightsquigarrow T$ is a relation then it is called a *semigroup relation* if

$$\mathcal{R}(a) * \mathcal{R}(b) \subseteq \mathcal{R}(ab) \quad \text{for } a, b \in S.$$

What does this mean? Well suppose that $c \in T$ with $(a, c) \in \mathcal{R}$ and $d \in T$ with $(b, d) \in \mathcal{R}$, then

$$c * d \in \mathcal{R}(a) * \mathcal{R}(b) \subseteq \mathcal{R}(ab)$$

which means that $(ab, c * d) \in \mathcal{R}$.

Example 1.6

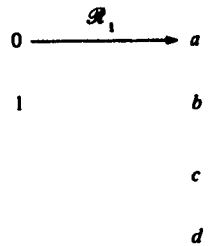
Let $S = \{0, 1\}$ and $T = \{a, b, c, d\}$ with multiplications defined by the following tables:

S	0	1
0	0	1
1	1	1

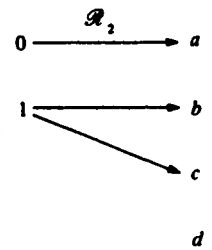
T	a	b	c	d
a	a	b	c	d
b	b	c	a	d
c	c	a	b	d
d	d	d	d	d

so that, for example $a * c = c$, where c is the common entry in the row labelled a and the column labelled c, etc.

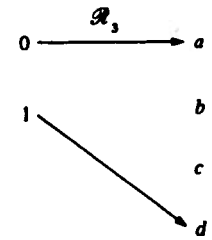
We will define various relations and functions between these semigroups:



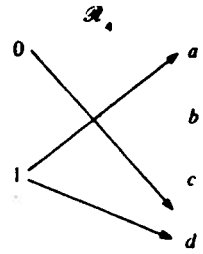
This is a semigroup relation (in fact a partial function) since
$\mathcal{R}_1(0) * \mathcal{R}_1(0) = \{a * a\} = \{a\} \subseteq \mathcal{R}_1(0)$.



This is not a semigroup relation since
$\mathcal{R}_2(1) * \mathcal{R}_2(1) = \{b, c\} * \{b, c\} =$
$\{a, b, c\} \not\subseteq \mathcal{R}_2(1) = \{b, c\}$,
although $\mathcal{R}_2(1) * \mathcal{R}_2(0) =$
$\{b, c\} * \{a\} = \{b, c\} = \mathcal{R}_2(1)$ etc.



This is a semigroup homomorphism.



This is a semigroup relation,
$(\mathcal{R}_4(0) * \mathcal{R}_4(0) = \{d\} \subseteq \mathcal{R}_4(0)$
$\mathcal{R}_4(1) * \mathcal{R}_4(0) = \{d\} \subset \mathcal{R}_4(1)$ etc.).

Let S be a semigroup and S' a subset of S such that if $a, b \in S'$ then $ab \in S'$, then we say that S' is a *subsemigroup* of S. Usually we denote this by $S' \subseteq S$ and rely on the context to indicate that S' is a subsemigroup of S rather than just a subset. If X is a subset of S define $\langle X \rangle$ to be the intersection of all subsemigroups of S that contain X. Then $\langle X \rangle$ is the *subsemigroup generated by* X.

In the case where S is a monoid with unit element $e \in S$ then a subsemigroup $S' \subseteq S$ is a *submonoid* of S if e also belongs to S'.

The proof of the following elementary fact is left to the reader.

Lemma 1.2.1

Let S and T be semigroups and $f: S \to T$ a partial semigroup homomorphism. Then $f(S)$ is a subsemigroup of T.

If S and T are monoids then $f(S)$ may not be a submonoid of T unless $f(e)$ is the unit element of T, where e is the unit of S. Semigroup homomorphisms possessing this unit-preserving property are the monoid homomorphisms.

We next examine the structure induced on a semigroup by the existence of a semigroup homomorphism.

Let $f: S \to T$ be a semigroup homomorphism, define a relation $\sim$ on S by

$$a \sim b \text{ if and only if } f(a) = f(b), \qquad \text{for } a, b \in S.$$

It is easily verified that $\sim$ is an equivalence relation on the set S. It also satisfies the property that if $a, b \in S$, $a \sim b$ and $s \in S$ then $as \sim bs$ and $sa \sim sb$. This is because $f(as) = f(a) * f(s) = f(b) * f(s) = f(bs)$ etc. Such a relation is called the *congruence relation on S defined by f*.

Given any semigroup S a *congruence relation* on S is an equivalence relation $\sim$ satisfying:

$$a \sim b \Rightarrow as \sim bs \text{ and } sa \sim sb \qquad \text{for all } s \in S.$$

We shall shortly see that every congruence relation is defined by some suitable semigroup homomorphism.

First let S be a semigroup and $\sim$ a congruence relation on S. Consider the set of all equivalence classes defined on S by $\sim$, denote this set by $S/\sim$. We define a multiplication on $S/\sim$ as follows:

$$\text{let } [a], [b] \in S/\sim$$
$$\text{put } [a] * [b] = [ab] \quad (a, b \in S).$$

This operation is well defined for if

$$[a] = [c] \text{ and } [b] = [d] \text{ then}$$
$$a \sim c \text{ and } b \sim d, \text{ consequently}$$
$$ab \sim cb \text{ and } cb \sim cd \text{ and so } ab \sim cd$$
$$\text{that is } [ab] = [cd] \text{ and thus } [a] * [b] = [c] * [d].$$

Furthermore $S/\sim$ under the operation $*$ is a semigroup. We call $(S/\sim, *)$ the *quotient semigroup of S with respect to* $\sim$. There is a semigroup homomorphism $f: S \to S/\sim$ defined by $f(a) = [a]$, $a \in S$. This is called the *natural homomorphism onto* $S/\sim$.

Notice further that the congruence relation on S defined by f is just the relation $\sim$ that we started with. There is thus a precise correspondence between semigroup homomorphisms and congruences.

We will finish this section with two important but elementary results that are of fundamental importance.

Theorem 1.2.2

Let $f: S \to T$ be a surjective semigroup homomorphism and $\sim$ the congruence induced on S by f. There exists a bijective homomorphism $f^*: S/\sim \to T$ such that $f^*([a]) = f(a)$ for each $a \in S$.

Proof The definition of f^* specified in the hypothesis is well-defined and using it we will just establish that f^* is bijective and a semigroup homomorphism. First let $f^*([a]) = f^*([b])$ where $[a], [b] \in S/\sim$ then $f(a) = f(b)$ and so $a \sim b$ which means $[a] = [b]$. Thus f^* is injective. Next let $t \in T$, then there exists $a \in S$ such that $t = f(a)$ and consequently $f^*([a]) = f(a) = t$ giving the surjectivity of f^*.

Finally let $[a], [b] \in S/\sim$, then $[a] * [b] = [ab]$ and

$$f^*([a] * [b]) = f^*([ab]) = f(ab) = f(a)f(b) = f^*([a])f^*([b])$$

proving that f^* is a semigroup homomorphism. □

We usually call a bijective semigroup homomorphism $f: S \to T$ an *isomorphism* and write $S \approx T$ to indicate that an isomorphism exists between S and T.

Theorem 1.2.3

Let Σ be a non-empty set, T a semigroup and $f: \Sigma \to T$ a mapping. There is a mapping $g: \Sigma^+ \to T$ which is a semigroup homomorphism and such that $g(\sigma) = f(\sigma)$ for all $\sigma \in \Sigma$.

Proof Let $a \in \Sigma^+$ then $a = \sigma_1 \sigma_2 \ldots \sigma_n$ for some $\sigma_i \in \Sigma$; $i = 1, 2, \ldots, n$. Define $g(a) = f(\sigma_1) f(\sigma_2) \ldots f(\sigma_n)$ (using the multiplication in T) and so $g: \Sigma^+ \to T$ is defined. For $a, b \in \Sigma^+$ it is immediate that

$$g(ab) = g(a)g(b)$$

and so g is a semigroup homomorphism. □

In fact g is the unique homomorphism satisfying $g(\sigma) = f(\sigma)$ for $\sigma \in \Sigma$. This property of the semigroup Σ^+ is what gives it the name 'free semigroup'.

Now let S be a semigroup, a semigroup homomorphism $f: S \to S$ will be called an *endomorphism*. For any semigroup S the set of all endomorphisms will be denoted by $\mathbf{End}(S)$. The set $\mathbf{End}(S)$ has a natural semigroup structure with respect to 'composition of mappings'. Since the identity mapping $1_S: S \to S$ defined by $1_S(a) = a$ for all $a \in S$ is clearly an endomorphism we see that $\mathbf{End}(S)$ is actually a monoid.

If S is a semigroup and X is a non-empty subset of S we define the subsemigroup of S generated by X to be the intersection of all the subsemigroups of S that contain the subset X and denote this by $\langle X \rangle$.

Let n be a positive integer and write

$$\mathbf{n} = \{0, 1, \ldots, n-1\}.$$

Consider the set S of all functions of $\mathbf{n}$ into itself. This is a semigroup under the operation of function composition. The semigroup S has order n^n and is in fact a monoid with identity the identity map $1_{\mathbf{n}}$.

Let $s \in S$ be defined by:

$$s(x) = x + 1 \quad \text{for } 0 \leq x < n-1$$

$$s(n-1) = 0.$$

The subsemigroup generated by the set $\{s\}$ is the set

$$\langle \{s\} \rangle = \{1_{\mathbf{n}}, s, s^2, \ldots, s^{n-1}\}$$

which is a group satisfying $s^n = 1_{\mathbf{n}}$.

Another subsemigroup of interest is the subset

$$R = \{t \in S \mid |t(\mathbf{n})| = 1\}$$
$$= \{\bar{k} \mid k \in \mathbf{n}\}$$

where $\bar{k}$ is the function defined by

$$\bar{k}(x) = k$$

for all $x \in \mathbf{n}$.

1.3 Products

Semigroups can be joined together in various ways to produce more semigroups. We will examine here some important methods of doing this.

Let S and T be semigroups. Consider the set $S \times T$, the cartesian product of S and T, and define a multiplication on $S \times T$ as follows:

$$(a, x) \cdot (b, y) = (ab, xy)$$

where $a, b \in S$; $x, y \in T$.

The result is a semigroup $(S \times T, \cdot)$ which is called the *direct product* of S and T, written $S \times T$. Associated with the direct product $S \times T$ are two important functions:

$$p_1: S \times T \to S \text{ defined by } p_1(a, x) = a \quad (a \in S, x \in T)$$

and

$$p_2: S \times T \to T \text{ defined by } p_2(a, x) = x \quad (a \in S, x \in T).$$

These are called the *projections* onto the first and second factors respectively and are easily seen to be surjective semigroup homomorphisms. They satisfy an important property:

Theorem 1.3.1

Let S, T and W be semigroups and $f_1: W \to S$, $f_2: W \to T$ semigroup homomorphisms. There exists a unique semigroup homomorphism $g: W \to S \times T$ such that $p_i \circ g = f_i$ for $i = 1, 2$.

Proof Define $g(w) = (f_1(w), f_2(w)) \in S \times T$ for each $w \in W$. Then for $w, w' \in W$

$$\begin{aligned} g(ww') &= (f_1(ww'), f_2(ww')) \\ &= (f_1(w)f_1(w'), f_2(w)f_2(w')) \\ &= (f_1(w), f_2(w)) \cdot (f_1(w'), f_2(w')) \\ &= g(w) \cdot g(w') \end{aligned}$$

and so g is a semigroup homomorphism.

Clearly

$$p_1(g(w)) = p_1(f_1(w), f_2(w)) = f_1(w)$$

and

$$p_2(g(w)) = p_2(f_1(w), f_2(w)) = f_2(w).$$

Finally let $h: W \to S \times T$ also be a semigroup homomorphism with the property that $p_i \circ h = f_i$, $i = 1, 2$. Now let $h(w) = (a, x)$ where $w \in W$, $a \in S$, $x \in T$. Then $p_1(h(w)) = a = f_1(w)$, $p_2(h(w)) = x = f_2(w)$ and so $h(w) = (f_1(w), f_2(w)) = g(w)$. □

Given three semigroups S, T, W we can form $S \times T$ and then $(S \times T) \times W$; similarly $S \times (T \times W)$ can be constructed and it is natural to ask about the relationship between these two semigroups.

Lemma 1.3.2

$(S \times T) \times W \approx S \times (T \times W)$.

Proof The isomorphism is $f: (S \times T) \times W \to S \times (T \times W)$ defined by

$$f((a, b), c) = (a, (b, c))$$

where $a \in S$, $b \in T$, $c \in W$. □

Now let S and T be semigroups and suppose that $\theta: T \to \mathbf{End}\ S$ is a semigroup homomorphism. We will define another product on the set $S \times T$, let $t, t' \in T$; s, $s' \in S$ put $(s, t) \circledast (s', t') = (s\theta(t)(s'), tt')$, where $\theta(t): S \to S$ and so $\theta(t)(s') \in S$. Clearly $S \times T$ is a semigroup with respect to this product, since

$$\begin{aligned}
&((s, t) \circledast (s', t')) \circledast (s'', t'') \\
&\quad = (s\theta(t)(s'), tt') \circledast (s'', t'') \\
&\quad = (s\theta(t)(s')\theta(tt')(s''), tt't'') \\
&\quad = (s\theta(t)(s')\theta(t)(\theta(t')(s'')), tt't''), \quad \text{as } \theta \text{ is a homomorphism,} \\
&\quad = (s\theta(t)(s'\theta(t')(s'')), tt't''), \quad \text{as } \theta(t) \in \mathbf{End}\ S, \\
&\quad = (s, t) \circledast (s'\theta(t')(s''), t't'') \\
&\quad = (s, t) \circledast ((s', t') \circledast (s'', t''))
\end{aligned}$$

This semigroup is called the *semidirect product* of S and T with respect to θ and denoted by $S \times_\theta T$.

Our third product is constructed in the following way, let S^{T^*} denote the set of all functions from the monoid T^* to the semigroup S. We define a multiplication $\circ$ on the set $S^{T^*} \times T$ by putting

$$(f, t) \circ (g, t') = (f \circ g, tt')$$

where $f \circ g \in S^{T^*}$ is defined by $(f \circ g)(x) = f(x)g(xt)$ for $x \in T^*$, $f, g \in S^{T^*}$, $t, t' \in T$. If $f, g, h \in S^{T^*}$ and $t, t', t'' \in T$ then

$$\begin{aligned}
((f, t) \circ (g, t')) \circ (h, t'') &= (f \circ g, tt') \circ (h, t'') \\
&= ((f \circ g) \circ h, tt't'')
\end{aligned}$$

and

$$(f, t) \circ ((g, t') \circ (h, t'')) = (f \circ (g \circ h), tt't'').$$

Now if $x \in T^*$ then

$$\begin{aligned}
((f \circ g) \circ h)(x) &= (f \circ g)(x)h(xtt') \\
&= f(x)g(xt)h(xtt')
\end{aligned}$$

and

$$\begin{aligned}
(f \circ (g \circ h))(x) &= f(x)(g \circ h)(xt) \\
&= f(x)g(xt)h(xtt')
\end{aligned}$$

and thus we have established the associativity of the multiplication on the set $S^{T^*} \times T$. We call the semigroup $S^{T^*} \times T$ the *wreath product of* S *and* T and write it as $S \circ T$.

Now let S and T be semigroups and consider the set S^{T^*} of all mappings from T^* into S. Suppose that $f: T^* \to S$ and $t \in T$ then we may define a mapping $f_t: T^* \to S$ by $f_t(x) = f(xt)$ where $x \in T^*$. The set S^{T^*} is a semigroup under the multiplication induced by the semigroup S, for example if $f, g \in S^{T^*}$ then

$$(fg)(x) = f(x)g(x) \in S \quad \text{where } x \in T^* \tag{*}$$

Furthermore the mapping $\theta_t: S^{T^*} \to S^{T^*}$ defined by

$$\theta_t(f) = f_t, \quad f \in S^{T^*}$$

is an endomorphism, for if $f, g \in S^{T^*}$ then $\theta_t(fg) = (fg)_t$ where $(fg)_t(x) = fg(xt) = f(xt)g(xt)$ by (*) and $\theta_t(f) \cdot \theta_t(g) = f_t g_t$ where $(f_t g_t)(x) = f_t(x)g_t(x) = f(xt)g(xt)$ and thus

$$\theta_t(fg) = \theta_t(f)\theta_t(g).$$

Consequently there exists a mapping $\theta: T \to \mathbf{End}(S^{T^*})$ defined by $\theta(t) = \theta_t$ for $t \in T$. It is now possible to define the semidirect product $S^{T^*} \times_\theta T$ and we note that if $(f, t), (g, t') \in S^{T^*} \times T$ then their semidirect multiplication is given by

$$(f, t) \circledast (g, t') = (f\theta_t(g), tt')$$

where

$$(f\theta_t(g))(x) = f(x)g(xt) \quad \text{for all } x \in T^{\cdot}$$
$$= (f \circ g)(x) \quad \text{for all } x \in T^{\cdot}$$

and so

$$(f, t) \circledast (g, t') = (f, t) \circ (g, t').$$

We thus have:

Theorem 1.3.3

If S and T are semigroups then there exists $\theta: T \to \mathbf{End}(S^{T^{\cdot}})$ such that

$$S \circ T = S^{T^{\cdot}} \times_{\theta} T.$$

Before we examine the last method for combining two semigroups we must introduce the idea of a zero.

If S is a semigroup an element $s \in S$ is called a *zero* of S if

$$sa = as = s \quad \text{for all } a \in S.$$

The empty partial function $\theta: A \to A$ is a zero for $\mathbf{PF}(A)$.

A zero element, if it exists, is necessarily unique. Not all semigroups possess a zero but one can easily be adjoined, so that if S is an arbitrary semigroup without a zero element we define $S^0 = S \cup \{0\}$ where $0 \notin S$ and define

$$a * b = \begin{cases} ab & \text{if } a, b \in S \\ 0 & \text{otherwise,} \end{cases}$$

then S^0 is a semigroup under $*$ and 0 is the zero element of S^0.

Now let S and T be semigroups and suppose that $S \neq \varnothing$ and $T \neq \varnothing$. Consider their disjoint union $S \cup T$: this is not in general a semigroup but if we adjoint a zero element in a suitable way we can construct a semigroup multiplication.

Let $0 \notin S \cup T$ and put

$$S \vee T = S \cup T \cup \{0\}$$

and define

$$a * b = \begin{cases} ab & \text{if } a, b \in S \\ ab & \text{if } a, b \in T \\ 0 & \text{otherwise.} \end{cases}$$

Then $S \vee T$ is a semigroup under $*$ with a zero. We call it the *join of S and T.*

Occasionally we need to consider the case when either S or T is the empty semigroup $\varnothing$ in which case we define

$$S \vee \varnothing = \varnothing \vee S = S.$$

1.4 Groups

Let G be a monoid with identity e. Suppose that for each $g \in G$ there exists an element $\hat{g} \in G$ such that $g\hat{g} = e = \hat{g}g$. We say that G is a *group* and usually write the element $\hat{g}$ as g^{-1} and call it the *inverse* of g.

Groups are an important mathematical concept, they arise in many situations and we now briefly state some of their important properties that we will need later.

A group G is *abelian* if $gg_1 = g_1g$ for any $g, g_1 \in G$.

The *cyclic group of order n* is the set

$$\mathbf{n} = \{0, 1, \ldots, n-1\}$$

with multiplication defined by

$$xy = r$$

where $x + y = qn + r$ and $0 \leq r \leq n-1$. This group is abelian and is usually written as $\mathbf{Z}_n$.

A *subgroup* of a group G is a submonoid $H \subseteq G$ such that $hh_1^{-1} \in H$ for each $h, h_1 \in H$. A subgroup must contain $\{e\}$. The identity singleton $\{e\}$ and the group G are both subgroups of G. If they are the only subgroups then $G \approx \mathbf{Z}_p$ for some prime number p. (Groups are said to be isomorphic if they are isomorphic as monoids.)

Let H be a subgroup of G, we say that H is *normal in G*, written $H \lhd G$, if $g^{-1}hg \in H$ for all $h \in H$, $g \in G$. This is equivalent to saying that a semigroup homomorphism $f: G \to G'$ exists where G and G' are groups, $H = f^{-1}(\{e'\})$ and e' is the identity of G'.

Given any finite group G and a subgroup H we form the *right cosets* of H in G. These are all subsets of the form $Hg = \{hg \mid h \in H\}$. It is easily verified that the set G/H of *distinct* right cosets forms a partition of the set G. The equivalence relation defined by this partition is given by

$$g \sim g_1 \Leftrightarrow g_1 = hg$$

for some $h \in H$ where $g, g_1 \in G$.

If H is normal in G we can define a multiplication on the set, G/H, of all distinct right cosets of H in G by

$$Hg \cdot Hg_1 = Hgg_1, \quad Hg, Hg_1 \in G/H.$$

This turns G/H into a group with identity $He = H$.

The function $f_H : G \to G/H$ defined by

$$f_H(g) = Hg \quad \text{for } g \in G$$

is a homomorphism onto G/H and $H = f_H^{-1}(\{H\})$. We call f_H the *natural* (or canonical) homomorphism onto G/H.

Theorem 1.4.1

Let $f: G \to G_1$ be a homomorphism of the group G onto G_1. If $H = f^{-1}(\{e_1\})$ then $G_1 \approx G/H$.

Proof Construct a function $\phi: G/H \to G_1$ by

$$\phi(Hg) = f(g) \quad \text{for } Hg \in G/H.$$

This is well-defined for if $Hg = Hg'$ then $g' = hg$ for some $h \in H$ and $\phi(Hg') = f(g') = f(hg) = f(h)f(g) = e_1 f(g) = f(g) = \phi(Hg)$.

Furthermore it is easy to establish that ϕ is an isomorphism. □

Theorem 1.4.2

Let $H \lhd G$. There is a one-one correspondence between the subgroups of G/H and subgroups of G that contain H.

Proof Let K be a subgroup of G/H, so that K is a collection of cosets of the form Hg, $(g \in G)$. Recall that $f_H : G \to G/H$ is an onto homomorphism. Let $L = f_H^{-1}(K)$, then $L \subseteq G$. Since He, the identity of G/H, belongs to K we see that $H = f_H^{-1}(He) \subseteq L$. If $l, l_1 \in L$ then

$$f_H(ll_1^{-1}) = Hll_1^{-1} = Hl(Hl_1)^{-1} \in K$$

so L is a subgroup. Similarly given a subgroup $L \subseteq G$ with $H \subseteq L$ then $f_H(L)$ is a subgroup of G/H. □

A group G, with $|G| > 1$, is *simple* if its only normal subgroups are $\{e\}$ and G. A normal subgroup $H \lhd G$ is a *maximal proper normal subgroup* if $H \neq G$ and whenever $H \subseteq K \subseteq G$ with $K \neq G$ and $K \lhd G$ then $K = H$.

Let G be a group and $H \lhd G$. H is a maximal proper normal subgroup if and only if G/H is simple.

Our next result is of particular importance.

Theorem 1.4.3

Let G be a finite group. A sequence of subgroups

$$G = G_n \supset G_{n-1} \supset \ldots \supset G_1 \supset G_0 = \{e\}$$

exists such that

(i) $G_i \lhd G_{i+1}$ for $i = 0, \ldots, n-1$

(ii) G_{i+1}/G_i is simple for $i = 0, \ldots, n-1$.

Proof Choose first a maximal proper normal subgroup of G. If this is $\{e\}$ then G is simple and the result holds. If not, let this subgroup be H, then $|H| < |G|$ and G/H is simple. Now put $G_{n-1} = H$. Consider G_{n-1} and choose a maximal proper normal subgroup of G_{n-1}, call it G_{n-2}, then G_{n-1}/G_{n-2} is simple. We may continue this process, the finiteness of the set G will force an end after a finite number of steps. □

We call such a sequence a *composition series for G of length n*. The following theorem, known as the Jordan–Hölder theorem, is proved in most text-books on group theory.

Theorem 1.4.4

Let

$$G = G_n \supset G_{n-1} \supset \ldots \supset G_1 \supset G_0 = \{e\}$$

and

$$G = K_m \supset K_{m-1} \supset \ldots \supset K_1 \supset K_0 = \{e\}$$

be composition series for the finite group G. Then $m = n$ and for each $j \in \{0, \ldots, m-1\}$ there exists a distinct $i \in \{0, \ldots, n-1\}$ such that

$$K_{j+1}/K_j \approx G_{i+1}/G_i, \quad \text{and conversely.}$$

1.5 Permutation groups

Let Q be a finite non-empty set with $|Q| > 1$. The set A of all bijective functions of Q onto Q can be given the structure of a group, by using the composition of functions as a multiplication. We will write the operation of the function on the right hand side, so that if $q \in Q$ and $\alpha \in A$ then $q\alpha$ will denote the image of q under α.

Now let $\alpha, \alpha' \in A$ and define $\alpha\alpha'$ by $q(\alpha\alpha') = (q\alpha)\alpha'$ for all $q \in Q$. Then $\alpha\alpha' \in A$ and under this multiplication A becomes a group with identity 1_Q.

We call A the *group of all permutations of Q*. If $Q = \mathbf{n} = \{0, 1, \ldots, n-1\}$ we denote A by S_n. The subgroups G of A are called *permutation groups on Q*. Notice that if G is a permutation group on Q then the following conditions are satisfied:

(i) There exists a function $F: Q \times G \to Q$ defined by $F(q, g) = qg$, $q \in Q$; $g \in G$, called the *action of G on Q*.

(ii) $(qg)g_1 = q(gg_1)$ for $q \in Q$; $g, g_1 \in G$.

(iii) If $qg = qg_1$ for all $q \in Q$ then $g = g_1$ $(g, g_1 \in G)$.

If G is a permutation group on Q we call G *transitive on* Q if given $q, q' \in Q$ there exists $g \in G$ such that $q' = qg$. If G equals A, the set of all bijective mappings of Q onto Q, then it is transitive. If G is a subgroup of A it may not be transitive. Given $q \in Q$ the subset $qG = \{qg \mid g \in G\}$ is called the *orbit of* q. The set of distinct orbits of Q (with respect to G) partitions the set Q, for if $qG \cap q'G \neq \emptyset$ then $x = qg = q'g'$ for some $g, g' \in G$. Then $q' = qg(g')^{-1}$ so $q'G \subseteq qG$ and similarly $qG \subseteq q'G$. Finally for $q \in Q$ we have $q = q1_Q \in qG$. This partition is called the *orbit decomposition* of Q (with respect to G). There is an equivalence relation on Q associated with this partition, it is defined by

$$q \sim q' \Leftrightarrow q' = qg \quad \text{for some } g \in G.$$

A transitive permutation group yields an orbit decomposition involving one orbit, namely $Q = qG$ for any $q \in Q$.

Let G be a transitive permutation group on Q, a subset $P \subseteq Q$ such that $|P| > 1$, $P \neq Q$ and $P \cap Pg = P$ or $\emptyset$ for each $g \in G$ is called a *primitive block* of Q with respect to G. Thus each permutation either fixes P, i.e. $Pg = P$, or moves it away from P $(P \cap Pg = \emptyset)$. A *primitive* permutation group is a transitive permutation group with no primitive blocks. An *imprimitive* permutation group is a transitive permutation group with primitive blocks.

1.6 Exercises

1.1 Let $\mathbf{Z}$ be the set of all integers and let n be any positive integer. Define a relation $\mathcal{R}_n$ by

$$(a, a') \in \mathcal{R}_n \Leftrightarrow n \text{ divides } a - a'.$$

Prove that $\mathcal{R}_n$ is an equivalence relation. Describe the set $\mathbf{Z}/\mathcal{R}_n$.

1.2 Prove that a relation $\mathcal{R}: X \rightsquigarrow Y$ may be identified with a function $f: \mathcal{P}(X) \to \mathcal{P}(Y)$ satisfying the condition $f(\bigcup_{i \in I} A_i) = \bigcup_{i \in I} f(A_i)$ where $\{A_i \mid i \in I\}$ is any collection of subsets of X. (Note that $\mathcal{P}(X)$ is the set of subsets of X.)

1.3 Let A be a non-empty set and $\mathcal{R}$ an equivalence relation on A. Define the relation $\bar{\mathcal{R}}: A \rightsquigarrow A/\mathcal{R}$ by

$$\bar{\mathcal{R}} = \{(a, [a]) \mid a \in A\}.$$

Prove that $\bar{\mathcal{R}}$ is a surjective function.

1.4 If A is any non-empty set show that no surjective function $f: A \to \mathcal{P}(A)$ can exist. Can a surjective relation $\mathcal{R}: A \rightsquigarrow \mathcal{P}(A)$ exist?

1.5 Let $\mathcal{R}: X \rightsquigarrow Y$ be any relation, show that

$$1_{\mathfrak{R}(\mathcal{R})} \subseteq \mathcal{R} \circ \mathcal{R}^{-1}, \quad 1_{\mathfrak{D}(\mathcal{R})} \subseteq \mathcal{R}^{-1} \circ \mathcal{R}.$$

If $\mathcal{R}$ is injective establish

$$\mathcal{R}^{-1} \circ \mathcal{R} = 1_{\mathfrak{D}(\mathcal{R})}.$$

If $\mathcal{R}$ is a partial function prove that

$$\mathcal{R} \circ \mathcal{R}^{-1} = 1_{\mathfrak{R}(\mathcal{R})}.$$

If $\mathcal{R}$ is an injective function then

$$\mathcal{R}^{-1} \circ \mathcal{R} = 1_X, \quad \mathcal{R} \circ \mathcal{R}^{-1} = 1_{\mathfrak{R}(\mathcal{R})}.$$

If $\mathcal{R}$ is a surjective partial function show that

$$\mathcal{R} \circ \mathcal{R}^{-1} = 1_Y.$$

If $\mathcal{R}$ is a surjective function establish that

$$\mathcal{R} \circ \mathcal{R}^{-1} = 1_Y, \quad 1_X \subseteq \mathcal{R}^{-1} \circ \mathcal{R}.$$

If $\mathcal{R}$ is a surjective and injective function prove that

$$\mathcal{R} \circ \mathcal{R}^{-1} = 1_Y, \quad \mathcal{R}^{-1} \circ \mathcal{R} = 1_X.$$

1.6 If $\mathcal{R}: S \rightsquigarrow T$ is a semigroup relation show that

$$\mathcal{R}^{-1}: T \rightsquigarrow S$$

is also a semigroup relation.

1.7 Investigate the monoid analogues of theorems 1.2.2 and 1.2.3.

1.8 Let S be a finite semigroup. Let $s \in S$, consider the set $\{s, s^2, \ldots, s^n, \ldots\}$. Since S is finite we must have integers p and r such that $s^{p+r} = s^p$. Show that, if p and r are chosen suitably the set $s^p, s^{p+1}, \ldots, s^{p+r-1}$ is a subgroup of S.

1.9 Show that a finite semigroup S contains an element $s \in S$ satisfying $s^2 = s$.

1.10 A semigroup S is called *free* on Σ if a set Σ exists such that $S = \Sigma^+$. Prove that S is free on Σ if and only if $\Sigma \subseteq S$ and every element of S can be expressed uniquely as a finite product of elements of Σ.

1.11 If $S = \Sigma^+$ then $\Sigma = S \backslash S^2$ where $S^2 = \{s \cdot s_1 \mid s, s_1 \in S\}$.

1.12 S is a free semigroup if and only if

(i) $sa = sb \Rightarrow a = b$,

(ii) $as = bs \Rightarrow a = b$,

(iii) s has no identity element,

(iv) if $as = bt$ then either $a = b$, $a = bc$ or $b = ad$ for c, $d \in S$,

(v) each element has a finite number of left divisors.

1.13 Let T be a subsemigroup of Σ^+ then T is a free semigroup if and only if

$$Ts \cap T \neq \varnothing \quad \text{and} \quad sT \cap T \neq \varnothing$$

implies $s \in T$.

1.14 Find an example of a subsemigroup $T \subseteq \Sigma^+$ such that T is not a free semigroup.

2

Machines and semigroups

One of the achievements of modern science has been the realization that very few things in the world are completely static. The behaviour of many systems, both organic and synthetic, is influenced greatly by environmental changes. This interaction between a system and its environment can be vastly complicated and yet it is an area that we must try to understand if we are going to be in a position to predict the behaviour of the system and its effect on its environment.

The particular type of analysis that we present here is based on techniques that are generally referred to as *algebraic*. In some cases we will draw on established algebraic results but in general it is a new type of algebra that has arisen from a desire to understand the behaviour of a system in an environment. This is perhaps the most refreshing aspect of the theory. Here, for a change, is a subject whose motivation can be linked to very real problems in the modern world, a subject that has a short but dramatic history and one which has played a large role in the development of the fundamentals of computer science. However its achievements have not been restricted to this case alone and we hope to illustrate this when we examine the examples at the end of this chapter.

In many of the systems environmental changes alter the behaviour of the system and these changes in behaviour then affect the environment in some way. In other examples the only thing altered in the system is some internal quality. These latter systems are easier to analyse mathematically and so we shall start our considerations with them, although, as we see later, the other types of system can also be brought into this discussion in a meaningful and elementary way.

2.1 State machines

Suppose that we have a system which is reacting to certain changes in its immediate environment and suppose, further, that this reaction is entirely one of changes in the internal qualities of the system. First of all we identify within the system the set of these internal qualities which we will call *internal states*. If we denote this set of internal states by the set Q we can then agree that at any given time, t, the system is in a particular internal state, $q(t)$, which is an element of the set Q. What these internal states are is not of great importance in general, we deliberately keep the definition fairly vague in order that we can then apply our model to a great many distinct situations.

Some examples of systems and possible sets of internal states may help to give a more intuitive idea of what we mean. Consider an electronic system, which involves various electrical components, such as transistors, connected together in a complex electrical circuit. As currents flow through the system some of these transistors 'fire', while others do not. If at a given time, t, we have a complete knowledge of what the various components of the circuit are doing, either firing or not firing, then we say that we know the state, $q(t)$, of the system at the time t. The set Q will then be the set of all the states $q(t)$ that are possible at some time or other. Obviously the larger the system, the more difficult will the definition of an internal state be, and the larger the set Q of all internal states. However, the total number of internal states will always be finite in examples of this kind.

For a biological example consider a single cell from a biological organism. Inside the cell there will be many chemical reactions taking place as the cell performs its role in the organism. Many chemicals are being formed and consumed in the cell, but at a given time, t, it is possible to conceive, at least theoretically, that each chemical has reached a certain concentration in the cell as a result of the reactions taking place. Thus the internal state, $q(t)$, at that time would be a list of all the chemical concentrations in the cell then. Clearly this would be a phenomenally complex piece of information, but the number of chemicals involved would again be finite. In this case the complete set of internal states may not be finite, since each chemical could clearly exist in one of an infinite number of concentrations. We overcome this difficulty by using the idea of a threshold. It is clear that some chemicals can exist in very tiny concentrations without substantially changing the behaviour of the cell, and as it is the behaviour of the cell, and in particular its response to changes in its environment, that interests us, we can often replace the infinite sets of chemical concentrations by finite ones, since the behaviour of the cell may change only after a chemical concentration has crossed a threshold value. (See example 2.8.)

Let us now consider the system as being described by a finite set Q of internal states. Changes in its environment will in many cases force changes in its internal state and we will now make the added assumption that this is the only way that internal states can be changed. So that if there is no change in the environment between times t and t_1 then $q(t) = q(t_1)$.

How do we model the environmental influences? Consider the environment and the system at a given time, t, and note all the relevant environmental factors that can affect the system; this particular environmental profile will be denoted by $\sigma(t)$. The set of all such $\sigma(t)$ that can affect the system is called Σ, the set of *environmental inputs* or the *input alphabet*.

In the examples discussed above the environmental input to the computer system at time t is either a *pulse of electricity* applied to the system or *no electrical charge*. For the biological example the input will be a particular profile involving, perhaps temperature, quantity of light, concentrations of various chemicals etc. and as before it may be considered to be a finite set by applying the threshold principle.

We are then left with two finite sets, Q representing the internal states of the system, and Σ representing the possible environmental influences acting on the system. Since the system will react to different environmental inputs by changing its internal state, the final stage in the modelling of the system is a function that tells us how the system will behave. We agree first that the internal changes and the reception of inputs take place in the context of a suitable discrete time scale based on the length of time that the system takes to react. In this way we will remove problems associated with the influence of time on the inputs and the internal states.

Let the system be in state $q \in Q$ and suppose that the environment changes to $\sigma \in \Sigma$. The change will cause the state to change at the next point on the time scale to a new state $q' \in Q$ and so we have the *resultant* of applying the environmental input σ to the system in state q. If we specify this resultant for some of the possible combinations of internal state and environmental input, we will be specifying a partial function $F: Q \times \Sigma \to Q$ in such a way that $F((q, \sigma)) = q'$ where $q \in Q$, $\sigma \in \Sigma$ and q' is the result of applying σ to the system in state q.

A *state machine* or *semiautomaton* is a triple $\mathcal{M} = (Q, \Sigma, F)$ where Q and Σ are finite sets and F is a partial function $F: Q \times \Sigma \to Q$. (We allow the possibilities that either Q or Σ or both are empty.) A state machine $\mathcal{M} = (Q, \Sigma, F)$ is called *complete* if the partial function $F: Q \times \Sigma \to Q$ is in fact a *function*. In this situation we can specify what the resultant $F((q, \sigma))$ is for all possible combinations of $q \in Q$ and $\sigma \in \Sigma$.

Such a system is clearly very general and can be applied to many different situations. It is almost too general, from a mathematical point of view, and the fact that we can investigate such systems successfully using algebraic techniques, is, in my opinion, one of the most remarkable achievements of modern mathematics.

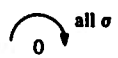
One advantage of such a general definition is that it is easy to find simple examples and their study amply repays the effort involved. We shall look at some now.

Examples 2.1

(i) Some simple cases are where Q and Σ are both singletons. Let $Q = \{0\}$ and $\Sigma = \{\sigma\}$, then we can have $F: Q \times \Sigma \to Q$ defined by $F(0, \sigma) = 0$. This is illustrated with a simple diagram:

where the arrow is labelled by the only input, σ.

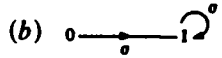
(ii) Suppose that $|Q| = 1$ and Σ is any finite set, we don't really get anything very different, just $F((0, \sigma)) = 0 \ \forall \sigma \in \Sigma$, or in diagrammatic form:



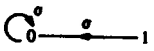
(iii) Letting $|Q| > 1$ does introduce some more interesting examples, thus if $Q = \{0, 1\}$ and $|\Sigma| = 1$ we could have any of the following:

(a) 

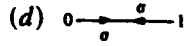
or

(b) 

or

(c) 

or

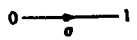
(d) 

(iv) These examples are all complete, and in fact incomplete state machines need not have any arrows. For example:

0 could represent $(\{0\}, \{\sigma\}, F)$ where

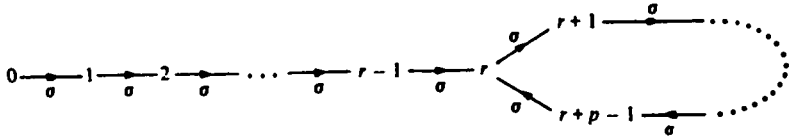
$F: Q \times \Sigma \to Q$ is not defined for $(0, \sigma) \in Q \times \Sigma$.

Another incomplete state machine is:



and here $F(1, \sigma)$ is undefined.

(v) We will introduce a *cyclic state machine* as follows. Let p, r be positive integers and put $Q = \{0, 1, 2, \ldots, r+p-1\}$, $\Sigma = \{\sigma\}$. Consider the diagram



so that $F(0, \sigma) = 1$, $F(1, \sigma) = 2$ etc.
This is called the cyclic state machine with *stem* of length r and *cycle* of length p; we note that this machine is complete.

These diagrams, or directed graphs, are sometimes quite useful tools. In these simple cases they clearly define the state machine precisely, and we will often use them for this purpose.

They can, however, also tell us something about the properties of the state machines. Take a look at the cyclic machine above, the states $0, \ldots, r-1$ have the property that once the machine leaves them it can never return. We could call these 'states of no return'. The cycle of states $r, r+1, \ldots, r+p-1$ is a 'cycle of no escape'.

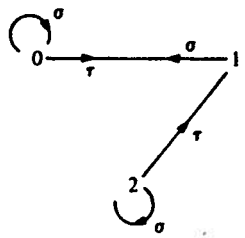
Another way of specifying a state machine is by writing out the partial function F in tabular form, for example:

(vi) $Q = \{0, 1, 2\}$, $\Sigma = \{\sigma, \tau\}$ and

F	0	1	2
σ	0	0	2
τ	1	$\varnothing$	1

(Here $F((1, \tau))$ is undefined, we write it as $\varnothing$ in the table.)

This represents the same machine as the diagram:

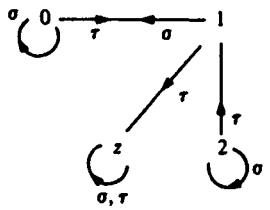


The fact that $F((1, \tau))$ is undefined is indicated on the diagram by the lack of an arrow labelled by τ emanating from the state 1.

An incomplete state machine can be completed by introducing new arrows from states that are lacking them. However we choose a more systematic method.

We introduce a new state to the machine and arrange for all the missing arrows to go to this new state.

For example in (vi) above we introduce the new state z so that $Q = \{0, 1, 2, z\}$, $\Sigma = \{\sigma, \tau\}$ and the graph of the completed machine is:



Formally let $\mathcal{M} = (Q, \Sigma, F)$ be an incomplete state machine. Define the *completion* $\mathcal{M}^c = (Q', \Sigma, F')$, of $\mathcal{M}$ by putting

$$Q' = Q \cup \{z\}$$

where $z \notin Q$, and

$$F'((q', \sigma)) = \begin{cases} F(q, \sigma) \text{ if } q \in Q \text{ and } F(q, \sigma) \text{ is defined} \\ z \quad \text{otherwise.} \end{cases}$$

The new state z is called the *sink state* of $\mathcal{M}^c$.

We will examine some practical examples of such machines at the end of the chapter. Our next task is to look at the way these machines operate.

Generally speaking we will present the machine $\mathcal{M} = (Q, \Sigma, F)$ with a symbol $\sigma \in \Sigma$ while it is in some state, say $q \in Q$. The machine then moves to state $F((q, \sigma)) \in Q$. This notation is a little cumbersome and we will introduce the idea of the state mapping induced by the input.

This concept is defined thus:

let $\sigma \in \Sigma$, define $F_\sigma : Q \to Q$ by

$qF_\sigma = F((q, \sigma))$ for each $q \in Q$.

Since the machine may not be complete, F_σ may only be a partial function of Q to itself. Each input symbol σ from Σ yields a partial function $F_\sigma : Q \to Q$.

Now suppose that σ is applied to the machine in the state q and consequently the machine moves to state qF_σ. (Using the usual convention that $qF_\sigma = \varnothing$ if $F((q, \sigma))$ is undefined.) If, further, another input, say $\sigma' \in \Sigma$, is applied to the machine we get the resultant state $qF_\sigma F_{\sigma'}$. We may extend our notation in the following way. Let $\alpha \in \Sigma^+$ be a word of length at least 1 with symbols from Σ.

Suppose that $\alpha = \sigma_1 \sigma_2 \ldots \sigma_k$ then we define

$$F_\alpha : Q \to Q$$

by

$$qF_\alpha = qF_{\sigma_1} F_{\sigma_2} \ldots F_{\sigma_k}.$$

Now it is perhaps apparent why we are writing the result of state mappings in the form qF_σ rather than $F_\sigma(q)$, it is caused by our convention of writing words from left to right!

Each word from Σ^+ will therefore correspond to some partial function of Q to itself.

Returning, once more, to example 2.1(vi) we note that some of the partial functions induced by words from Σ^+ are:

	0	1	2
F_σ	0	0	2
F_τ	1	$\varnothing$	1
$F_{\sigma\sigma}$	0	0	2
$F_{\tau\tau}$	$\varnothing$	$\varnothing$	$\varnothing$
$F_{\sigma\tau}$	1	1	1
$F_{\tau\sigma}$	0	$\varnothing$	0

Note that $F_{\tau\tau}$ is the empty function $\theta : Q \to Q$ and so we have a natural example of what one might have originally thought was a rather artificial concept. The function $F_{\tau\sigma}$ is, like F_τ a partial function.

2.2 The semigroup of a state machine

The state set of a state machine is finite and so the number of partial mappings definable on the state set is also finite. Therefore the

number of *distinct* state mappings induced by words from the symbol set is also finite. Consequently some words will yield the same state mappings. We will use this idea to introduce a relation on the free semigroup generated by the symbol set.

Let $\mathcal{M} = (Q, \Sigma, F)$ be a state machine and consider the set Σ^+ of all words of length greater than or equal to 1 in the alphabet Σ. Define a relation $\sim$ on Σ^+ by

$$\alpha \sim \beta \Leftrightarrow F_\alpha = F_\beta \quad \text{where } \alpha, \beta \in \Sigma^+$$

This relation is easily seen to be an equivalence relation. Since Σ^+ has a natural semigroup structure, using concatenation of words as the operation, it is natural to ask whether $\sim$ is a congruence on Σ^+. This is indeed the case, for example if $\alpha, \beta, \gamma \in \Sigma^+$ and $\alpha \sim \beta$ then $F_\alpha = F_\beta$ and for any $q \in Q$, $qF_{\gamma\alpha} = qF_\gamma F_\alpha = (qF_\gamma)F_\alpha = (qF_\gamma)F_\beta = qF_{\gamma\beta}$ and so $F_{\gamma\alpha} = F_{\gamma\beta}$ which yields $\gamma\alpha \sim \gamma\beta$, etc. We now construct the quotient semigroup $\Sigma^+/\sim$ and call it the *semigroup of the state machine* $\mathcal{M}$, the notation used being $\mathbf{S}(\mathcal{M})$. The elements of $\mathbf{S}(\mathcal{M})$ will be equivalence classes $[\alpha]$, $\alpha \in \Sigma^+$.

We have already noted that each $\sigma \in \Sigma$ defines a partial mapping $F_\sigma : Q \to Q$ and so there is a natural function $\mathbf{F} : \Sigma \to \mathbf{PF}(Q)$, given by $\mathbf{F}(\sigma) = F_\sigma$ for $\sigma \in \Sigma$. If we denote by $\langle \mathbf{F}(\mathcal{M}) \rangle$ the subsemigroup of $\mathbf{PF}(Q)$ generated by the set of functions $\{F_\sigma \mid \sigma \in \Sigma\}$ we obtain an isomorphic copy of the semigroup $\mathbf{S}(\mathcal{M})$ of the state machine $\mathcal{M}$. To see this just note that there is a surjection θ from Σ^+ onto $\langle \mathbf{F}(\mathcal{M}) \rangle$ defined by $\theta(\alpha) = F_\alpha$ for $\alpha \in \Sigma^+$, with corresponding congruence defined by the relation $\sim$. The first isomorphism theorem for semigroups yields the result. We thus have:

Proposition 2.2.1

Let $\mathcal{M} = (Q, \Sigma, F)$ be a state machine and $\langle \mathbf{F}(\mathcal{M}) \rangle$ the subsemigroup of $\mathbf{PF}(Q)$ generated by $\{F_\sigma \mid \sigma \in \Sigma\}$, then $\langle \mathbf{F}(\mathcal{M}) \rangle \cong \mathbf{S}(\mathcal{M}) = \Sigma^+/\sim$. Furthermore $\mathbf{S}(\mathcal{M})$ is a finite semigroup.

The last statement follows from the fact that $\mathbf{PF}(Q)$ is finite when Q is finite.

The semigroup $\mathbf{PF}(Q)$ is actually a monoid and while $\mathbf{S}(\mathcal{M})$ may also be a monoid it can happen that $\mathbf{S}(\mathcal{M})$ does not possess an identity. We can easily construct a monoid from the state machine $\mathcal{M}$ by forming the monoid Σ^* of all words in Σ, including the empty word Λ, and extending the relation $\sim$ to Σ^* by putting

$$\alpha \sim \beta \Leftrightarrow F_\alpha = F_\beta \quad \text{for } \alpha, \beta \in \Sigma^*.$$

Again $\sim$ is a congruence, but $\Sigma^*/\sim$ is a finite monoid isomorphic to $\langle F(\mathcal{M}) \rangle \cup \{1_Q\}$. We write $\Sigma^*/\sim$ as $\mathbf{M}(\mathcal{M})$, and call it the *monoid of* $\mathcal{M}$.

Note that in both cases the relations $\sim$ defined on Σ^+ and Σ^* depend on the state machine $\mathcal{M}$. However, it is quite possible for different state machines to have the same, or at least isomorphic, semigroups.

Given a state machine $\mathcal{M} = (Q, \Sigma, F)$ we have now associated with it a semigroup $\mathbf{S}(\mathcal{M})$. In many situations it is more convenient to study this semigroup rather than the original machine $\mathcal{M}$. However we don't want to lose sight of the set of states and so we consider the pair $(Q, \mathbf{S}(\mathcal{M}))$ consisting of the set of states Q of $\mathcal{M}$ and the semigroup $\mathbf{S}(\mathcal{M})$ of $\mathcal{M}$. Each element of $\mathbf{S}(\mathcal{M})$ is an equivalence class of Σ^+, which acts on Q as follows: $q[\alpha] = qF_\alpha$ where $q \in Q$, $\alpha \in \Sigma^+$. This is an example of a transformation semigroup and it is these that we will be studying in detail.

A *transformation semigroup* is a pair (Q, S) consisting of a finite set Q, a finite semigroup S and an *action* of S on Q, that is a partial function $\lambda : Q \times S \to Q$ satisfying two conditions:

(i) $\lambda(\lambda(q, s), s_1) = \lambda(q, ss_1)$ for all $q \in Q$; $s, s_1 \in S$.

(ii) $\lambda(q, s) = \lambda(q, s_1)$ for all $q \in Q$ implies $s = s_1$ where $s, s_1 \in S$.

It is usual to write $\lambda(q, s)$ as $q \cdot s$ or qs for $q \in Q$, $s \in S$ and these conditions become

(i)′ $(qs)s_1 = q(ss_1)$ for all $q \in Q$; $s, s_1 \in S$.

(ii)′ $qs = qs_1$ for *all* $q \in Q$ implies $s = s_1$ where $s, s_1 \in S$.

We write the operation of S on Q on the right to preserve the connection with state machines. Notice that there is a natural embedding of the semigroup S into the monoid $\mathbf{PF}(Q)$ obtained by defining $\theta(s) : Q \to Q$ to be given by $q\theta(s) = qs$ for each $q \in Q$, and each $s \in S$. Then $\theta : S \to \mathbf{PF}(Q)$ is a semigroup monomorphism. Conversely given any set Q and a subsemigroup $S \subseteq \mathbf{PF}(Q)$ then (Q, S) is a transformation semigroup.

Associated with any state machine $\mathcal{M} = (Q, \Sigma, F)$ there is then a transformation semigroup $(Q, \mathbf{S}(\mathcal{M}))$ which we will denote by $\mathbf{TS}(\mathcal{M})$ and call the *transformation semigroup of* $\mathcal{M}$.

Now each transformation semigroup determines a state machine, for suppose that $\mathcal{A} = (Q, S)$ is a transformation semigroup, we define the state machine $\mathcal{M} = (Q, S, F)$ where

$$F : Q \times S \to Q$$

is given by

$$F(q, s) = qs \quad \text{for all } q \in Q, s \in S.$$

Clearly $\mathcal{M}$ is a state machine, we call it the *state machine of* (Q, S) and denote it by $\mathbf{SM}(\mathcal{A})$. The relationship between state machines and transformation semigroups is very close.

In some situations the semigroup $\mathbf{S}(\mathcal{M})$ of a state machine $\mathcal{M}$ is in fact a monoid and $\mathbf{TS}(\mathcal{M})$ is a *transformation monoid.* Generally we define a transformation monoid as a transformation semigroup (Q, S) where S is a monoid *and* the identity 1 of S satisfies

$$q \cdot 1 = q \quad \text{for all } q \in Q.$$

For a given state machine $\mathcal{M} = (Q, \Sigma, F)$ we may define the *transformation monoid of* $\mathcal{M}$, $\mathbf{TM}(\mathcal{M})$, as being $(Q, \mathbf{M}(\mathcal{M}))$.

Now is the time to look at some examples.

Examples 2.2

(i) The examples of state machines discussed in 2.1(i) and (ii) both yield the transformation monoid $(\{0\}, S)$ where S is the group of order 1.

(ii) The transformation semigroups of the examples in 2.1(iii) are

(a) $(\{0, 1\}, S)$, which is a transformation monoid, with $S = \{1_Q\}$;

(b) $(\{0, 1\}, \{\sigma\})$, which is not a transformation monoid although $\sigma^2 = \sigma$;

(c) also of the form $(\{0, 1\}, \{\sigma\})$ with $\sigma^2 = \sigma$ although the action is not the same;

(d) $(\{0, 1\}, \{\sigma, \sigma^2\})$ with $\sigma^2 = 1_Q$.

(iii) Example 2.1(vi) has the transformation semigroup $(\{0, 1, 2\}, \{\theta, \sigma, \tau, \sigma\tau, \tau\sigma, \sigma\tau\sigma\})$ with the semigroup composition given by the following table:

	θ	σ	τ	$\sigma\tau$	$\tau\sigma$	$\sigma\tau\sigma$
θ	θ	θ	θ	θ	θ	θ
σ	θ	σ	$\sigma\tau$	$\sigma\tau$	$\sigma\tau\sigma$	$\sigma\tau\sigma$
τ	θ	$\tau\sigma$	θ	τ	θ	$\tau\sigma$
$\sigma\tau$	θ	$\sigma\tau\sigma$	θ	$\sigma\tau$	θ	$\sigma\tau\sigma$
$\tau\sigma$	θ	$\tau\sigma$	τ	τ	$\tau\sigma$	$\tau\sigma$
$\sigma\tau\sigma$	θ	$\sigma\tau\sigma$	$\sigma\tau$	$\sigma\tau$	$\sigma\tau\sigma$	$\sigma\tau\sigma$

Since we will be repeatedly dealing with transformation semigroups it will be convenient to introduce some notation to help us refer to some of the more common types. First we will consider a general finite set Q. Let $q \in Q$, then there is a mapping $\bar{q} : Q \to Q$ defined by $y\bar{q} = q$ for all $y \in Q$. Thus $\bar{q}$ is the constant mapping defined by the element q. The set of all the constant mappings on Q generates a semigroup $\bar{Q}$ as a subsemigroup of $\mathbf{PF}(Q)$. We can now consider the transformation semigroup $(Q, \bar{Q})$.

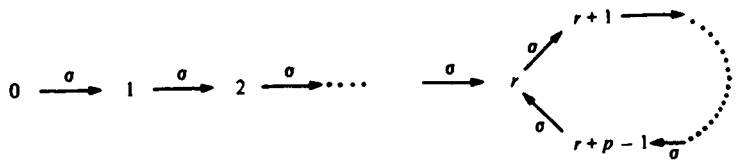
Let $\mathcal{A} = (Q, S)$ be any transformation semigroup, define the *closure* $\bar{\mathcal{A}}$ of $\mathcal{A}$ to be the transformation semigroup $(Q, \langle S \cup \bar{Q} \rangle)$. We call $\mathcal{A}$ *closed* if $\bar{\mathcal{A}} = \mathcal{A}$.

Given any transformation semigroup $\mathcal{A} = (Q, S)$ define the transformation monoid $\mathcal{A}^{\cdot} = (Q, S \cup \{1_Q\})$. For any finite set Q we can form a transformation semigroup $\mathcal{Q} = (Q, \varnothing)$. Then if n is a positive integer recall that the set $\mathbf{n} = \{0, 1, \ldots, n-1\}$ and so we have a transformation semigroup, also denoted by $\mathbf{n}$ and given by $\mathbf{n} = (\mathbf{n}, \varnothing)$. We can now specify some of the transformation semigroups in examples 2.2, these are:

2.2(i) $\mathbf{1}^{\cdot}$, 2.2(ii) $\mathbf{2}^{\cdot}$.

The example 2.2(ii)(b) will be denoted by $\mathscr{C}$.

The transformation semigroup generated by the state machine



will be written $\mathscr{C}_{(p,r)}$

If G is a group then G may be considered as the transformation monoid (G, G) where the group G acts on the set G by right multiplication, that is $g'g = g' \cdot g$ $(g' \in G, g \in G)$, we will denote this transformation monoid by $\mathscr{G}$, and since the monoid is a group it will be sensible to call it a *transformation group.* Thus example 2.1(iii)(d) generates $(\mathbf{Z}_2, \mathbf{Z}_2)$, a transformation group; we will write this as $\mathbf{Z}_2$.

In the case of the transformation group $\mathscr{G}$ formed from a group G it is clear that the action is faithful. However if S is a semigroup the action of S on the set S defined by right multiplication need not be faithful. This is a special case of a more general situation.

Suppose that Q is a finite set and S is a semigroup, suppose further that an action qs $(q \in Q, s \in S)$ is given. The pair (Q, S) may not be a transformation semigroup even if the action satisfies $(qs)s_1 = q(ss_1)$ for all $q \in Q$, $s, s_1 \in S$. However such a pair may be converted into a transformation semigroup. Let $\sim$ define a relation on S defined by $s \sim s_1 \Leftrightarrow qs = qs_1$ for all $q \in Q$. Then $\sim$ is a congruence and we may form the quotient

semigroup $S/\sim$. The pair $(Q, S/\sim)$ now becomes a transformation semigroup with action defined by $q[s] = qs$, $q \in Q$, $[s] \in S/\sim$. We call this the transformation semigroup *represented* by the pair (Q, S).

Now if S is a semigroup then the pair (S, S) represents a transformation semigroup $(S, S/\sim)$.

Another way of defining a transformation semigroup from an arbitrary semigroup is to consider the 'semigroup made into a monoid' by the adjunction of an identity element. So if S is a semigroup which is not a monoid then $S^{\sim} = S \cup \{e\}$ where $e \notin S$ is suitably chosen and is defined to act as an identity for $S^{\sim}$. Then we can construct a transformation semigroup $(S^{\sim}, S)$ with action by right multiplication; this is denoted by $\mathcal{S}$. If S is a monoid then (S, S) is a transformation monoid, also written as $\mathcal{S}$.

A transformation semigroup $\mathcal{A} = (Q, S)$ may not be complete, that is qs may not be defined for some $q \in Q$, $s \in S$. The *completion*, $\mathcal{A}^c$, is defined to be

$$\mathcal{A}^c = (Q', S)$$

where

$$Q' = Q \cup \{z\}$$

for some $z \notin Q$ and

$$q' \cdot s = \begin{cases} qs & \text{if } q \in Q \text{ and } qs \text{ is defined in } \mathcal{A} \\ z & \text{otherwise.} \end{cases}$$

Naturally, if $\mathcal{A}$ is complete we will define $\mathcal{A}^c = \mathcal{A}$.

If $\mathcal{A} = (Q, S)$ is a transformation monoid, $Q \neq \emptyset$ and S is a group, we call $\mathcal{A}$ a transformation group. If $\mathcal{A} = (Q, S)$ is such that either $\mathcal{A}$ is a transformation group or $Q \neq \emptyset$ and $S = \emptyset$ we say that $\mathcal{A}$ is a *generalized transformation group.*

2.3 Homomorphisms and quotients

Let $\mathcal{M} = (Q, \Sigma, F)$ and $\mathcal{M}' = (Q', \Sigma', F')$ be state machines. Let $\alpha : Q \to Q'$, $\beta : \Sigma \to \Sigma'$ be mappings such that

$$\alpha(qF_\sigma) \subseteq (\alpha(q))F'_{\beta(\sigma)}$$

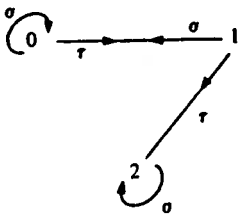
for any $q \in Q$, $\sigma \in \Sigma$. (This means that if qF_σ is undefined we put $\alpha(qF_\sigma) = \emptyset$ and if qF_σ is defined then so is $(\alpha(q))F'_{\beta(\sigma)}$ and $\alpha(qF_\sigma) = (\alpha(q))F'_{\beta(\sigma)}$.)

We call the pair (α, β) a *state machine homomorphism* from $\mathcal{M}$ to $\mathcal{M}'$ and write $(\alpha, \beta) : \mathcal{M} \to \mathcal{M}'$.

If α and β are both one-one mappings then we call (α, β) a *monomorphism* and if α and β are both onto mappings then (α, β) is called an *epimorphism.* An *isomorphism* of state machines is both a monomorphism and an epimorphism, in this case we write $\mathcal{M} \cong \mathcal{M}'$.

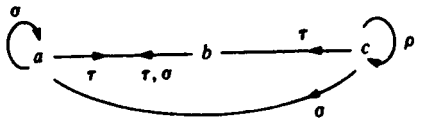
Example 2.3

Let $\mathcal{M} = (Q, \Sigma, F)$ be the state machine defined by the diagram



This is example 2.1 (vi).

If $\mathcal{M}' = (Q', \Sigma', F')$ is the state machine defined by the diagram



where $Q' = \{a, b, c\}$ and $\Sigma' = \{\sigma, \tau, \rho\}$,
define

$$\alpha : Q \to Q' \quad \text{by } \alpha(0) = \alpha(2) = a, \alpha(1) = b$$

$$\beta : \Sigma \to \Sigma' \quad \text{by } \beta(\sigma) = \sigma, \beta(\tau) = \tau.$$

Then $(\alpha, \beta) : \mathcal{M} \to \mathcal{M}'$ is a homomorphism; note that

$$\alpha(2F_\tau) = \emptyset \subseteq (\alpha(2))F'_{\beta(\tau)} = b$$

$$\alpha(0F_\sigma) = a = (\alpha(0)F'_{\beta(\sigma)})$$

etc.

If $\mathcal{A} = (Q, S)$, $\mathcal{A}' = (Q', S')$ are transformation semigroups, $f : Q \to Q'$ is a mapping and $g : S \to S'$ a semigroup homomorphism, then the pair (f, g) is said to be a *transformation semigroup homomorphism* from $\mathcal{A}$ to $\mathcal{A}'$ if

$$f(qs) \subseteq f(q) \cdot g(s) \quad \text{for all } q \in Q, s \in S.$$

(It should be realized that in incomplete transformation semigroups the left hand side may be undefined and is then by convention the empty set.) As before we write $(f, g) : \mathcal{A} \to \mathcal{A}'$.

(f, g) is a *monomorphism* if f and g are one-one; an *epimorphism* if f and g are onto; and an *isomorphism* if (f, g) is both a monomorphism and an epimorphism, we then write $\mathscr{A} \cong \mathscr{A}'$.

Theorem 2.3.1

Let $\mathscr{M} = (Q, \Sigma, F)$, $\mathscr{M}' = (Q', \Sigma', F')$ be complete state machines and $(\alpha, \beta): \mathscr{M} \to \mathscr{M}'$ a homomorphism with α onto. There exists a homomorphism

$$(f_\alpha, g_\beta): \mathbf{TS}(\mathscr{M}) \to \mathbf{TS}(\mathscr{M}').$$

Proof Define $f_\alpha: Q \to Q'$ by $f_\alpha = \alpha$. Let $S = \mathbf{S}(\mathscr{M})$, $S' = \mathbf{S}(\mathscr{M}')$ and suppose that $s \in S$. Then there exists $a \in \Sigma^+$ such that $s = [a]$, the $\sim$-equivalence class containing a. Suppose that $a = \sigma_1 \ldots \sigma_n$, $\sigma_i \in \Sigma$ define $g_\beta(s) = [\beta(a)]'$ where $[\beta(a)]'$ is the $\sim'$-equivalence class containing $\beta(a) = \beta(\sigma_1) \ldots \beta(\sigma_n) \in (\Sigma')^+$. (Note that $\sim$ is induced by $\mathscr{M}$ and $\sim'$ is induced by $\mathscr{M}'$.)

We must first establish that $g_\beta: S \to S'$ is well-defined. Suppose that $s = [b]$ where $b \in \Sigma^+$, then $b = \tau_1 \ldots \tau_m$ where $\tau_j \in \Sigma$. Now for any $q \in Q$, $qF_a = qF_b$. Let $q' \in Q'$, there exists $q \in Q$ such that $q' = \alpha(q)$. Then $q'F'_{\beta(a)} = (\alpha(q))F'_{\beta(a)}$ and $q'F'_{\beta(b)} = (\alpha(q))F'_{\beta(b)}$. However $\alpha(qF_a) = \alpha(qF_b)$ and then $\alpha(qF_a) = (\alpha(q))F'_{\beta(a)} = (\alpha(q))F'_{\beta(b)}$ so $q'F'_{\beta(a)} = q'F'_{\beta(b)}$. Thus $\beta(a) \sim' \beta(b)$ and g_β is well-defined. Now let $q \in Q$, $s \in S$, then $f_\alpha(qs) = \alpha(qs) = \alpha(qF_a) = (\alpha(q))F'_{\beta(a)} = f_\alpha(q)[\beta(a)]' = f_\alpha(q)g_\beta(s)$, where $s = [a]$ and $a \in \Sigma^+$. □

This result gives us some useful information concerning the relationship between a state machine homomorphism and a homomorphism of the related transformation semigroups. We consider, now, two elementary results that link state machines with transformation semigroups.

Theorem 2.3.2

Let $\mathscr{A} = (Q, S)$ be a transformation semigroup; then

$$\mathbf{TS}(\mathbf{SM}(\mathscr{A})) \cong \mathscr{A}.$$

Proof Let $\mathbf{SM}(\mathscr{A}) = (Q, S, F)$ and consider the semigroup $K = \langle \mathbf{F}(\mathbf{SM}(\mathscr{A})) \rangle$ generated by $\mathbf{SM}(\mathscr{A})$. There is clearly an isomorphism $\theta: S \to K$ defined by $\theta(s) = F_s$, $s \in S$ and this yields the isomorphism $(1_Q, \theta): \mathscr{A} \to \mathbf{TS}(\mathbf{SM}(\mathscr{A}))$. □

Theorem 2.3.3

Let $\mathscr{M} = (Q, \Sigma, F)$ be a state machine; there exists a state machine monomorphism

$$(\alpha, \beta): \mathscr{M} \to \mathbf{SM}(\mathbf{TS}(\mathscr{M})).$$

As with most algebraic systems, homomorphisms are closely linked with 'congruence' relations. Suppose that we have a homomorphism of state machines

$$(\alpha, \beta): \mathscr{M} \to \mathscr{M}' \quad \text{where } \mathscr{M} = (Q, \Sigma, F) \text{ and } \mathscr{M}' = (Q', \Sigma', F').$$

The mapping $\alpha: Q \to Q'$ induces an equivalence relation R_α on the set Q defined by

$$qR_\alpha q_1 \Leftrightarrow \alpha(q) = \alpha(q_1) \quad \text{for } q, q_1 \in Q.$$

The relation R_α satisfies the following condition: let $qR_\alpha q_1$ and $\sigma \in \Sigma$ and suppose that qF_σ and q_1F_σ are both defined, then

$$(qF_\sigma)R_\alpha(q_1F_\sigma).$$

This follows because $\alpha(qF_\sigma) = \alpha(q)F'_{\beta(\sigma)} = \alpha(q_1)F'_{\beta(\sigma)} = \alpha(q_1F_\sigma)$.

The relation R_α is an example of an *admissible* relation on Q. Formally if $\mathscr{M} = (Q, \Sigma, F)$ is a state machine then a relation R on Q is *admissible* if:

(i) R is an equivalence relation;

(ii) given $q, q_1 \in Q$, $\sigma \in \Sigma$ such that qRq_1 and both qF_σ, q_1F_σ are defined then $(qF_\sigma)R(q_1F_\sigma)$.

An admissible relation R defines a partition on the set Q of the state machine $\mathscr{M} = (Q, \Sigma, F)$. Suppose we denote this partition by $\pi = \{H_i\}_{i \in I}$ where each H_i is an equivalence class of Q for $i \in I$. Then $Q = \bigcup_{i \in I} H_i$ and $H_i \cap H_j = \varnothing$ for $i \neq j$, $i, j \in I$. Furthermore given $H_i \in \pi$ and $\sigma \in \Sigma$ we form the set $H_iF_\sigma = \{qF_\sigma \mid q \in H_i\}$ and then $H_iF_\sigma \subseteq H_j$ for some $j \in I$. (Clearly H_iF_σ may be empty.) Consequently we have a special type of partition π which will be called an admissible partition.

Thus a *partition* $\pi = \{H_i\}_{i \in I}$ of the state set Q of the state machine $\mathscr{M} = (Q, \Sigma, F)$ is called *admissible* if given $i \in I$, $\sigma \in \Sigma$ either there exists $j \in I$ such that

$$H_iF_\sigma \subseteq H_j$$

or

$$H_iF_\sigma = \varnothing.$$

If the machine $\mathscr{M}$ is complete then the choice of j, given i and σ, is unique. For incomplete machines this is not always true.

Turning now to transformation semigroups we make the following parallel definitions.

Let $\mathscr{A} = (Q, S)$ be a transformation semigroup, an *admissible relation on* Q is a relation R such that if $q, q_1 \in Q$, $s \in S$, $qs \neq \emptyset$, $q_1 s \neq \emptyset$ and qRq_1 then $qsRq_1 s$.

A *partition* $\pi = \{H_i\}_{i \in I}$ on Q is *admissible* if given $i \in I$, $s \in S$ either there exists $j \in I$ such that

$$H_i s \subseteq H_j$$

or

$$H_i s = \emptyset.$$

The idea of an admissible partition leads to a procedure for constructing quotient systems in the following way.

Let $\mathscr{M} = (Q, \Sigma, F)$ be a state machine and $\pi = \{H_i\}_{i \in I}$ an admissible partition on Q, construct a state machine $\mathscr{M}/\pi = (Y, \Sigma, G)$ by defining $Y = \pi$, the set of π-blocks, and putting $H_i G_\sigma = H_j$ where

$$\left.\begin{array}{l} H_i F_\sigma \subseteq H_j \\ H_i G_\sigma = \emptyset \quad \text{if } H_i F_\sigma = \emptyset \end{array}\right\} (i, j \in I, \sigma \in \Sigma).$$

This definition of G_σ is well-defined since π is a partition and admissible. The state machine $\mathscr{M}/\pi$ is called the *quotient state machine* of $\mathscr{M}$ *with respect to* π.

If we change the scene to that of transformation semigroups a similar construction emerges.

Let $\mathscr{A} = (Q, S)$ be a transformation semigroup and $\pi = \{H_i\}_{i \in I}$ an admissible partition on Q, construct a pair (Y, S) where $Y = \pi$, the set of π-blocks. Now S acts on Y with respect to the operation $*$ defined by:

$$\left.\begin{array}{l} H_i * s = H_j \Leftrightarrow H_i s \subseteq H_j \\ H_i * s = \emptyset \Leftrightarrow H_i s = \emptyset \end{array}\right\} (i, j \in I, s \in S).$$

Clearly $(H_i * s) * s' = H_i * (ss')$ but it may be that $H_i * s = H_i * s^*$ for all $H_i \in Y$ and yet $s \neq s'$. To make (Y, S) into a transformation semigroup it is necessary that we remove this possibility. The usual procedure is to define a relation $\sim$, this time on the semigroup S.

Put $s \sim s' \Leftrightarrow H_i * s = H_i * s'$, $i \in I$, where $s, s' \in S$. This relation is clearly a congruence on S and if we form the quotient semigroup $S' = S/\sim$ we now obtain a transformation semigroup

$$\mathscr{A}/\langle\pi\rangle = (Y, S')$$

with the operation $*$ defined by

$$H_i * [s] = H_i * s$$

where $[s]$ denotes the $\sim$-class containing s $(H_i \in Y, s \in S)$.

Some remarks concerning the relationships between these two concepts of quotients are worth making.

First of all consider the state machine $\mathscr{M} = (Q, \Sigma, F)$ and its transformation semigroup $\mathbf{TS}(\mathscr{M})$. A partition π on Q is admissible with respect to $\mathscr{M}$ if and only if it is admissible with respect to $\mathbf{TS}(\mathscr{M})$.

Secondly the transformation semigroup of $\mathscr{M}/\pi$, $\mathbf{TS}(\mathscr{M}/\pi)$, is isomorphic to

$$(\mathbf{TS}(\mathscr{M}))/\langle\pi\rangle.$$

There are natural epimorphisms defined by quotient state machines and quotient transformation semigroups.

If $\mathscr{M} = (Q, \Sigma, F)$ is a state machine and $\pi = \{H_i\}_{i \in I}$ is an admissible partition on $\mathscr{M}$ then the epimorphism $(\alpha^\pi, 1_\Sigma) : \mathscr{M} \to \mathscr{M}/\pi$ defined by $\alpha^\pi(q) = H_i \Leftrightarrow q \in H_i$ $(q \in Q, H_i \in \pi)$, is called the *natural epimorphism defined by* π. If $\mathscr{A} = (Q, S)$ is a transformation semigroup and $\pi = \{H_i\}$ is an admissible partition on $\mathscr{A}$ then the epimorphism $(f^\pi, g^\pi) : \mathscr{A} \to \mathscr{A}/\pi$ defined by

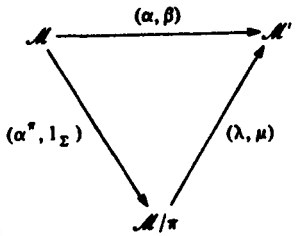
$$f^\pi(q) = H_i \Leftrightarrow q \in H_i \quad (q \in Q, H_i \in \pi)$$
$$g^\pi(s) = [s] \qquad\qquad (s \in S)$$

is called the *natural epimorphism defined by* π.

Suppose that $\mathscr{M} = (Q, \Sigma, F)$ is a state machine, and let $\pi = \{H_i\}_{i \in I}$, $\pi' = \{K_j\}_{j \in J}$ be admissible partitions on $\mathscr{M}$. If $\pi \leq \pi'$, that is, if given $i \in I$ there exists $j \in J$ with $H_i \subseteq K_j$, we can construct an epimorphism $(\alpha, 1_\Sigma) : \mathscr{M}/\pi \to \mathscr{M}/\pi'$ by $\alpha(H_i) = K_j$. This leads us to a homomorphism theorem for state machines.

Theorem 2.3.4

Let $\mathscr{M} = (Q, \Sigma, F)$ and $\mathscr{M}' = (Q', \Sigma', F')$ be state machines and $(\alpha, \beta) : \mathscr{M} \to \mathscr{M}'$ an epimorphism. Suppose that π_α is the admissible partition defined by α on $\mathscr{M}$ (so π_α is the partition of the relation R_α defined by α) and that π is an admissible partition on $\mathscr{M}$ satisfying the condition $\pi \leq \pi_\alpha$ then there exists an epimorphism $(\lambda, \mu) : \mathscr{M}/\pi \to \mathscr{M}'$ such that the following diagram of homomorphisms is commutative:

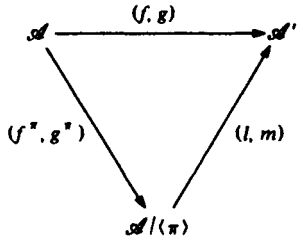


Furthermore if $\pi = \pi_\alpha$ then (λ, μ) is an isomorphism.

Proof Let $\pi = \{H_i\}_{i\in I}$, $\pi_\alpha = \{K_j\}_{j\in J}$. We define $\lambda : \pi \to Q'$ by $\lambda(H_i) = \alpha(q)$ where $q \in H_i$ $(i \in I)$. This is well-defined for if $q_1 \in H_i$ then $q, q_1 \in H_i \subseteq K_j$ for some $j \in J$ and so $\alpha(q) = \alpha(q_1)$. If we define $\mu : \Sigma \to \Sigma'$ by putting $\mu = \beta$ the result then follows easily. □

Theorem 2.3.5

Let $\mathscr{A} = (Q, S)$ and $\mathscr{A}' = (Q', S')$ be transformation semigroups and $(f, g) : \mathscr{A} \to \mathscr{A}'$ an epimorphism. Suppose that π_f is the admissible partition defined on $\mathscr{A}$ by f and that π is an admissible partition on $\mathscr{A}$ satisfying the condition $\pi \leq \pi_f$ then there exists an epimorphism $(l, m) : \mathscr{A}/\langle\pi\rangle \to \mathscr{A}'$ such that the following diagram of homomorphisms is commutative,



Furthermore if $\pi = \pi_f$ then (l, m) is an isomorphism.

Proof See exercise 2.2.

There are many other results concerned with homomorphisms and quotients of both state machines and transformation semigroups. While these are of independent algebraic interest they have not yet proved particularly useful in the study of automata and related areas. In fact the algebraic theory of machines diverges from the direction taken in other algebraic theories in one important respect. The idea of isomorphism is crucially important in many algebraic theories and many important classification theorems involve the establishment of isomorphisms between particular algebraic objects: an example would be the Wedderburn–Artin theorem for semi-simple Artinian associative rings which are shown to be *isomorphic* to a direct sum of matrix rings over various division rings. The emphasis in automata theory is, however, not what machines 'look like' but what 'they can do'. We will regard two machines as being very closely related if they can both 'do the same thing', they may however not be algebraically isomorphic!

2.4 Coverings

Before we can talk about two state machines doing the same thing we must first examine what the function of a state machine actually is. Let $\mathscr{M} = (Q, \Sigma, F)$ be a complete state machine and choose any $q \in Q$. Each word $\alpha \in \Sigma^*$ defines a partial function $F_\alpha : Q \to Q$ given by

$$qF_\alpha = F(q, \alpha) \text{ for all } q \in Q.$$

Therefore $\mathscr{M}$ is just a collection of partial functions $\{F_\alpha \mid \alpha \in \Sigma^*\}$. Now suppose that $\mathscr{M}' = (Q', \Sigma, F')$ is another state machine that 'performs the same function' as $\mathscr{M}$. Each state in $\mathscr{M}$ must correspond to a state in $\mathscr{M}'$ in such a way that the image under F_α in $\mathscr{M}$ corresponds to the image under F'_α in $\mathscr{M}'$ for each $\alpha \in \Sigma^*$. Formally we require a surjective partial function $\eta : Q' \to Q$, called a *covering*, such that $\eta(q')F_\alpha = \eta(q'F'_\alpha)$ for all $\alpha \in \Sigma^*$ and all q' belonging to the domain of η. To tidy up the notation and also to extend the notion to incomplete state machines we will write $\eta(q') = \varnothing$ if q' does not belong to the domain of η and also $qF_\alpha = \varnothing$ if $F(q, \alpha)$ is undefined. We may then define the covering requirement as

$$\eta(q')F_\alpha \subseteq \eta(q'F'_\alpha),$$

so that if q' is not in the domain of η we have

$$\varnothing \subseteq \eta(q'F'_\alpha),$$

similarly if

$$F(\eta(q'), \alpha)$$

is undefined then again

$$\varnothing \subseteq \eta(q'F'_\alpha).$$

However, if for some reason $\eta(q'F'_\alpha) = \varnothing$, then unless $\eta(q')F_\alpha = \varnothing$ also, the partial function η will not be a covering.

In general the input alphabets of $\mathscr{M}$ and $\mathscr{M}'$ may not be the same and so we must extend our covering concept to include this case.

Let $\mathscr{M} = (Q, \Sigma, F)$, $\mathscr{M}' = (Q', \Sigma', F')$ be state machines. If $\xi : \Sigma \to \Sigma'$ is a function and $\eta : Q' \to Q$ is a surjective partial function such that

$$\eta(q')F_\alpha \subseteq \eta(q'F'_{\xi(\alpha)})$$

for each $q' \in Q'$ and $\alpha \in \Sigma^*$, we say that (η, ξ) is a *covering of $\mathscr{M}$ by $\mathscr{M}'$*, written $\mathscr{M} \leq \mathscr{M}'$.

Examples 2.4

(i) Let $\mathscr{M} = (Q, \Sigma, F)$ be a state machine, define a relation $\sim$ on Σ by

$$\sigma \sim \sigma_1 \Leftrightarrow F_\sigma = F_{\sigma_1} \quad \text{for } \sigma, \sigma_1 \in \Sigma.$$

Construct a state machine $\mathcal{M}' = (Q, \Sigma/\sim, \bar{F})$ by defining $\bar{F}(q, [\sigma]) = F(q, \sigma)$ for $q \in Q$ and $[\sigma] \in \Sigma/\sim$. Now form $\xi : \Sigma \to \Sigma/\sim$ by putting

$$\xi(\sigma) = [\sigma] \quad \text{for } \sigma \in \Sigma$$

and

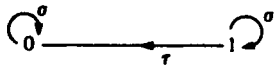
$$\eta : Q \to Q$$

by

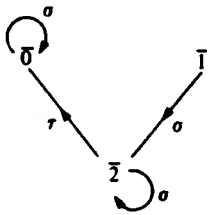
$$\eta(q) = q \quad \text{for } q \in Q$$

and we will obtain a covering (η, ξ) of $\mathcal{M}$ by $\mathcal{M}'$. We say that $\mathcal{M}'$ has been constructed from $\mathcal{M}$ by 'coinciding equal inputs'.

(ii) Let $\mathcal{M}$ be defined by the diagram



and $\mathcal{M}'$ by



Defining $\eta : \bar{0} \to 0, \bar{1} \to 1$ does not yield a covering $(\eta, 1_\Sigma)$ since

$$\eta(\bar{1})F_\tau = 0 \not\subseteq \eta(\bar{1}F'_\tau) = \varnothing.$$

However by putting $\eta' : \bar{0} \to 0, \bar{2} \to 1$ we may check that $(\eta', 1_\Sigma)$ gives a covering $\mathcal{M} \leq \mathcal{M}'$.

(iii) Let $\mathcal{M} = (Q, \Sigma, F)$, $\mathcal{M}_1 = (Q_1, \Sigma, F_1)$ and $\mathcal{M}' = (Q', \Sigma, F')$ be state machines. Suppose that $(f, 1_\Sigma) : \mathcal{M}' \to \mathcal{M}$ and $(f_1, 1_\Sigma) : \mathcal{M}' \to \mathcal{M}_1$ are homomorphisms with f an injective function and f_1 surjective. Construct a partial mapping $\eta : Q \to Q_1$ by $\eta(q) = f_1(f^{-1}(q))$ for $q \in f(Q')$. For $q \in f(Q')$, $\alpha \in \Sigma^*$,

$$\begin{aligned}
\eta(q)F_{1\alpha} &= [f_1(f^{-1}(q))]F_{1\alpha} \\
&= f_1((f^{-1}(q))F'_\alpha) \\
&= f_1((f^{-1}f(q'))F'_\alpha) \quad \text{if } q = f(q'), q' \in Q' \\
&= f_1(q'F'_\alpha) \\
&= f_1(f^{-1}f(q'F'_\alpha)) \\
&= f_1 f^{-1}(f(q')F_\alpha) \\
&= \eta(qF_\alpha)
\end{aligned}$$

For $q \in Q \backslash f(Q')$, $\eta(q)F_{1\alpha} = \varnothing \subseteq \eta(qF_\alpha)$. Thus $\mathcal{M}_1 \leq \mathcal{M}$.

The concept of covering also has an important role to play in transformation semigroups.

Let $\mathcal{A} = (Q, S)$, $\mathcal{B} = (P, T)$ be transformation semigroups and suppose that $\eta : P \to Q$ is a surjective partial function and that for each $s \in S$ there exists a $t_s \in T$ such that

$$\eta(p) \cdot s \subseteq \eta(p \cdot t_s) \quad \text{for } p \in P. \tag{*}$$

We say that $\mathcal{B}$ *covers* $\mathcal{A}$, written $\mathcal{A} \leq \mathcal{B}$ and that η is *a covering of* $\mathcal{A}$ *by* $\mathcal{B}$. Furthermore we will say that t_s is a *covering element* for s.

If $s, s' \in S$ then

$$\eta(p) \cdot ss' = (\eta(p) \cdot s) \cdot s' \subseteq \eta(p \cdot t_s) \cdot s' \subseteq \eta(p \cdot t_s \cdot t_{s'})$$

and so by defining $t_{ss'} = t_s \cdot t_{s'}$ the relationship (*) is satisfied.

This is a slightly more general concept for transformation semigroups than might seem necessary from the analogy with state machines. We could have asked for a semigroup homomorphism $\xi : S \to T$ such that $\eta(p) \cdot s \subseteq \eta(p \cdot \xi(s))$ for $p \in P$, however this possibility for a definition of covering is too restrictive for our purposes. See exercise 2.33. If we define $\xi(s)$ to be some element $t_s \in T$ that covers s we will have to show that $\xi(s \cdot s') = \xi(s) \cdot \xi(s')$, for $s, s' \in S$ and the element chosen as $\xi(s \cdot s')$ may differ from $\xi(s) \cdot \xi(s')$.

Theorem 2.4.1

Let $\mathcal{M}, \mathcal{M}'$ be state machines such that $\mathcal{M} \leq \mathcal{M}'$, then

$$\mathbf{TS}(\mathcal{M}) \leq \mathbf{TS}(\mathcal{M}').$$

Proof Let $\mathcal{M} = (Q, \Sigma, F)$, $\mathcal{M}' = (Q', \Sigma', F')$, let $\eta : Q' \to Q$ be a surjective partial covering function and $\xi : \Sigma \to \Sigma'$ a function, then

$$\eta(q')F_\alpha \subseteq \eta(q'F'_{\xi(\alpha)})$$

for $q' \in Q'$, $\alpha \in \Sigma^*$.

Suppose that $\mathbf{TS}(\mathcal{M}) = (Q, S)$ and $\mathbf{TS}(\mathcal{M}') = (Q', S')$. Let $s \in S$, then $a \in \Sigma^*$ such that $s = [a]$. Put $t_s = [\xi(a)] \in S'$. Now if $q' \in Q'$, $\eta(q') \cdot s = \eta(q') \cdot F_a \subseteq \eta(q'F'_{\xi(a)}) = \eta(q't_s)$ and so η defines a covering

$$\mathbf{TS}(\mathcal{M}) \leq \mathbf{TS}(\mathcal{M}'). \qquad \square$$

Recall the definition of the transformation semigroup of a finite semigroup S, it is the pair (S, S). Suppose that T is also a semigroup and form the transformation semigroup (T, T); if (S, S) is covered by (T, T) what can be said about the relationship between S and T?

Theorem 2.4.2

Let S, T be finite semigroups, then $(\mathcal{S}, S) \leq (\mathcal{T}, T)$ if and only if there exists a subsemigroup $\bar{T}$ of T such that S is a homomorphic image of $\bar{T}$.

Proof Let $(\mathcal{S}, S) \leq (\mathcal{T}, T)$, then there exists a surjective partial function $\eta : \mathcal{T} \to \mathcal{S}$, and for each $s \in S$ there exists $t_s \in T$ such that

$$\eta(y) \cdot s \subseteq \eta(y \cdot t_s) \quad \text{for all } y \in \mathcal{T}.$$

If η is a surjective partial function there exists a right inverse $\eta^{-1} : \mathcal{S} \to \mathcal{T}$ defined by choosing a y such that $\eta(y) = x$ and putting $\eta^{-1}(x) = y$. Then $\eta(\eta^{-1}(x)) = x$ for each $x \in \mathcal{S}$. Now $\eta(\eta^{-1}(x)) \cdot s \subseteq \eta(\eta^{-1}(x) \cdot t_s)$ and so

$$x \cdot s \subseteq \eta(\eta^{-1}(x) \cdot t_s).$$

However, $x \cdot s \neq \emptyset$ in this case, thus $x \cdot s = \eta(\eta^{-1}(x) \cdot t_s)$ for $x \in \mathcal{S}$. Now suppose that there exists $s' \in S$ such that

$$\eta(y) \cdot s' \subseteq \eta(y \cdot t_s) \quad \text{for all } y \in \mathcal{T},$$

so that when $x \in \mathcal{S}$,

$$x \cdot s' = \eta(\eta^{-1}(x) \cdot t_s) = x \cdot s.$$

However this implies that $s' = s$ because of the faithfulness of the semigroup action. We may now define a partial function $f : T \to S$ by

$$f(t) = s \Leftrightarrow t = t_s, \ t \in T.$$

Suppose that $\bar{T}$ is the domain of f, and $t_1, t_2 \in \bar{T}$, then $t_1 = t_{s_1}$, $t_2 = t_{s_2}$ for $s_1, s_2 \in S$. Since $t_{s_1} \cdot t_{s_2} = t_{s_1 \cdot s_2} = t_1 \cdot t_2$ we see that $t_1 \cdot t_2 \in \bar{T}$ and so $\bar{T}$ is a subsemigroup of T. Finally, $f(t_1 \cdot t_2) = f(t_{s_1 \cdot s_2}) = s_1 \cdot s_2 = f(t_1) \cdot f(t_2)$ and so f restricted to $\bar{T}$ is a *semigroup* homomorphism onto S.

Conversely let $g : \bar{T} \to S$ be a *semigroup* homomorphism from a subsemigroup $\bar{T}$ of T onto S. Consider the partial function $\dot{g} : \mathcal{T} \to \mathcal{S}$ defined by

$$\dot{g}(t) = g(t) \quad \text{if } t \in \bar{T}$$
$$\dot{g}(1) = 1 \quad \text{if } 1 \in \mathcal{T} \setminus T.$$

Then $\dot{g}$ is a surjective partial function from $\mathcal{T}$ onto $\mathcal{S}$. Let $s \in S$, there exists a $t \in T$ such that $s = g(t)$ and we write $t_s = t$.

Now for $y \in \mathcal{T}$, $\dot{g}(y) \cdot s = g(y) \cdot s = g(y) \cdot g(t) = g(y \cdot t) = \dot{g}(y \cdot t_s)$ if $y \in T$, and $\dot{g}(1) \cdot s = 1 \cdot s = g(t) = g(1 \cdot t_s) = \dot{g}(1 \cdot t_s)$ and so $\dot{g}$ is a covering map and

$$(\mathcal{S}, S) \leq (\mathcal{T}, T). \qquad \square$$

We say that S *divides* T in this situation and write $S | T$.

2.5 Mealy machines

So far we have examined state machines without any formal output, and we will now digress for a short while to look at machines with outputs. The reason for this is to motivate the next section on state machine products. Throughout this section all state machines are assumed to be complete.

Let $\mathcal{M} = (Q, \Sigma, F)$ be a state machine, and suppose that Θ is a non-empty finite set and $G : Q \times \Sigma \to \Theta$ is a function. The quintuple $\hat{\mathcal{M}} = (Q, \Sigma, \Theta, F, G)$ will be called a *Mealy machine* (after G. Mealy, 1955), Σ is the input alphabet, F the state transition function, Θ the output alphabet and G the output function. The machine works as follows.

Suppose that the input word $\sigma \in \Sigma$ is applied to the machine in state q, the machine then moves to state qF_σ and produces an output $G(q, \sigma) \in \Theta$ at the same instant. We will have, for each $\sigma \in \Sigma$, a mapping

$$G_\sigma : Q \to \Theta \quad \text{defined by} \quad qG_\sigma = G(q, \sigma), \ q \in Q.$$

To see what the machine does when we apply an input word $\alpha = \sigma_1 \ldots \sigma_k \in \Sigma^*$ to the machine in state q it is best to imagine the symbols printed on a tape and treat the machine as a black box that changes state and at the same time prints symbols from Θ on an output tape. The input tape will be fed into the machine on the right hand side and will move from right to left. The output tape also moves from right to left. See figure 2.1.

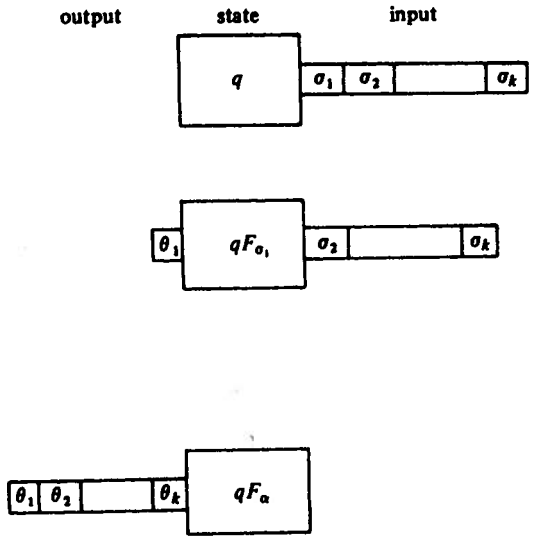


Figure 2.1. The action of a Mealy machine.

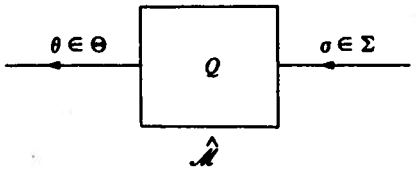
The final state will be qF_α and the output word is $\beta = \theta_1\theta_2 \dots \theta_k \in \Theta^*$ where

$$\theta_1 = qG_{\sigma_1},\ \theta_2 = qF_{\sigma_1}G_{\sigma_2},\ \dots$$
$$\theta_k = qF_{\sigma_1} \dots F_{\sigma_{k-1}}G_{\sigma_k}.$$

We will study the theory of Mealy machines in more detail later, it is sufficient to remark that a suitable notion of covering of Mealy machines can be formulated and this concept is closely related to the covering of the underlying state machines, for inside every Mealy machine there is a state machine.

A Mealy machine is just a set of translators, one for each internal state, which translates words of length k in Σ^* into words of length k in Θ^*, in fact each translator is nothing more than a rather special semigroup homomorphism.

These Mealy machines have been introduced here for the sole purpose of examining how they may be connected together to produce other Mealy machines. Each machine will be regarded as a black box with an input channel and an output channel.



There are two major methods of connecting up two Mealy machines, by parallel and by series.

Parallel connections

Suppose that $\hat{\mathcal{M}} = (Q, \Sigma, \Theta, F, G)$ and $\hat{\mathcal{M}}' = (Q', \Sigma, \Theta', F', G')$ are Mealy machines with the same input set Σ. Connecting them up in parallel as in figure 2.2 will produce a new Mealy machine:

$$\hat{\mathcal{M}} \wedge \hat{\mathcal{M}}' = (Q \times Q', \Sigma, \Theta \times \Theta', F \wedge F', G \wedge G')$$

where

$$(F \wedge F')((q, q'), \sigma) = (F(q, \sigma), F'(q', \sigma)),$$
$$(G \wedge G')((q, q'), \sigma) = (G(q, \sigma), G'(q', \sigma))$$

for each $\sigma \in \Sigma$, $(q, q') \in Q \times Q'$.

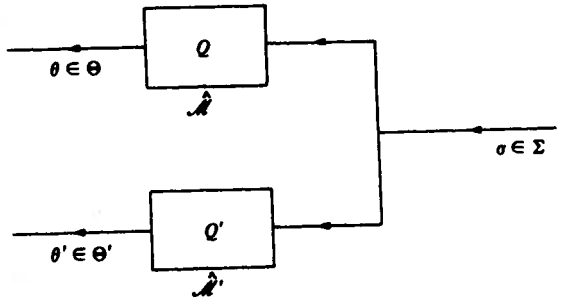


Figure 2.2. A restricted parallel connection.

We call this machine the *restricted direct product* of $\hat{\mathcal{M}}$ and $\hat{\mathcal{M}}'$ and it clearly produces words in $(\Theta \times \Theta')^*$ as outputs in response to input words from Σ^*.

Another type of parallel connection can be made, even when the input alphabets are different.

Let $\hat{\mathcal{M}} = (Q, \Sigma, \Theta, F, G)$, $\hat{\mathcal{M}}' = (Q', \Sigma', \Theta', F', G')$ be Mealy machines and define

$$\hat{\mathcal{M}} \times \hat{\mathcal{M}}' = (Q \times Q', \Sigma \times \Sigma', \Theta \times \Theta', F \times F', G \times G')$$

where

$$(F \times F')((q, q'), (\sigma, \sigma')) = (F(q, \sigma), F'(q', \sigma'))$$
$$(G \times G')((q, q'), (\sigma, \sigma')) = (G(q, \sigma), G'(q', \sigma'))$$

for each $(\sigma, \sigma') \in \Sigma \times \Sigma'$, $(q, q') \in Q \times Q'$.

This Mealy machine is called the (full) *direct product* of $\hat{\mathcal{M}}$ and $\hat{\mathcal{M}}'$. It converts words from $(\Sigma \times \Sigma')^*$ into words from $(\Theta \times \Theta')^*$. See figure 2.3. Note that each input word $(\sigma_1, \sigma_1')(\sigma_2, \sigma_2') \dots (\sigma_k, \sigma_k') \in (\Sigma \times \Sigma')^*$ can be written as $(\sigma_1\sigma_2 \dots \sigma_k, \sigma_1'\sigma_2' \dots \sigma_k') \in \Sigma^* \times (\Sigma')^*$ but not any word

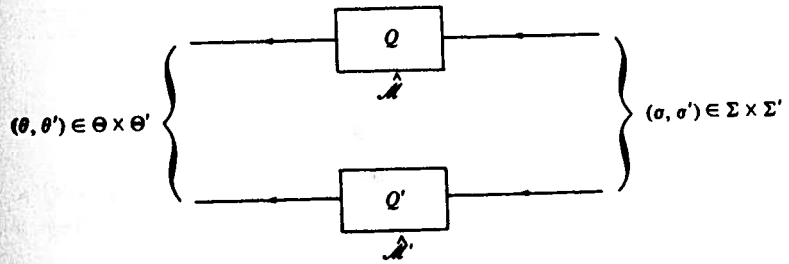


Figure 2.3. A full parallel connection.

from $\Sigma^* \times (\Sigma')^*$ can be used as an input unless it is of the form (α, α') where the length of α equals the length of α'.

This can be generalized in the following way. Consider figure 2.4, where $\lambda : \bar{\Sigma} \to \Sigma \times \Sigma'$ represents a mapping.

The machine is $(Q \times Q', \bar{\Sigma}, \Theta \times \Theta', F^\lambda, G^\lambda)$ where

$$F^\lambda((q, q'), \bar{\sigma}) = (F(q, p_1\lambda(\bar{\sigma})), F'(q', p_2\lambda(\bar{\sigma})))$$

$$G^\lambda((q, q'), \bar{\sigma}) = (G(q, p_1\lambda(\bar{\sigma})), G'(q', p_2\lambda(\bar{\sigma})))$$

for $(q, q') \in Q \times Q'$, $\bar{\sigma} \in \bar{\Sigma}$ and p_1, p_2 are the projection mappings associated with $\Sigma \times \Sigma'$. This machine generalizes both forms of the direct product and will be denoted by $\hat{\mathcal{M}} * \hat{\mathcal{M}}'$ and called the *general direct product.*

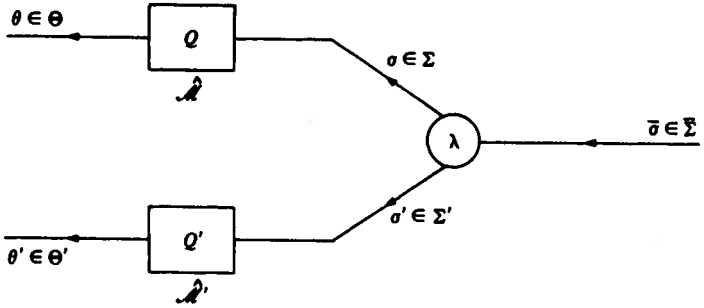


Figure 2.4. A general parallel connection.

Series connections

If we wish to connect two Mealy machines up in series we must ensure that we can 'hook up' the output from the first machine to the input of the second. See figure 2.5. One way of doing this is to define a function $\lambda : \Theta' \to \Sigma$ and so convert each output word $\beta' \in (\Theta')^*$ into an input word $\lambda(\beta') \in \Sigma^*$ before applying it to the machine $\hat{\mathcal{M}}$.

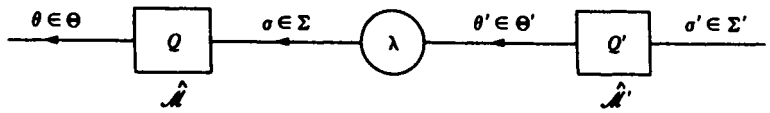


Figure 2.5. A cascade connection.

Once such a mapping λ is specified we can define a mapping $\omega : Q' \times \Sigma' \to \Sigma$ by $\omega(q', \sigma') = \lambda(G'(q', \sigma'))$. Then each input $\sigma' \in \Sigma'$ defines a mapping

$$\omega_{\sigma'} : Q' \to \Sigma$$

by

$$\omega_{\sigma'}(q') = \omega(q', \sigma')$$

for $q' \in Q'$.

The Mealy machine we have formed is:

$$\hat{\mathcal{M}}\omega\hat{\mathcal{M}}' = (Q \times Q', \Sigma', \Theta, F^\omega, G^\omega)$$

where

$$F^\omega((q, q'), \sigma') = (F(q, \omega_{\sigma'}(q')), F'(q', \sigma'))$$

$$G^\omega((q, q'), \sigma') = G(q, \omega_{\sigma'}(q'))$$

for $\sigma' \in \Sigma'$, $(q, q') \in Q \times Q'$.

Such a machine is called the *cascade product* of $\hat{\mathcal{M}}$ and $\hat{\mathcal{M}}'$ *induced* by ω.

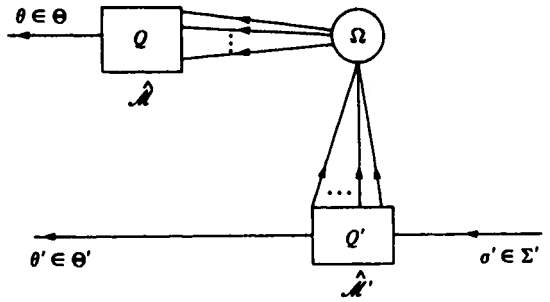


Figure 2.6. An alternative interpretation of a cascade connection.

Since ω defines a set of mappings $\Omega = \{\omega_{\sigma'} : Q' \to \Sigma\}_{\sigma' \in \Sigma'} \subseteq \Sigma^{Q'}$ we can visualize the connections as depicted in figure 2.6. This is but a short step from the useful generalization of figure 2.7 where $\Omega = \Sigma^{Q'}$. The

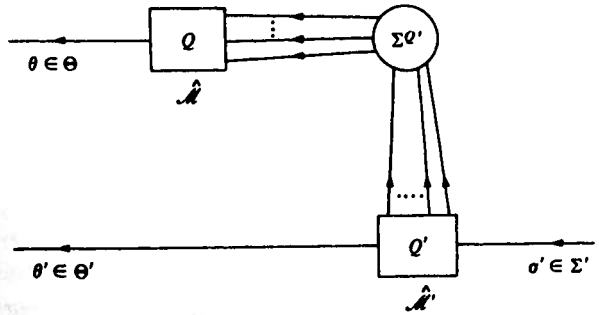


Figure 2.7. A wreath connection.

wreath product of $\hat{\mathcal{M}}$ and $\hat{\mathcal{M}}'$ is

$$\hat{\mathcal{M}} \circ \hat{\mathcal{M}}' = (Q \times Q', \Sigma^{Q'} \times \Sigma', \Theta \times \Theta', F^\circ, G^\circ)$$

where

$$F^\circ((q, q'), (f, \sigma')) = (F(q, f(q')), F'(q', \sigma'))$$

$$G^\circ((q, q'), (f, \sigma')) = (G(q, f(q')), G'(q', \sigma'))$$

for $\sigma' \in \Sigma'$, $f \in \Sigma^{Q'}$, $(q, q') \in Q \times Q'$.

There are further types of connection, most notably the feedback connections, but we will not be requiring them here.

The various products of Mealy machines show us how to define products of state machines; we merely remove the output sets and functions.

Let $\mathcal{M} = (Q, \Sigma, F)$, $\mathcal{M}' = (Q', \Sigma', F')$ be state machines. Define their *restricted direct product*:

$$\mathcal{M} \wedge \mathcal{M}' = (Q \times Q', \Sigma, F \wedge F'),$$

in the special case where $\Sigma = \Sigma'$ only, by:

$$(F \wedge F')((q, q'), \sigma) = (F(q, \sigma), F'(q', \sigma))$$

for $\sigma \in \Sigma$, $(q, q') \in Q \times Q'$.

Let $\mathcal{M} \times \mathcal{M}' = (Q \times Q', \Sigma \times \Sigma', F \times F')$ be the (full) *direct product* of $\mathcal{M}$ and $\mathcal{M}'$ where

$$(F \times F')((q, q'), (\sigma, \sigma')) = (F(q, \sigma), F'(q', \sigma'))$$

for $\sigma \in \Sigma$, $\sigma' \in \Sigma'$, $(q, q') \in Q \times Q'$.

Define the *cascade product* of $\mathcal{M}$ and $\mathcal{M}'$ with *respect to* $\omega : Q' \times \Sigma' \to \Sigma$ by

$$\mathcal{M} \omega \mathcal{M}' = (Q \times Q', \Sigma', F^\omega)$$

where $F^\omega((q, q'), \sigma') = (F(q, \omega(q', \sigma')), F'(q', \sigma'))$, for $\sigma' \in \Sigma'$, $(q, q') \in Q \times Q'$.

Finally we consider the *wreath product*, $\mathcal{M} \circ \mathcal{M}'$, of $\mathcal{M}$ and $\mathcal{M}'$ where

$$\mathcal{M} \circ \mathcal{M}' = (Q \times Q', \Sigma^{Q'} \times \Sigma', F^\circ)$$

and

$$F^\circ((q, q'), (f, \sigma')) = (F(q, f(q')), F'(q', \sigma'))$$

for $\sigma' \in \Sigma'$, $f \in \Sigma^{Q'}$, $(q, q') \in Q \times Q'$.

2.6 Products of transformation semigroups

The most useful ways of forming products of transformation semigroups will emerge if we consider the transformation semigroup of a product of state machines and compare it with the transformation

semigroups of the original state machines. Again all state machines and transformation semigroups will be assumed to be complete in this section. We examine the restricted direct product $\mathcal{M} \wedge \mathcal{M}'$, where $\mathcal{M} = (Q, \Sigma, F)$, and $\mathcal{M}' = (Q', \Sigma, F')$, and find its transformation semigroup.

Let $\alpha \in \Sigma^+$, then α defines a class $[\alpha]_\wedge$ with respect to the state machine $\mathcal{M} \wedge \mathcal{M}'$. Now let $\beta \in \Sigma^+$, then $\beta \in [\alpha]_\wedge$ if and only if

$$(F \wedge F')((q, q'), \beta) = (F \wedge F')((q, q'), \alpha)$$

for all $(q, q') \in Q \times Q'$ i.e.

$$(qF_\beta, q'F'_\beta) = (qF_\alpha, q'F'_\alpha)$$

i.e.

$$qF_\beta = qF_\alpha \quad \text{for all } q \in Q$$

and

$$q'F'_\beta = q'F'_\alpha \quad \text{for all } q' \in Q'.$$

Thus $\beta \in [\alpha]_\wedge$ if and only if $\beta \in [\alpha] \cap [\alpha]'$ where $[\alpha]$ and $[\alpha]'$ are the equivalence classes containing α with respect to the state machines $\mathcal{M}$ and $\mathcal{M}'$ respectively. Therefore the semigroup of $\mathcal{M} \wedge \mathcal{M}'$ is isomorphic to the quotient semigroup

$$\Sigma^+ / \sim \cap \sim'$$

where $\sim$ and $\sim'$ are the equivalence relations defined by $\mathcal{M}$ and $\mathcal{M}'$ respectively.

It is clear that we will not be able to form a restricted direct product between two arbitrary transformation semigroups. If $\mathscr{A} = (Q, S)$, $\mathscr{A}' = (Q', S')$ are transformation semigroups and there exists a free semigroup Σ^+ with epimorphisms $\theta : \Sigma^+ \to S$, $\theta' : \Sigma^+ \to S'$ then we can form the transformation semigroup

$$\mathscr{A} \wedge \mathscr{A}' = (Q \times Q', T)$$

where $T = \Sigma^+ / (R_\theta \cap R_{\theta'})$ (here R_θ corresponds to $\sim$ and $R_{\theta'}$ to $\sim'$) and the action is given by:

$$(q, q')[\alpha]_\wedge = (q\theta(\alpha), q'\theta'(\alpha))$$

for $(q, q') \in Q \times Q'$ and $[\alpha]_\wedge \in T$. (It is an easy matter to check that T acts faithfully on $Q \times Q'$.)

The definition of $\mathscr{A} \wedge \mathscr{A}'$ will depend on the choice of θ and θ', so we call $\mathscr{A} \wedge \mathscr{A}'$ the *restricted direct product* of $\mathscr{A}$ and $\mathscr{A}'$ (with respect to θ and θ'). We can now state:

Theorem 2.6.1
Let $\mathcal{M} = (Q, \Sigma, F)$ and $\mathcal{M}' = (Q', \Sigma, F')$ then

$$\mathbf{TS}(\mathcal{M} \wedge \mathcal{M}') = \mathbf{TS}(\mathcal{M}) \wedge \mathbf{TS}(\mathcal{M}')$$

(for suitable epimorphisms $\theta : \Sigma^+ \to \mathbf{S}(\mathcal{M})$, $\theta' : \Sigma^+ \to \mathbf{S}(\mathcal{M}')$).

Turning our attention to the full direct product we immediately see that the situation is more straightforward. In chapter 1 the direct product of two semigroups was introduced. We can extend this concept easily to the (full) direct product of two transformation semigroups.

Let $\mathscr{A} = (Q, S)$, $\mathscr{A}' = (Q', S')$ be transformation semigroups, define the (full) *direct product*

$$\mathscr{A} \times \mathscr{A}' = (Q \times Q', S \times S')$$

where the action is given by:

$$(q, q')(s, s') = (qs, q's')$$

for $(q, q') \in Q \times Q'$, $(s, s') \in S \times S'$.

Clearly the action is faithful.

We write $\prod^r \mathscr{A}$ to denote $\mathscr{A} \times \mathscr{A} \times \ldots \times \mathscr{A}$ (r times).

Theorem 2.6.2
Let $\mathcal{M} = (Q, \Sigma, F)$, $\mathcal{M}' = (Q', \Sigma', F')$ be state machines.
$\mathbf{TS}(\mathcal{M} \times \mathcal{M}') \leq \mathbf{TS}(\mathcal{M}) \times \mathbf{TS}(\mathcal{M}')$.

Proof Now $\mathcal{M} \times \mathcal{M}' = (Q \times Q', \Sigma \times \Sigma', F \times F')$, and so $\mathbf{S}(\mathcal{M} \times \mathcal{M}')$ will be a quotient semigroup of the free semigroup $(\Sigma \times \Sigma')^+$. Let $(\alpha, \beta) \in (\Sigma \times \Sigma')^+$, then

$$(\alpha, \beta) = (\sigma_1, \sigma_1') \ldots (\sigma_n, \sigma_n')$$

for some $\sigma_i \in \Sigma$, $\sigma_i' \in \Sigma'$, $i = 1, \ldots, n$. The elements of $\mathbf{S}(\mathcal{M} \times \mathcal{M}')$ will be equivalence classes of the form $[(\alpha, \beta)]_\times$ where

$$(\alpha, \beta) \sim_\times (\alpha_1, \beta_1) \Leftrightarrow (F \times F')_{(\alpha, \beta)} = (F \times F')_{(\alpha_1, \beta_1)}$$
$$\Leftrightarrow F_\alpha = F_{\alpha_1} \text{ and } F'_\beta = F'_{\beta_1} \Leftrightarrow \alpha \sim \alpha_1 \text{ and } \beta \sim' \beta_1.$$

Define a function $g : \mathbf{S}(\mathcal{M} \times \mathcal{M}') \to \mathbf{S}(\mathcal{M}) \times \mathbf{S}(\mathcal{M}')$ by

$$g([(\alpha, \beta)]_\times) = ([\alpha], [\beta]')$$

for each $[(\alpha, \beta)]_\times \in \mathbf{S}(\mathcal{M} \times \mathcal{M}')$. It is a routine matter to establish that g is a semigroup monomorphism and finally $(1_{Q \times Q'}, g) : \mathbf{TS}(\mathcal{M} \times \mathcal{M}') \to \mathbf{TS}(\mathcal{M}) \times \mathbf{TS}(\mathcal{M})'$ is a transformation semigroup covering. □

Theorem 2.6.3
Let $\mathscr{A} = (Q, S)$, $\mathscr{A}' = (Q', S')$ be transformation semigroups, Σ a finite non-empty set and $\theta : \Sigma^+ \to S$, $\theta' : \Sigma^+ \to S'$ semigroup epimorphisms, then

$$\mathbf{TS}(\mathscr{A} \wedge \mathscr{A}') \leq \mathbf{TS}(\mathscr{A} \times \mathscr{A}').$$

Proof Using the notation of 2.6.1 we will take the identity map $1_{Q \times Q'}$ as the covering map. Now let $[\alpha]_\wedge \in T$, the semigroup of $\mathscr{A} \wedge \mathscr{A}'$, so that $\alpha \in \Sigma^+$. Consider $\theta(\alpha) \in S$ and $\theta'(\alpha) \in S'$ and define $(\theta(\alpha), \theta'(\alpha))$ to be a covering element for $[\alpha]_\wedge$. Then, for $(q, q') \in Q \times Q'$, we have

$$(q, q')[\alpha]_\wedge = (q\theta(\alpha), q'\theta'(\alpha))$$
$$= (q, q')(\theta(\alpha), \theta'(\alpha))$$

and so the covering exists. □

Our next topic is the examination of the cascade and wreath products and their implications for transformation semigroup theory. Suppose that $\mathcal{M} = (Q, \Sigma, F)$, $\mathcal{M}' = (Q', \Sigma', F')$ are state machines and $\omega : Q' \times \Sigma' \to \Sigma$ is a mapping. Let $\mathscr{A} = (Q, S)$, $\mathscr{A}' = (Q', S')$ be the transformation semigroups of $\mathcal{M}$ and $\mathcal{M}'$ respectively. If $\mathscr{B} = (Q \times Q', T)$ is the transformation semigroup of $\mathcal{M} \omega \mathcal{M}'$, we wish to find a relationship between $\mathscr{A}$, $\mathscr{A}'$ and $\mathscr{B}$. Unfortunately there is no simple straightforward construction that yields the transformation semigroup $\mathscr{B}$ from a suitable combination of $\mathscr{A}$ and $\mathscr{A}'$. What we will do here is to show that $\mathscr{B}$ can be covered by the wreath product of the transformation semigroups $\mathscr{A}$ and $\mathscr{A}'$. This will now be defined.

Let $\mathscr{A} = (Q, S)$, $\mathscr{A}' = (Q', S')$ be transformation semigroups. Define

$$\mathscr{A} \circ \mathscr{A}' = (Q \times Q', S^{Q'} \times S')$$

where $S^{Q'}$ is the set of all mappings from Q' to S. The set $S^{Q'} \times S'$ is a semigroup, for if $f : Q' \to S$, $f_1 : Q' \to S$, s', $s_1' \in S'$ then we may define a mapping $f * f_1 : Q' \to S$ by

$$f * f_1(q') = f(q') \cdot f_1(q's')$$

for all $q' \in Q'$ and put $(f, s) \cdot (f_1, s') = (f * f_1, ss')$. Then the action of $S^{Q'} \times S'$ on $Q \times Q'$ is defined by

$$(q, q')(f, s') = (q(f(q')), q's')$$

for $(q, q') \in Q \times Q'$, $(f, s') \in S^{Q'} \times S'$.

The faithfulness of this action is easily checked and thus $\mathscr{A} \circ \mathscr{A}'$ is a transformation semigroup, called the *wreath product* of $\mathscr{A}$ and $\mathscr{A}'$. If

$\mathcal{S}=(S, S)$ and $\mathcal{T}=(T, T)$ are transformation semigroups with S and T semigroups then $\mathcal{S}\circ\mathcal{T}=(S\times T, S\circ T)$ where $S\circ T$ is defined in section 1.3. We have the following result.

Theorem 2.6.4

Let $\mathcal{M}=(Q, \Sigma, F)$, $\mathcal{M}'=(Q', \Sigma', F')$ be state machines and $\omega: Q'\times\Sigma'\to\Sigma$ a mapping.

(i) $\mathbf{TS}(\mathcal{M}\omega\mathcal{M}')\leq\mathbf{TS}(\mathcal{M})\circ\mathbf{TS}(\mathcal{M}')$

(ii) $\mathbf{TS}(\mathcal{M}\circ\mathcal{M}')\leq\mathbf{TS}(\mathcal{M})\circ\mathbf{TS}(\mathcal{M}')$.

Proof (i) Let $\alpha'\in(\Sigma')^+$ and consider the element $[\alpha']_\omega$ defined by α' in the semigroup $\mathbf{S}(\mathcal{M}\omega\mathcal{M}')$. We must find an element of the semigroup $S^{Q'}\times S'$ which will cover this element $[\alpha']_\omega$ where $S=\mathbf{S}(\mathcal{M})$ and $S'=\mathbf{S}(\mathcal{M}')$. The word α' will clearly define an element $[\alpha']'$ of S' but we must also find a suitable mapping $f_{\alpha'}: Q'\to S$. First of all we have to examine the mapping $\omega: Q'\times\Sigma'\to\Sigma$. Each $\sigma'\in\Sigma'$ defines a mapping $\omega_{\sigma'}: Q'\to\Sigma$ by

$$\omega_{\sigma'}(q')=\omega(q', \sigma')$$

for all $q'\in Q'$. The mapping ω describes the link-up between the two machines in the cascade connection, so we must investigate what happens when we input a word from $(\Sigma')^+$ into the leading machine. Suppose that we apply a word of length 2 from $(\Sigma')^+$.

If $(q, q')\in Q\times Q'$, $\sigma', \sigma_1'\in\Sigma'$ then

$$\begin{aligned}(q, q')F^{\omega}_{\sigma'\sigma_1'} &= (qF_{\omega(q', \sigma')}, q'F'_{\sigma'})F^{\omega}_{\sigma_1'}\\ &= (qF_{\omega(q', \sigma')}F_{\omega(q'F'_{\sigma'}, \sigma_1')}, q'F'_{\sigma'}F'_{\sigma_1'})\\ &= (qF_{\omega(q', \sigma')\omega(q'F'_{\sigma'}, \sigma_1')}, q'F'_{\sigma'\sigma_1'}).\end{aligned}\qquad(*)$$

It makes sense to define a generalization of the map ω to cover the cases of words in $(\Sigma')^+$ of length greater than 1. To do this we will define $\omega^+: Q'\times(\Sigma')^+\to\Sigma^+$ in such a way that

$$(q, q')F^{\omega}_{\sigma'\sigma_1'}=(qF_{\omega^+(q', \sigma'\sigma_1')}, q'F'_{\sigma'\sigma_1'})$$

and so we need

$$\omega^+(q', \sigma'\sigma_1')=\omega(q', \sigma')\omega(q'F'_{\sigma'}, \sigma_1')$$

by analogy with (*). Generalizing further we define $\omega^+: Q'\times(\Sigma')^+\to\Sigma^+$ inductively by $\omega^+(q', \sigma'\alpha')=\omega(q', \sigma')\cdot\omega^+(q'F'_{\sigma'}, \alpha')$ if $\alpha'\in(\Sigma')^+$ and $\omega^+(q', \sigma')=\omega(q', \sigma')$ where $\sigma'\in\Sigma'$, $\alpha'\in(\Sigma')^+$, $q'\in Q'$. Then $(q, q')F^{\omega}_{\alpha'}=(qF_{\omega^+(q', \alpha')}, q'F'_{\alpha'})$ for $\alpha'\in(\Sigma')^+$. Now, for each $\alpha'\in(\Sigma')^+$ we have a mapping $\omega^+_{\alpha'}: Q'\to(\Sigma)^+$ defined by $\omega^+_{\alpha'}(q')=\omega^+(q', \alpha')$, $q'\in Q'$. Returning to our problem we can now define

$$f_{\alpha'}: Q'\to S \quad\text{by}\quad f_{\alpha'}(Q')=[\omega^+_{\alpha'}(q')]$$

for $q'\in Q'$. Then to each $[\alpha']_\omega$ we will associate the pair $(f_{\alpha'}, [\alpha']')$. The first thing to check is that this definition is well-defined. Suppose that $\beta'\in(\Sigma')^+$ and that $[\alpha']_\omega=[\beta']_\omega$, then for $(q, q')\in Q\times Q'$, $(q, q')F^{\omega}_{\alpha'}=(q, q')F^{\omega}_{\beta'}$ that is

$$qF_{\omega^+(q', \alpha')}=qF_{\omega^+(q', \beta')}$$

for each $q\in Q$ and

$$q'F'_{\alpha'}=q'F'_{\beta'}$$

for each $q'\in Q'$.

Therefore

$$\begin{aligned}f_{\beta'}(q') &= [\omega^+_{\beta'}(q')]\\ &= [\omega^+(q', \beta')]\\ &= [\omega^+(q', \alpha')]\\ &= f_{\alpha'}(q')\end{aligned}$$

for each $q'\in Q'$ and $[\alpha']'_\omega=[\beta']'_\omega$. Hence $[\alpha']_\omega=[\beta']_\omega$ implies $(f_{\alpha'}, [\alpha']')=(f_{\beta'}, [\beta']')$. Our final task is the verification of the covering relationship using the identity mapping on $Q\times Q'$. That is

$$(q, q')\cdot[\alpha']\subseteq(q, q')\cdot(f_{\alpha'}, [\alpha']')$$

or in other words,

$$\begin{aligned}(qF_{\omega^+(q', \alpha')}, q'F'_{\alpha'}) &\subseteq (q\cdot[\omega^+(q', \alpha')], q'\cdot[\alpha']')\\ &= (q\cdot f_{\alpha'}(q'), q'\cdot[\alpha']')\\ &= (q, q')\cdot(f_{\alpha'}, [\alpha']').\end{aligned}$$

The covering is thus established.

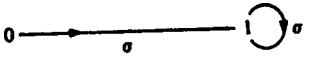
(ii) We will leave the proof of the covering $\mathbf{TS}(\mathcal{M}\circ\mathcal{M}')\leq\mathbf{TS}(\mathcal{M})\circ\mathbf{TS}(\mathcal{M}')$ as an exercise. □

The product $\mathcal{A}\circ\mathcal{A}\circ\ldots\circ\mathcal{A}$ with r factors is written $\mathcal{A}^r$.

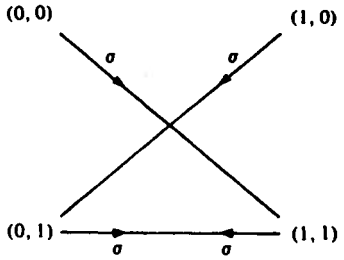
Example 2.5

Consider the state machine $\mathcal{M}$ defined by:

with transformation semigroup $\mathbf{Z}_2$ and the state machine $\mathcal{M}'$ defined by:

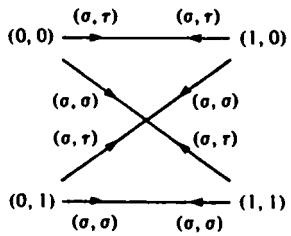


with transformation semigroup $\mathscr{C}$. We may form the restricted direct product $\mathscr{M} \wedge \mathscr{M}'$ which can be described by the diagram:



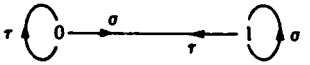
This has semigroup $\{\sigma, \sigma^2\}$ with the identity $\sigma^3 = \sigma$. The full direct product $\mathscr{M} \times \mathscr{M}'$ has the same description: in this case because the input alphabets to $\mathscr{M}$ and $\mathscr{M}'$ are both singletons. However if $\mathscr{M}''$ is given by

with transformation semigroup $\bar{\mathbf{2}}$ then $\mathscr{M} \wedge \mathscr{M}''$ cannot be defined and $\mathscr{M} \times \mathscr{M}''$ is given by

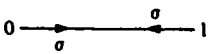


The semigroup of $\mathscr{M} \times \mathscr{M}''$ has four elements and $\mathbf{TS}(\mathscr{M} \times \mathscr{M}'') \cong \mathbf{Z}_2 \times \bar{\mathbf{2}}$.

Example 2.6
Let $\mathscr{M}$ be given by



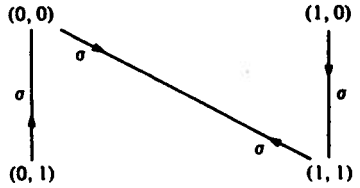
and $\mathscr{M}'$ be given by



Define a mapping $\omega : Q' \times \Sigma' \to \Sigma$ where $Q' = \{0, 1\}$, $\Sigma' = \{\sigma\}$, $\Sigma = \{\sigma, \tau\}$ by

$$\omega(0, \sigma) = \sigma, \quad \omega(1, \sigma) = \tau.$$

It is now possible to define the cascade product $\mathscr{M} \omega \mathscr{M}'$ which has diagram



and semigroup $\{\sigma, \sigma^2\}$ subject to the relation $\sigma^3 = \sigma$.

Finally we examine the wreath product of $\mathscr{M}$ and $\mathscr{M}'$, this has input alphabet $\Sigma^{Q'} \times \Sigma'$. Denote the four elements of $\Sigma^{Q'}$ by $\alpha, \beta, \gamma, \delta$ where

$$\alpha(0) = \alpha(1) = \sigma, \beta(0) = \sigma, \beta(1) = \tau,$$

$$\gamma(0) = \tau, \gamma(1) = \sigma, \delta(0) = \delta(1) = \tau.$$

Then the state machine $\mathscr{M} \circ \mathscr{M}'$ has the table

	(0, 0)	(1, 0)	(0, 1)	(1, 1)
(α, σ)	(1, 1)	(1, 1)	(1, 0)	(1, 0)
(β, σ)	(1, 1)	(1, 1)	(0, 0)	(0, 0)
(γ, σ)	(0, 1)	(0, 1)	(1, 0)	(1, 0)
(δ, σ)	(0, 1)	(0, 1)	(0, 0)	(0, 0)

The semigroup of this machine has eight elements and the transformation semigroup is isomorphic to the wreath product $\bar{\mathbf{2}} \circ \mathbf{Z}_2$.

However the state table of $\mathscr{M}' \circ \mathscr{M}$ is given by

	(0, 0)	(1, 0)	(0, 1)	(1, 1)
(α, σ)	(1, 1)	(0, 1)	(1, 1)	(0, 1)
(α, τ)	(1, 0)	(0, 0)	(1, 0)	(0, 0)

where $\alpha : Q \to \Sigma'$. This has a semigroup with four elements whereas the semigroup of $\mathbf{Z}_2 \circ \bar{\mathbf{2}}$ has eight elements. Hence the covering in 2.6.4(ii) cannot in general be replaced by an isomorphism.

Our final task in this section is to examine the associativity of the two main product constructions.

Theorem 2.6.5

Let (Q_i, S_i) be transformation semigroups for $i = 1, 2, 3$, then $(Q_1, S_1) \times ((Q_2, S_2) \times (Q_3, S_3)) \cong ((Q_1, S_1) \times (Q_2, S_2)) \times (Q_3, S_3)$.

Proof This is left to the reader. □

Before embarking on a similar result for the wreath product it is best to examine in detail what the wreath product $(Q_1, S_1) \circ ((Q_2, S_2) \circ (Q_3, S_3))$ looks like.

Now

$$(Q_2, S_2) \circ (Q_3, S_3) = (Q_2 \times Q_3, S_2^{Q_3} \times S_3)$$

and

$$(Q_1, S_1) \circ ((Q_2, S_2) \circ (Q_3, S_3)) = (Q_1 \times (Q_2 \times Q_3), S_1^{Q_2 \times Q_3} \times (S_2^{Q_3} \times S_3)).$$

Similarly

$$((Q_1, S_1) \circ (Q_2, S_2)) \circ (Q_3, S_3) = (Q_1 \times Q_2, S_1^{Q_2} \times S_2) \circ (Q_3, S_3)$$
$$= ((Q_1 \times Q_2) \times Q_3, (S_1^{Q_2} \times S_2)^{Q_3} \times S_3).$$

However $(S_1^{Q_2} \times S_2)^{Q_3}$ is the set of maps from Q_3 to $S_1^{Q_2} \times S_2$ and if $f: Q_3 \to S_1^{Q_2} \times S_2$ we can consider the mappings $f_1 = p_1 \circ f$, $f_2 = p_2 \circ f$ obtained by projecting f onto the factors. Then

$$f_1: Q_3 \to S_1^{Q_2} \quad \text{and} \quad f_2: Q_3 \to S_2.$$

Let $q_3 \in Q_3$, then $f_1(q_3) = g$ say, where $g: Q_2 \to S_1$. We can construct a mapping $\bar{f}_1: Q_2 \times Q_3 \to S_1$ by

$$\bar{f}_1(q_2, q_3) = g(q_2) = (f_1(q_3))(q_2).$$

It is a routine matter to check that the function

$$\Theta: (S_1^{Q_2} \times S_2)^{Q_3} \times S_3 \to S_1^{Q_2 \times Q_3} \times (S_2^{Q_3} \times S_3)$$

defined by $\Theta(f, s_3) = (\bar{f}_1, (f_2, s_3))$ for $f \in (S_1^{Q_2} \times S_2)^{Q_3}$, $s_3 \in S_3$, is an isomorphism of semigroups. Thus we may establish:

Theorem 2.6.6

If (Q_i, S_i) are transformation semigroups for $i = 1, 2, 3$ then

$$(Q_1, S_1) \circ ((Q_2, S_2) \circ (Q_3, S_3)) \cong ((Q_1, S_1) \circ (Q_2, S_2)) \circ (Q_3, S_3).$$

Associativity relations for the other products defined in this chapter will be examined in the exercises. As for the direct product and wreath product of any finite number of transformation semigroups, we shall usually rearrange brackets or remove them altogether when this serves to clarify the notation.

There are some 'distributive laws' connecting the direct product and the wreath product but we will postpone discussion of these until the next section.

2.7 More on products

We now extend our definitions of products of state machines and transformation semigroups to include the incomplete cases.

First let $\mathcal{M} = (Q, \Sigma, F)$ be a state machine and suppose that $P \subseteq Q$. Define the *restriction of $\mathcal{M}$ to P* to be the state machine

$$\mathcal{M}|_P = (P, \Sigma, F')$$

where

$$F': P \times \Sigma \to P$$

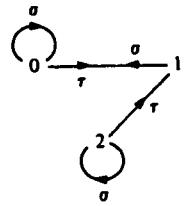
is defined by

$$F'(p, \sigma) = \begin{cases} F(p, \sigma) & \text{if } F(p, \sigma) \in P, \\ \varnothing & \text{otherwise.} \end{cases}$$

What this amounts to is that all the states in $Q \backslash P$ have been removed together with all the arrows leading to or from these states.

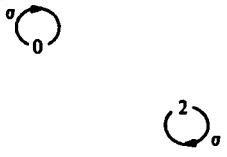
From example 2.1(vi) we have the state machine $\mathcal{M}$ given by

$\mathcal{M}$	0	1	2
F_σ	0	0	2
F_τ	1	∅	1



and if $P = \{0, 2\}$ then $\mathcal{M}|_P$ is given by:

$\mathcal{M}|_P$	0	2
F_σ	0	2
F_τ	∅	∅



Now suppose that $\mathscr{A} = (Q, S)$ is a transformation semigroup and $P \subseteq Q$. We define the restriction $\mathscr{A}|_P$ to be the transformation semigroup $\mathbf{TS}((\mathbf{SM}(\mathscr{A}))|_P)$. Now $\mathbf{SM}(\mathscr{A}) = (Q, S, F)$ where $F: Q \times S \to Q$ is defined by $qF_s = qs$ for all $q \in Q$, $s \in S$. The restriction $(\mathbf{SM}(\mathscr{A}))|_P$ is (P, S, F')

where $F' : P \times S \to P$ is defined by

$$pF'_s = \begin{cases} ps & \text{if } ps \in P \\ \varnothing & \text{otherwise} \end{cases}$$

Now $\mathbf{TS}((\mathbf{SM}(\mathscr{A}))|_P) = (P, T)$ where $T = \mathbf{S}((P, S, F')) = S^*/\sim$ and $\sim$ is defined by $\alpha \sim \beta \Leftrightarrow pF'_\alpha = pF'_\beta$ for all $p \in P$, and $\alpha, \beta \in S^*$. Note that if α is the word $s_1 \ldots s_n \in S^*$ then $[\alpha] = [s]$ where s is the product in S of $s_1 \ldots s_n$, and $[s]$ is the equivalence class containing s.

Thus

$$\mathscr{A}|_P = (P, S/\sim).$$

Another way of looking at $\mathscr{A}|_P$ is to consider the pair (P, S) with the operation $*$ defined by

$$p * s = \begin{cases} ps & \text{if } ps \in P \\ \varnothing & \text{otherwise.} \end{cases}$$

Under this operation S may not be faithful on P and so we have to define the relation ρ on S by

$$s \rho s' \Leftrightarrow p * s = p * s' \quad \text{for all } p \in P, \text{ where } s, s' \in S.$$

Thus $\mathscr{A}|_P$ may be regarded as the transformation semigroup $(P, S/\rho)$ under the operation induced by $*$, namely

$$p\langle s \rangle = p * s \quad \text{for } p \in P, \langle s \rangle \in S/\rho.$$

In our example $\mathscr{A}|_P = (P, \{\sigma, \theta\})$ where $\sigma^2 = \sigma = 1_P$.

Now let $\mathscr{M} = (Q, \Sigma, F)$, $\mathscr{M}' = (Q', \Sigma', F')$ be two state machines which are not necessarily complete. Define $\mathscr{M} \times \mathscr{M}' = (\mathscr{M}^c \times (\mathscr{M}')^c)|_{Q \times Q'}$. So we complete $\mathscr{M}$ and $\mathscr{M}'$ if necessary, form the direct product with the complete machines and then restrict the resultant to the product of the original state sets.

Similarly $\mathscr{M} \wedge \mathscr{M}' = (\mathscr{M}^c \wedge (\mathscr{M}')^c)|_{Q \times Q'}$ if $\Sigma = \Sigma'$, and $\mathscr{M} \circ \mathscr{M}' = (\mathscr{M}^c \circ (\mathscr{M}')^c)|_{Q \times Q'}$. Now let $\omega : Q' \times \Sigma' \to \Sigma$ be a function, and suppose that $\mathscr{M}'$ is incomplete. If $\bar{q}'$ is the new sink state for $\mathscr{M}'$ define

$$\omega^c : (Q' \cup \{\bar{q}'\}) \times \Sigma' \to \Sigma$$

by

$$\omega^c(q', \sigma') = \omega(q', \sigma')$$

for $q' \in Q'$, $\sigma' \in \Sigma'$

$$\omega^c(\bar{q}', \sigma') = \text{arbitrary}$$

for $\sigma' \in \Sigma'$. Now put $\mathscr{M}\omega\mathscr{M}' = (\mathscr{M}^c \omega^c (\mathscr{M}')^c)|_{Q \times Q'}$. We make similar definitions for transformation semigroups. If $\mathscr{A} = (Q, S)$, $\mathscr{A}' = (Q', S')$ then

$$\mathscr{A} \times \mathscr{A}' = (\mathscr{A}^c \times (\mathscr{A}')^c)|_{Q \times Q'}$$

and

$$\mathscr{A} \circ \mathscr{A}' = (\mathscr{A}^c \circ (\mathscr{A}')^c)|_{Q \times Q'}.$$

We now prove some straightfoward identities. As usual we will assume that all things are complete but the necessary adjustments for the incomplete cases should be regarded as exercises.

Theorem 2.7.1

Let $\mathscr{A} = (Q, S)$, $\mathscr{B} = (P, T)$ be transformation semigroups. Then

$$(\mathscr{A} \circ \mathscr{B})^\cdot \leq \mathscr{A}^\cdot \circ \mathscr{B}^\cdot.$$

Proof We will only consider the case where $\mathscr{A}$ and $\mathscr{B}$ are both *not* transformation monoids. Then $\mathscr{A}^\cdot = (Q, S \cup \{1_Q\})$ and $\mathscr{B}^\cdot = (P, T \cup \{1_P\})$. We have $(\mathscr{A} \circ \mathscr{B})^\cdot = (Q \times P, (S^P \times T) \cup \{1_{Q \times P}\})$. Define $1_{Q \times P}$ as the covering function, for $(f, t) \in S^P \times T$ we will use $(\bar{f}, t)$ where $\bar{f} : P \to S \cup \{1_Q\}$ is defined by

$$\bar{f}(p) = f(p)$$

for $p \in P$; and for $(1_Q, t)$ we will use (g, t) where

$$g : P \to S \cup \{1_Q\}$$

is defined by

$$g(p) = 1_Q$$

for $p \in P$.

Now

$$(q, p)(f, t) = (qf(p), pt) \leq (q, p)(\bar{f}, t)$$

and

$$(q, p)(1_Q, t) = (q, pt) \leq (q, p)(g, t)$$

and the covering is established. □

Theorem 2.7.2

Let $\mathscr{A}_1, \mathscr{A}_2, \mathscr{B}_1, \mathscr{B}_2$ be transformation semigroups, then

$$(\mathscr{A}_1 \circ \mathscr{A}_2) \times (\mathscr{B}_1 \circ \mathscr{B}_2) \leq (\mathscr{A}_1 \times \mathscr{B}_1) \circ (\mathscr{A}_2 \times \mathscr{B}_2).$$

Proof Let $\mathscr{A}_i = (Q_i, S_i)$, $\mathscr{B}_i = (P_i, T_i)$ for $i = 1, 2$.

$$(\mathscr{A}_1 \circ \mathscr{A}_2) \times (\mathscr{B}_1 \circ \mathscr{B}_2) = (Q_1 \times Q_2 \times P_1 \times P_2, S_1^{Q_2} \times S_2 \times T_1^{P_2} \times T_2)$$

$$(\mathscr{A}_1 \times \mathscr{B}_1) \circ (\mathscr{A}_2 \times \mathscr{B}_2) = (Q_1 \times P_1 \times Q_2 \times P_2, (S_1 \times T_1)^{Q_2 \times P_2} \times S_2 \times T_2).$$

Define $\phi : Q_1 \times P_1 \times Q_2 \times P_2 \to Q_1 \times Q_2 \times P_1 \times P_2$ by

$$\phi(q_1, p_1, q_2, p_2) = (q_1, q_2, p_1, p_2)$$

for $q_i \in Q_i$, $p_i \in P_i$, $i = 1, 2$. Then (f_1, s_2, g_1, t_2) where $f_1 : Q_2 \to S_1$, $g_1 : P_2 \to T_1$, $s_2 \in S_2$, $t_2 \in T_2$ is covered by $(f_1 \times g_1, s_2, t_2)$ where

$$f_1 \times g_1 : Q_2 \times P_2 \to S_1 \times T_1$$

is defined by

$$(f_1 \times g_1)(q_2, p_2) = (f_1(q_2), g_1(p_2)).$$ □

Theorem 2.7.3

Let $\mathscr{A}$, $\mathscr{A}'$, $\mathscr{B}$ be transformation semigroups such that $\mathscr{A} \leq \mathscr{B}$. Then

$$\mathscr{A} \circ \mathscr{A}' \leq \mathscr{B} \circ \mathscr{A}'.$$

Proof Suppose that $\mathscr{A} = (Q, S)$, $\mathscr{B} = (P, T)$, $\mathscr{A}' = (Q', S')$ and $\phi : P \to Q$ is a surjective partial covering function, and given $s \in S$ there is an element $t_s \in T$ covering s. Define $\phi : P \times Q' \to Q \times Q'$ by $\phi(p, q') = (\phi(p), q')$ for $(p, q') \in P \times Q'$.

Now let $s' \in S'$ and $f : Q' \to S$, define the pair (g, s') where $g : Q' \to T$ is given by

$$g(q') = t_{f(q')} \quad \text{for } q' \in Q',$$

($t_{f(q')}$ is an element of T that covers the element $f(q')$).

Now

$$\begin{aligned} \phi(p, q')(f, s') &= (\phi(p)f(q'), q's') \\ &\subseteq (\phi(pt_{f(q')}), q's') \\ &= \phi((p, q')(g, s')) \end{aligned}$$

and so (g, s') covers (f, s') with respect to ϕ. □

2.8 Examples and applications

Having patiently worked through some of the abstract theory of state machines we can now come to a brief survey of situations that give rise to such objects.

Example 2.7 Transistor components

The $NAND_2$ component consists of two transistors connected up in a simple circuit as in figure 2.8. The input at terminal I_1 is either *a current applied* denoted by 1; or *no current applied*, denoted by 0. The same choice of inputs are applied to I_2. The complete input description is an ordered pair of the form (a, b) where $a, b \in \{0, 1\}$. The transistors T_1 and T_2 are either 'off', when a current is flowing through CE; or 'on', when no current can flow through CE. The internal states of the

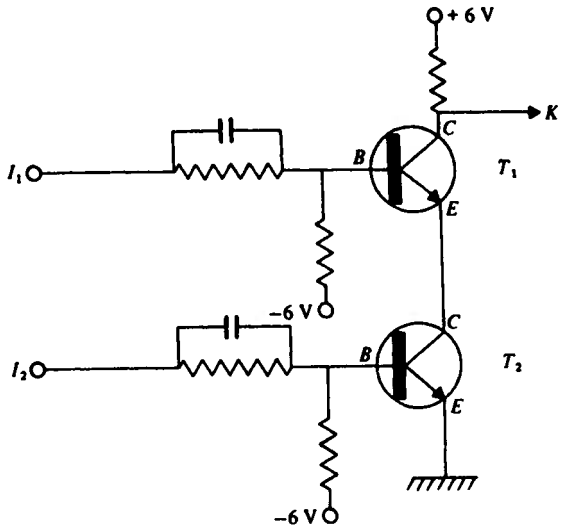


Figure 2.8. $NAND_2$ circuit.

circuit are thus described by the ordered pair $\langle \alpha, \beta \rangle$ where $\alpha, \beta \in$ {OFF, ON}. There is an output at K, either a current (1) or no current (0). We can now describe this circuit by means of a Mealy machine $\hat{\mathscr{M}} = (Q, \Sigma, \Theta, F, G)$ where

$$Q = \{\langle \text{off, off} \rangle, \langle \text{off, on} \rangle, \langle \text{on, off} \rangle, \langle \text{on, on} \rangle\}$$

$$\Sigma = \{0, 1\} \times \{0, 1\}, \Theta = \{0, 1\},$$

F is given by:

F	⟨off, off⟩	⟨off, on⟩	⟨on, off⟩	⟨on, on⟩
(0, 0)	⟨off, off⟩	⟨off, off⟩	⟨off, off⟩	⟨off, off⟩
(0, 1)	⟨off, on⟩	⟨off, on⟩	⟨off, on⟩	⟨off, on⟩
(1, 0)	⟨on, off⟩	⟨on, off⟩	⟨on, off⟩	⟨on, off⟩
(1, 1)	⟨on, on⟩	⟨on, on⟩	⟨on, on⟩	⟨on, on⟩

G is given by:

G	⟨off, off⟩	⟨off, on⟩	⟨on, off⟩	⟨on, on⟩
(0, 0)	1	1	1	1
(0, 1)	1	1	1	1
(1, 0)	1	1	1	1
(1, 1)	0	0	0	0

Many types of electronic components, from simple cases like this to complete computers can be analysed in terms of Mealy machines. Naturally the Mealy machine associated with a computer will have an enormous set of internal states, but since this set is finite the theory of finite state machines is still applicable.

As well as computer hardware it is possible to consider computer software as a type of state machine. See Chittenden [1978].

Example 2.8 Biological cells

The many complex chemical reactions that take place within various biological organisms provide a difficult problem for us to model. These reactions involve the input of various types of environmental stimulus, ranging from light of varying intensity and wavelength, temperature, different chemicals of varying concentrations, etc., both from the external environment of the organism and the more immediate environment of the surrounding cells. In many cases the chemical reactions that take place inside the cell are controlled by the *genetic* component of the *nucleus*, by the synthesis of various enzymes, etc. The net result of all this *metabolic activity* in the cell is the *synthesis* of certain chemicals necessary for growth and the operation of the organism, the storing of energy, heat, etc. So we may regard the activity of the cell as a kind of biological machine with inputs and outputs. It is not immediately apparent that the cell behaves like a Mealy machine: for example are there a finite number of internal states and a finite number of inputs and outputs? To answer this we will examine an argument which we could call the *threshold principle*.

Most of the parameters involved in the description of the state of the cell at a particular moment will be concerned with the concentrations of various chemicals, temperature, etc. and these are measured on a continuous scale. However in many situations minute changes in the concentration of a chemical do not affect the behaviour of a cell and it is only when the cell concentrations pass a *threshold value* on the measurement scale that the cell enters a different phase of behaviour. The same is true of environmental inputs: small changes in these may have no influence but larger changes, exceeding certain threshold values, will cause the cell to react in some way. The next step is to assume that there are only a finite number of these threshold values for each parameter involved in the internal description of the cell and a finite number are also assumed to exist for each input and output parameter. Let $A_1, \ldots, A_n$ be the sets of input parameters and let $\bar{A}_i$ be the set of

threshold values of the parameter set A_i. Now we form the set $\Sigma = \prod_{i=1}^{n} \bar{A}_i$, which will be the finite input set. Similarly if $B_1, \ldots, B_m$ are the sets of output parameters then we form the sets $\bar{B}_j$ $(j = 1, \ldots, m)$ of associated threshold values and put $\Theta = \prod_{j=1}^{m} \bar{B}_j$. Finally let $C_1, \ldots, C_l$ be the sets of internal parameters and $\bar{C}_k$ $(k = 1, \ldots, l)$ the associated sets of threshold values. Let $S = \prod_{k=1}^{l} \bar{C}_k$, this will be the set of internal states; Σ is the input set and Θ is the output set. The whole system can now be represented by the Mealy machine $(S, \Sigma, \Theta, F, G)$ where the next state and output functions F and G are defined appropriately. One benefit of this view is that we can consider groups of biological cells also as a Mealy machine, we just have to extend our sets S, Σ, Θ and the mappings F, G suitably. (See, e.g. Rosen [1972].)

Example 2.9 Neural networks

A model of the brain can be constructed using a simple model of the brain cell or *neuron*, see figure 2.9. We will not concern ourselves with the neurological and anatomical details of a typical brain cell here. We concentrate, instead, on the basic function of such a cell.

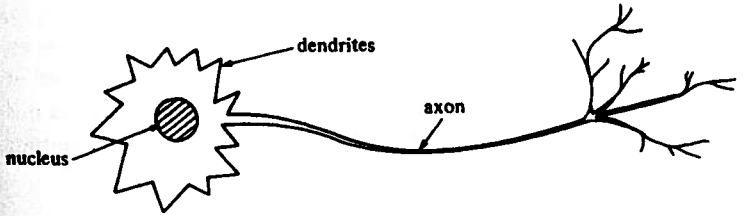


Figure 2.9. An example of a neuron.

Small electrical impulses arrive at the dendrites of the neuron, and over a short period of time they are 'summed up' by the nucleus. If the total exceeds a particular threshold value, the cell reacts by sending an impulse down the axon, the end of which branches into a number of small filaments which are in electrical contact with the dendrites of other neurons. In this way the neuron receives and propagates electrical impulses in the network of interconnected neurons which is the brain. The details of this will be found in Arbib [1964].

We can represent a neuron by a diagram such as figure 2.10, where $w_1, \ldots, w_n$ are the weights associated with the neuron's dendrites, some will be positive real numbers, indicating *excitatory* dendrites and others will be negative and mark the *inhibitory* dendrites. The threshold value

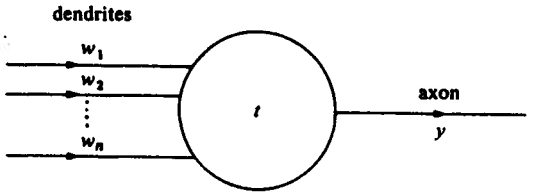


Figure 2.10. A general neuron.

is t and y is the weight of the output axon, indicating the strength of an output impulse.

Consider the simple example shown in figure 2.11. We interpret this as a Mealy machine in the following way. The set of states

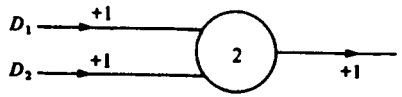


Figure 2.11. A simple neuron.

$Q = \{'on', 'off'\}$. The input and output sets are $\Sigma = \{(0, 0), (1, 0), (0, 1), (1, 1)\}$ and $\Theta = \{0, 1\}$ respectively. If the neuron is *on* and it receives an input of 1 at D_1 and 0 at D_2 we see that the neuron then turns *off* at the next point on the discrete time scale because the threshold has not been reached. However there is an output of 1. The state and output tables are:

	on	off	on	off
(0, 0)	off	off	1	0
(1, 0)	off	off	1	0
(0, 1)	off	off	1	0
(1, 1)	on	on	1	0

By constructing a model of the neural network, using simple mathematical models of the neurons connected together in certain ways we can investigate the way in which information, in the form of electrical impulses, is conveyed around the nervous system and the brain. There are, however, certain limitations to the use of this model as there are several basic assumptions that have to be made in order that the mathematics can be handled. These include the synchronization of the neurons in a convenient manner. Despite these drawbacks the model, known as

the *neural network*, has proved useful. It can be shown that such a neural network is equivalent, in a natural way, to a Mealy machine. See Arbib [1964], Minsky [1967].

Other systems which model cell behaviour, for example metabolism repair systems, are also known to be equivalent to Mealy machines (see Rosen [1972]).

Example 2.10 Metabolic pathways

The following metabolic pathway illustrates the complex series of chemical reactions that make up the Krebs cycle in mammals, this is the process by which carbohydrates are converted into energy. Figure 2.12 illustrates the progress of one molecule of oxalacetic acid through the cycle.

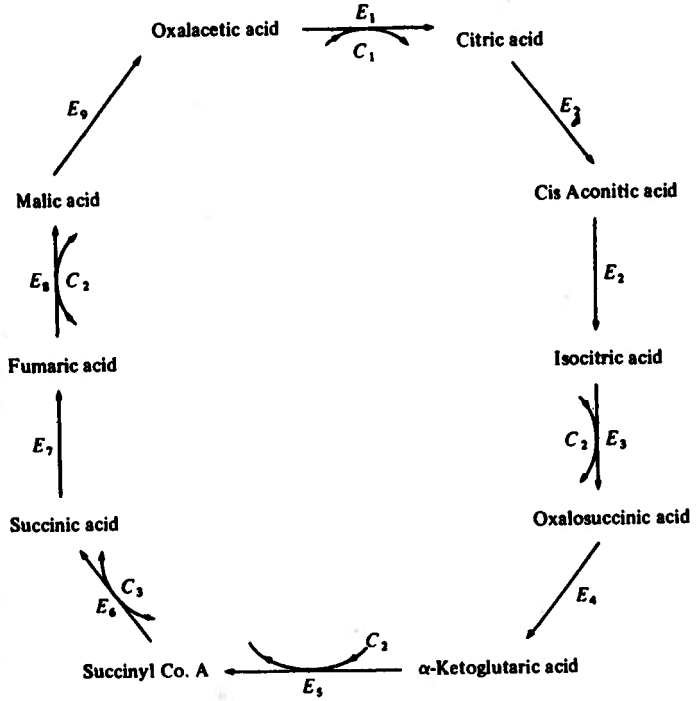


Figure 2.12. The Krebs or tricarboxylic cycle.

The letters $E_1, \ldots, E_9$ represent enzymes that are assumed to be present in sufficient concentrations during the cycle to permit the individual reactions to occur. The letters C_1, C_2, C_3 represent coenzymes.

These are also necessary for some of the reactions to take place, they combine with atoms from the substrates and are then involved in further reactions in other metabolic pathways and are eventually released for further use back in the cycle. If we make certain assumptions about the rates at which the reactions proceed and combine reactions that do not involve coenzymes we can simplify the diagram to figure 2.13. This new diagram involves the substrates:

S_1 – Oxalacetic acid, S_4 – Isocitric acid,
S_6 – α-Ketoglutaric acid, S_7 – Succinyl Co. A,
S_9 – Fumaric acid

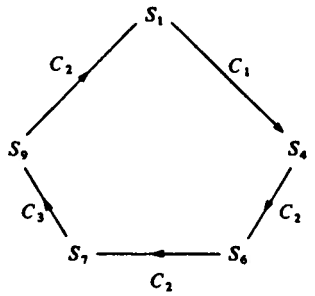


Figure 2.13. Reduced Krebs cycle.

Now an input of C_2 applied to a molecule of S_4 produces eventually a molecule of S_6 plus a molecule involving C_2 and two hydrogen atoms which are then 'passed on' in a separate series of reactions which eventually release the molecule C_2 for further use in the cycle. It is then reasonable to regard the coenzymes C_1, C_2, C_3 as inputs to a machine with states S_1, S_4, S_6, S_7, S_9. The state table of the machine is then given by:

	S_1	S_4	S_6	S_7	S_9
C_1	S_4	S_4	S_6	S_7	S_9
C_2	S_1	S_6	S_7	S_7	S_1
C_3	S_1	S_4	S_6	S_9	S_9

Note that coenzyme C_1 does not play any role in the reactions involving the states S_4, S_6, S_7, S_9 and so we regard C_1 as acting as an identity on these states. The semigroup of this state machine can now be calculated,

however the size of this semigroup is rather large. Clearly it cannot exceed $5^5 = 3125$ and in this case it is only practical to use a computer for this task. The number of elements is in fact 183. A program (in Pascal suitable for use on an Apple or ITT 2020 microcomputer) is in the appendix. This program evaluates the semigroup of a complete state machine with 5 states or less and 9 inputs or less. See Krohn, Langer and Rhodes [1967].

State machines have been used in various investigations in psychology. The work of Chomsky in psycholinguistics caused a considerable amount of interest in the use of machines for modelling the learning of language etc. For some time it appeared that the stimulus–response (S–R) theory of learning was incapable of dealing with the acquisition of machine-like behaviour, but a paper by P. Suppes [1969] claimed to indicate the ways in which S–R techniques would enable subjects to behave like simple machines. This paper resulted in some controversy (Arbib [1969], Nelson [1975]) which was then re-examined by Kieras [1976] who pointed out some ambiguities in Suppes' original work.

Another aspect of psychology that has been influenced by machines is the problem of systems that can answer questions, see Fiksel and Bower [1976]. In this paper the authors consider a network with a type of finite automaton at each node. This network models human memory and the process of question-answering is then considered to be the process of determining paths in the network bearing given sequences of labels. The automata at the nodes are only in direct communication with their immediate neighbours but this is enough for the system to determine a shortest such path.

Other areas where automata have been used as models are in economics. W. Roedding [1975] has examined the use of 'indeterminate' Mealy machines in the modelling of various economic processes.

It is likely that automata theory will feature in many modelling processes in many different subjects. This seems justification for a continued detailed study of the theory of automata and in our next chapter we will take the first steps in the procedures for simplifying and decomposing finite state machines.

2.9 Exercises

2.1 If (Q, S) is a transformation semigroup and $\mathcal{M} = (Q, S, F)$ is the state machine of (Q, S) show that $\mathbf{TS}(\mathcal{M}) = (Q, S)$. If $\mathcal{M}$ is a state machine and $(Q, S) = \mathbf{TS}(\mathcal{M})$ show that $\mathcal{M}$ may be embedded isomorphically inside the state machine of (Q, S).

2.2 Prove theorems 2.3.3 and 2.3.5.

2.3 Prove that if π is an admissible partition on $\mathcal{M} = (Q, \Sigma, F)$ then $\mathbf{TS}(\mathcal{M}/\pi) = (\mathbf{TS}(\mathcal{M}))/\pi$. Prove that $(\alpha, 1): \mathcal{M} \to \mathcal{M}/\pi$ induces the homomorphism

$$(f, g): \mathbf{TS}(\mathcal{M}) \to \mathbf{TS}(\mathcal{M}/\pi).$$

2.4 Find an example of an incomplete transformation semigroup $\mathcal{A}$ and a proper admissible partition π such that $\mathcal{A}/\langle\pi\rangle$ is complete.

2.5 Prove the homomorphism theorem for transformation semigroups.

2.6 Let $\mathcal{M}$ and $\mathcal{M}'$ be given by the tables:

$\mathcal{M}$	q_0	q_1	q_2
a	q_1	q_2	q_0
b	q_0	q_1	q_2

$\mathcal{M}'$	p_0	p_1	p_2
a	p_2	p_2	p_0
b	p_0	p_1	p_2

Calculate $\mathbf{S}(\mathcal{M})$ and $\mathbf{S}(\mathcal{M}')$ and verify theorem 2.6.3.

2.7 A partition π on the state set Q of the machine $\mathcal{M} = (Q, \Sigma, F)$ is called *elementary* if π is admissible and if given any admissible partition π' with $\pi' < \pi$ then π' is the identity partition. Let $(f, 1): (Q, \Sigma, F) \to (Q', \Sigma, F')$ be a homomorphism, show that there exists a sequence of state machines and homomorphisms

$$(Q, \Sigma, F) \to (Q_1, \Sigma, F_1) \to \ldots \to (Q_r, \Sigma, F_r) \cong (Q', \Sigma, F')$$

such that each epimorphism $(Q_i, \Sigma, F_i) \to (Q_{i+1}, \Sigma, F_{i+1})$ is induced by an elementary partition.

2.8 $\mathcal{M} = (Q, \Sigma, F)$ is called *transitive* if, for any $q, q_1 \in Q$, there exists $\alpha \in \Sigma^*$ such that

$$q_1 = qF_\alpha.$$

Let C be the group of all state machine isomorphisms $(f, 1): \mathcal{M} \to \mathcal{M}$. Prove that if $(f, 1) \in C$ and $f \neq 1_Q$ then $f(q) \neq q$ for each $q \in Q$. Define a relation $\sim$ on Q by

$$q \sim q' \Leftrightarrow q' = f(q) \quad \text{for some } (f, 1) \in C.$$

Prove that $\sim$ defines an admissible partition on $\mathcal{M}$.

2.9 Find the semigroup of the state machine $\mathcal{M}$ given by

$\mathcal{M}$	a	b	c
0	a	a	a
1	b	a	c

Construct a semigroup homomorphism between this semigroup and a proper subsemigroup of it.

2.10 Let $\mathcal{M} = (Q, \Sigma, F)$. Prove that the set $G = \{\alpha \in \mathbf{S}(\mathcal{M}) \mid Q\alpha = Q\}$ is a group.

2.11 If $\mathcal{M}$ and $\mathcal{M}'$ are state machines with $\mathcal{M} \leq \mathcal{M}'$ then there exists a subsemigroup $A \subseteq \mathbf{S}(\mathcal{M}')$ and a semigroup epimorphism $f: A \to \mathbf{S}(\mathcal{M})$.

2.12 Establish the following identities where $\mathcal{M}$, $\mathcal{M}'$ and $\mathcal{M}''$ are state machines:

$$(\mathcal{M} \times \mathcal{M}') \times \mathcal{M}'' \cong \mathcal{M} \times (\mathcal{M}' \times \mathcal{M}'')$$
$$(\mathcal{M} \wedge \mathcal{M}') \wedge \mathcal{M}'' \cong \mathcal{M} \wedge (\mathcal{M}' \wedge \mathcal{M}'')$$

(when the restricted direct product is defined).
Investigate the situation for the cascade product.

2.13 A transformation semigroup (Q, S) is called *irreducible* if

(i) $|Q| > 1$

an (ii) the only admissible partitions are the trivial partitions, i.e. the identity partition consisting of singleton blocks and the partition $\{Q\}$.
Prove that given any $q \in Q$ either $|qS| = 1$ or $qS = Q$.

2.14 If $\mathcal{M} = (Q, \Sigma, F)$ is an incomplete state machine show that

$$(\mathbf{TS}(\mathcal{M}))^c \leq \mathbf{TS}(\mathcal{M}^c).$$

2.15 Let $\mathcal{M} = (Q, \Sigma, F)$ be a state machine and suppose that $P \subseteq Q$. Calculate $\mathbf{TS}(\mathcal{M}|_P)$.

2.16 Prove theorem 2.6.4(ii).

2.17 Prove theorem 2.6.5.

2.18 If $\mathcal{M}$ and $\mathcal{N}$ are suitably defined state machines show that $\mathcal{M} \wedge \mathcal{N} \leq \mathcal{M} \times \mathcal{N}$ and $\mathcal{M} \omega \mathcal{N} \leq \mathcal{M} \circ \mathcal{N}$.

2.19 If $\mathcal{M} \leq \mathcal{M}_1$ and $\mathcal{M}_1 \leq \mathcal{M}_2$ show that $\mathcal{M} \leq \mathcal{M}_2$ where $\mathcal{M}$, $\mathcal{M}_1$, $\mathcal{M}_2$ are state machines. Establish a similar result for transformation semigroups.

2.20 Let $(\alpha, \beta): \mathcal{M} \to \mathcal{M}_1$ be a state machine homomorphism. If (α, β) is an epimorphism, prove that $\mathcal{M}_1 \leq \mathcal{M}$ and if (α, β) is a monomorphism prove that $\mathcal{M} \leq \mathcal{M}_1$.

2.21 Prove that if $\mathcal{A}$ and $\mathcal{B}$ are transformation semigroups such that

$$\mathcal{A} \leq \mathcal{B}$$

then $\mathcal{A}^{\cdot} \leq \mathcal{B}^{\cdot}$, $\bar{\mathcal{A}} \leq \bar{\mathcal{B}}$, $\mathcal{A}^c \leq \mathcal{B}^c$.

2.22 Prove that for state machines $\mathcal{M}$, $\mathcal{N}$, $\overline{\mathcal{M} \times \mathcal{N}} \leq \bar{\mathcal{M}} \times \bar{\mathcal{N}}$, $\overline{\mathcal{M} \wedge \mathcal{N}} \leq \bar{\mathcal{M}} \wedge \bar{\mathcal{N}}$, $\overline{\mathcal{M} \circ \mathcal{N}} \leq \bar{\mathcal{M}} \circ \bar{\mathcal{N}}$, and $\overline{\mathcal{M}\omega\mathcal{N}} \leq \bar{\mathcal{M}}\bar{\omega}\bar{\mathcal{N}}$ for suitable ω, $\bar{\omega}$.

2.23 Let $\mathcal{M} = (Q, \Sigma, F)$ be a state machine. Put $\mathcal{M}^{\cdot} = (Q, \Sigma \cup \{1_Q\}, F^{\cdot})$ where $qF^{\cdot}_{\sigma} = qF_{\sigma}$, $qF^{\cdot}_{1_Q} = q$, $q \in Q$. Prove that $\mathbf{TS}(\mathcal{M}^{\cdot}) \leq (\mathbf{TS}(\mathcal{M}))^{\cdot}$.

2.24 *If* $\mathcal{M}$, $\mathcal{N}$, $\mathcal{P}$, $\mathcal{R}$ are state machines and $\mathcal{M} \leq \mathcal{N}$ prove that $\mathcal{P}\omega\mathcal{M} \leq \mathcal{P}\omega\mathcal{N}$, and

$$\mathcal{P} \leq \mathcal{M}\omega\mathcal{R} \Rightarrow \mathcal{P} \leq \mathcal{N}\omega\mathcal{R}$$

$$\mathcal{P} \leq \mathcal{R}\omega\mathcal{M} \Rightarrow \mathcal{P} \leq \mathcal{R}\omega\mathcal{N}.$$

2.25 Show that if $\mathcal{A}$ is a transformation semigroup then

$$\mathcal{A} \circ \mathbf{1}^{\cdot} \cong \mathcal{A} \cong \mathbf{1}^{\cdot} \circ \mathcal{A}.$$

2.26 Let $\mathcal{M} = (Q, \Sigma, F)$ and consider $\bar{\mathcal{M}} = (Q, \Sigma \cup Q, \bar{F})$ where $\bar{F}_{\sigma}(q) = F_{\sigma}(q)$, $\bar{F}_{q'}(q) = q'$ for $q, q' \in Q$, $\sigma \in \Sigma$. Show that $\mathbf{TS}(\bar{\mathcal{M}}) = \overline{\mathbf{TS}(\mathcal{M})}$.

2.27 *Let* $\mathcal{M}$ be a state machine. Prove that

$$\mathbf{TS}(\mathcal{M})^{\cdot} = \mathbf{TM}(\mathcal{M}).$$

2.28 $\mathcal{A}$, $\mathcal{B}$ are transformation semigroups, show that

$$\overline{\mathcal{A} \circ \mathcal{B}} \leq \bar{\mathcal{A}} \circ \bar{\mathcal{B}}, \quad (\mathcal{A} \circ \mathcal{B})^c \leq \mathcal{A}^c \circ \mathcal{B}^c.$$

2.29 Let $\mathcal{A} = (Q, S)$, $\mathcal{B} = (P, T)$ be transformation semigroups such that $Q \cap P = \varnothing$. We define the *join* of $\mathcal{A}$ and $\mathcal{B}$ to be $\mathcal{A} \vee \mathcal{B} = (Q \cup P, S \vee T)$ where the action $*$ is defined by:

$$q * s = qs \quad (q \in Q, s \in S)$$

$$q * t = \varnothing \quad (q \in Q, t \in T)$$

$$p * s = \varnothing \quad (p \in P, s \in S)$$

$$p * t = pt \quad (p \in P, t \in T)$$

$$q * 0 = \varnothing \quad (q \in Q)$$

$$p * 0 = \varnothing \quad (p \in P)$$

Prove that $\mathcal{A} \vee \mathcal{B}$ is a transformation semigroup and that the join is an associative product.

2.30 If $\mathcal{A}$, $\mathcal{A}_1$, $\mathcal{B}$, $\mathcal{B}_1$ are transformation semigroups show that

$$\bar{\mathcal{A}} \vee \bar{\mathcal{B}} \leq \overline{\mathcal{A} \vee \mathcal{B}}$$

$$\mathcal{A} \vee \mathcal{B} \leq (\mathcal{A} \vee \mathcal{B})^{\cdot}$$

$$\mathcal{A} \circ (\mathcal{B}_1 \vee \mathcal{B}_2) \leq (\mathcal{A} \circ \mathcal{B}_1) \vee (\mathcal{A} \circ \mathcal{B}_2)$$

$\mathcal{A} \leq \mathcal{B}$ implies $\mathcal{A} \vee \mathcal{A}_1 \leq \mathcal{B} \vee \mathcal{A}_1$.

2.31 *Let* $\mathcal{A} = (Q, S)$, $\mathcal{B} = (P, T)$ be transformation semigroups with $S \neq \varnothing$, $T \neq \varnothing$. Define the *sum* $\mathcal{A} + \mathcal{B} = (Q \cup P, S \times T)$ with the operation $*$ given by

$$q(s, t) = qs \quad (q \in Q)$$

$$p(s, t) = pt \quad (p \in P).$$

In the case where S is empty we put

$$\mathcal{A} + \mathcal{B} = (Q \cup P, T)$$

and if T is empty define

$$\mathcal{A} + \mathcal{B} = (Q \cup P, S).$$

Prove that $\mathcal{A} + \mathcal{B}$ is a transformation semigroup.
Show that $\mathcal{A} \vee \mathcal{B} \leq \mathcal{A} + \mathcal{B}$.

2.32 Find algebraic descriptions for the transformation semigroups $\mathcal{A} \times \mathcal{B}$ and $\mathcal{A} \circ \mathcal{B}$ where $\mathcal{A}$ and $\mathcal{B}$ are incomplete.

2.33 Let $\mathcal{F}$ be given by

then $\mathcal{F} \leq \bar{\mathbf{2}}$. Show that if the semigroup of $\mathcal{F}$ is $S = \{\theta, \sigma, \tau, \sigma\tau, \tau\sigma\}$ and the semigroup of $\bar{\mathbf{2}}$ is $T = \{\alpha, \beta\}$ then the function $\xi: S \to T$ given by $\xi(\sigma) = \xi(\tau\sigma) = \xi(\theta) = \alpha$, $\xi(\tau) = \xi(\sigma\tau) = \beta$, is not a semigroup homomorphism.

3

Decompositions

The previous chapter established that finite state machines and transformation semigroups are natural subjects for study and our next task is to initiate the algebraic theory of these objects. As with other algebraic theories one approach is to replace a general state machine (or transformation semigroup)by a collection of 'algebraically simpler' machines (or transformation semigroups) connected up in suitable ways. We have already remarked that one distinguishing feature of this algebraic theory compared to others is that we are more interested in what the machines do than what they look like. Consequently we will be using the concept of a covering rather than the concept of an isomorphism. The extra flexibility allowed by this approach will be of great use to us.

Our main aim is the development of decomposition theorems. To take the case of the finite state machines first, we will construct coverings of a given state machine in such a way that the covering machine is a product, either direct or cascade, of 'simpler' machines. So we will expect statements of the form

$$\mathcal{M} \leq \mathcal{N}_1 \omega_1 \mathcal{N}_2 \omega_2 \ldots \omega_{n-1} \mathcal{N}_n$$

where $\mathcal{M}, \mathcal{N}_1, \ldots, \mathcal{N}_n$ are state machines and the connecting mappings $\omega_1, \ldots, \omega_n$ are defined suitably. Recall that the cascade product is a generalization of the restricted direct product so that this type of decomposition will be the most general. However if we can replace some of the cascade products by restricted direct products we will do so because this will yield a much more efficient covering. (The semigroup of the covering machine will be smaller.)

Similarly we will attempt to derive coverings of transformation semigroups of the form $\mathcal{A} \leq \mathcal{B}_1 \circ \mathcal{B}_2 \circ \ldots \circ \mathcal{B}_n$ where $\mathcal{A}, \mathcal{B}_1, \ldots, \mathcal{B}_n$ are

transformation semigroups. To be valuable the transformation semigroups $\mathcal{B}_1, \ldots, \mathcal{B}_n$ should have some desirable properties lacking in $\mathcal{A}$.

There will be a possibility of transferring from decompositions of state machines to decompositions of transformation semigroups and vice versa. Sometimes this process can be carried out without losing much in the way of efficiency, but generally there will be a slight loss.

There are two basic ways of obtaining decompositions. One way examines properties of the state set and for these decompositions it is usually easiest to deal with state machines and state machine decompositions. The other approach is based on properties of the input set and as these properties are usually expressed in terms of the semigroup of the machine, it is the theory of transformation semigroups that will be the most useful here. In the next chapter we will consider a decomposition theory that uses both approaches and this is best examined using transformation semigroups.

First, however, we must find a relationship between the two approaches to decomposition theories.

3.1 Decompositions

Let $\mathcal{M}$ be a state machine. A *cascade decomposition* for $\mathcal{M}$ is a covering

$$\mathcal{M} \leq \mathcal{N}_1 \omega_1 \mathcal{N}_2 \omega_2 \ldots \omega_{n-1} \mathcal{N}_n$$

where $\mathcal{N}_1, \ldots, \mathcal{N}_n$ are state machines. Naturally it is easy to construct trivial' examples of such coverings, but we will only be interested in those cases where the machines $\mathcal{N}_1, \ldots, \mathcal{N}_n$ are in some sense simpler than $\mathcal{M}$; usually this means that the state sets of $\mathcal{N}_1, \ldots, \mathcal{N}_n$ are all 'smaller' than the state set of $\mathcal{M}$ or the semigroups of the $\mathcal{N}_1, \ldots, \mathcal{N}_n$ are 'simpler' than the semigroup of $\mathcal{M}$.

A decomposition of the form

$$\mathcal{M} \leq \mathcal{N}_1 \circ \mathcal{N}_2 \circ \ldots \circ \mathcal{N}_n,$$

where $\mathcal{M}, \mathcal{N}_1, \ldots, \mathcal{N}_n$ are state machines is called a *wreath decomposition of* $\mathcal{M}$. Clearly $\mathcal{M} \leq \mathcal{N}_1 \omega_1 \mathcal{N}_2 \omega_2 \ldots \omega_{n-1} \mathcal{N}_n$ implies $\mathcal{M} \leq \mathcal{N}_1 \circ \mathcal{N}_2 \circ \ldots \circ \mathcal{N}_n$ by exercise 2.18.

Now let $\mathcal{A}$ be a transformation semigroup, a *wreath decomposition* for $\mathcal{A}$ is a covering

$$\mathcal{A} \leq \mathcal{B}_1 \circ \mathcal{B}_2 \circ \ldots \circ \mathcal{B}_n$$

where $\mathcal{B}_1, \ldots, \mathcal{B}_n$ are transformation semigroups. In many cases the semigroups of $\mathcal{B}_1, \ldots, \mathcal{B}_n$ are smaller than the semigroup of $\mathcal{A}$.

To compare the two concepts we need the following results.

Theorem 3.1.1

(i) Let $\mathcal{M}$ and $\mathcal{N}$ be state machines with $\mathcal{M} \leq \mathcal{N}$, then $\mathbf{TS}(\mathcal{M}) \leq \mathbf{TS}(\mathcal{N})$.

(ii) Let $\mathcal{A}$ and $\mathcal{B}$ be transformation semigroups with $\mathcal{A} \leq \mathcal{B}$, then $\mathbf{SM}(\mathcal{A}) \leq \mathbf{SM}(\mathcal{B})$.

Theorem 3.1.2

(i) Let $\mathcal{M} \leq \mathcal{N}_1 \omega_1 \mathcal{N}_2 \omega_2 \ldots \omega_{n-1} \mathcal{N}_{n-1}$ be a cascade covering of state machines, then

$$\mathbf{TS}(\mathcal{M}) \leq \mathbf{TS}(\mathcal{N}_1) \circ \mathbf{TS}(\mathcal{N}_2) \circ \ldots \circ \mathbf{TS}(\mathcal{N}_n)$$

is a wreath decomposition of transformation semigroups.

(ii) Let $\mathcal{A} \leq \mathcal{B}_1 \circ \mathcal{B}_2 \circ \ldots \circ \mathcal{B}_n$ be a wreath decomposition of transformation semigroups, then

$$\mathbf{SM}(\mathcal{A}) \leq \mathbf{SM}(\mathcal{B}_1) \circ \mathbf{SM}(\mathcal{B}_2) \circ \ldots \circ \mathbf{SM}(\mathcal{B}_n)$$

is a wreath decomposition of state machines.

Theorem 3.1.3

(i) Let $\mathcal{M} \leq \mathcal{N}_1 \wedge \mathcal{N}_2 \wedge \ldots \wedge \mathcal{N}_n$ be a covering of state machines, then

$$\mathbf{TS}(\mathcal{M}) \leq \mathbf{TS}(\mathcal{N}_1) \times \ldots \times \mathbf{TS}(\mathcal{N}_n)$$

is a covering of transformation semigroups.

(ii) Let $\mathcal{A} \leq \mathcal{B}_1 \times \mathcal{B}_2 \times \ldots \times \mathcal{B}_n$ be a covering of transformation semigroups, then

$$\mathbf{SM}(\mathcal{A}) \leq \mathbf{SM}(\mathcal{B}_1) \times \mathbf{SM}(\mathcal{B}_2) \times \ldots \times \mathbf{SM}(\mathcal{B}_n)$$

is a covering of state machines.

We will prove theorem 3.1.1; the other two results follow from it and results in chapter 2 using induction.

Proof of theorem 3.1.1

(i) Let $\mathcal{M} = (Q, \Sigma, F)$, $\mathcal{N} = (Q', \Sigma', F')$ and $\phi : Q' \to Q$ a partial surjective function and $\xi : \Sigma \to \Sigma'$ a function. Let $s \in \mathbf{S}(\mathcal{M})$, then $s = [\alpha]$ for some $\alpha \in \Sigma^+$. Put $\beta = \xi(\alpha)$ and consider the element $t_s = [\beta]' \in \mathbf{S}(\mathcal{N})$. We will establish that t_s covers s, for if $q' \in Q$ then

$$\begin{aligned} \phi(q') \cdot s = \phi(q') \cdot [\alpha] &= (\phi(q'))F_\alpha \\ &\subseteq \phi(q' F'_{\xi(\alpha)}) \\ &= \phi(q'[\beta]') \\ &= \phi(q' \cdot t_s). \end{aligned}$$

Therefore $\mathbf{TS}(\mathcal{M}) \leq \mathbf{TS}(\mathcal{N})$.

(ii) Now let $\mathcal{A} = (Q, S)$, $\mathcal{B} = (Q', T)$ and consider

$$\mathbf{SM}(\mathcal{A}) = (Q, S, F), \mathbf{SM}(\mathcal{B}) = (Q', T, F')$$

where

$$qF_s = qs \quad \text{for } q \in Q, s \in S$$

and

$$q'F'_t = q't \text{ for } q' \in Q', t \in T.$$

Define a function $\phi : S \to T$ by

$$\xi(s) = t_s$$

where t_s is a suitably chosen element of T that covers $s \in S$.

Then, for $q' \in Q'$, $s \in S$ we have

$$\phi(q')F_s = \phi(q')s \subseteq \phi(q' \cdot t_s) = \phi(q'F'_{\xi(s)})$$

and thus $\mathbf{SM}(\mathcal{M}) \leq \mathbf{SM}(\mathcal{N})$. □

We start our decomposition with some useful results involving state machines.

3.2 Orthogonal partitions

Let $\mathcal{M} = (Q, \Sigma, F)$ be a state machine and suppose that $\pi = \{H_i\}_{i \in I}$ is a non-trivial admissible partition on Q. We call π *orthogonal* if there exists a non-trivial admissible partition τ on Q such that $\pi \cap \tau = 1_Q$.

If $\tau = \{K_j\}_{j \in J}$ then we have $|H_i \cap K_j| \leq 1$ for any $i \in I$, $j \in J$.

Given an admissible partition π on Q it is easy to construct a partition τ on Q such that $\pi \cap \tau = 1_Q$, for example if H_1 is a π-block of maximal

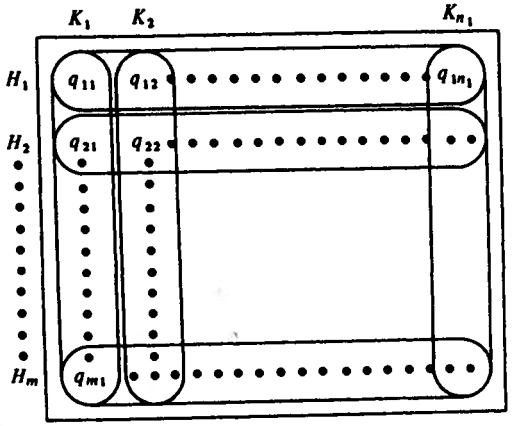
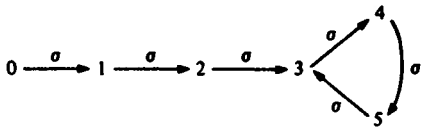


Figure 3.1. A general construction for τ.

order and $H_1 = \{q_{11}, \ldots, q_{1n_1}\}$ we construct K_1 by selecting q_{11} and one element from each other π-block. The block K_2 is constructed from q_{12} plus a new element from every other π-block, and so on. We eventually obtain a partition $\tau = \{K_1, K_2, \ldots, K_{n_1}\}$ satisfying $\pi \cap \tau = 1_Q$ with τ non-trivial. However τ may not be admissible. Note that τ contains n_1 distinct blocks. See figure 3.1.

Example 3.1

Consider the cyclic state machine $\mathcal{M} = (Q, \Sigma, F)$ defined by



and let $\pi = \{\{0\}, \{1\}, \{2\}, \{3, 4, 5\}\}$, then π is an admissible partition. Let $\tau' = \{\{0, 1, 2, 3\}, \{4\}, \{5\}\}$ then $\pi \cap \tau' = 1_Q$ but τ' is not admissible. However $\tau = \{\{0, 3\}, \{1, 4\}, \{2, 5\}\}$ is an admissible partition, $\pi \cap \tau = 1_Q$ and π is thus orthogonal (of course so is τ!).

The existence of an orthogonal admissible partition is very valuable.

Theorem 3.2.1

Let $\mathcal{M} = (Q, \Sigma, F)$ be a state machine and suppose that π is an orthogonal admissible partition on Q. If τ is a non-trivial admissible partition on Q such that $\pi \cap \tau = 1_Q$ then

$$\mathcal{M} \leq \mathcal{M}/\pi \wedge \mathcal{M}/\tau.$$

Proof Let $\pi = \{H_i\}_{i \in I}$, $\tau = \{K_j\}_{j \in J}$ and put

$$L = \{(H_i, K_j) \in \pi \times \tau \mid H_i \cap K_j \neq \varnothing\}.$$

Define a partial function $\phi : \pi \times \tau \to Q$ with domain L by

$$\phi((H_i, K_j)) = q \Leftrightarrow H_i \cap K_j = \{q\}.$$

Then ϕ maps a pair consisting of a π-block and a τ-block to their common element, if it exists. The state machine $\mathcal{M}/\pi \wedge \mathcal{M}/\tau$ is defined to be

$$(\pi \times \tau, \Sigma, F^{\wedge})$$

where

$$(H_i, K_j)F_\sigma^{\wedge} = ([(H_i)F_\sigma]_\pi, [(K_j)F_\sigma]_\tau)$$

for $H_i \in \pi$, $K_j \in \tau$, $\sigma \in \Sigma$; where $[(H_i)F_\sigma]_\pi$ is the π-block containing the states $(H_i)F_\sigma = \{qF_\sigma \mid q \in H_i\}$ etc.

Now for $(H_i, K_j) \in L$ and $\sigma \in \Sigma$

$$\begin{aligned}(\phi((H_i, K_j)))F_\sigma &= qF_\sigma \qquad \text{where } H_i \cap K_j = \{q\} \\ &\in (H_i)F_\sigma \cap (K_j)F_\sigma \\ &\subseteq [(H_i)F_\sigma]_\pi \cap [(K_j)F_\sigma]_\tau\end{aligned}$$

so

$$\begin{aligned}qF_\sigma &= \phi([(H_i)F_\sigma]_\pi \cap [(K_j)F_\sigma]_\tau) \\ &= \phi((H_i, K_j)F_\sigma^{\wedge}) \qquad \text{if } qF_\sigma \neq \varnothing\end{aligned}$$

and generally

$$qF_\sigma \subseteq \phi((H_i, K_j)F_\sigma^{\wedge})$$

□

Corollary 3.2.2

$$\begin{aligned}\mathrm{TS}(\mathcal{M}) &\leq \mathrm{TS}(\mathcal{M}/\pi) \times \mathrm{TS}(\mathcal{M}/\tau) \\ &= (\mathrm{TS}(\mathcal{M}))/\langle\pi\rangle \times (\mathrm{TS}(\mathcal{M}))/\langle\tau\rangle\end{aligned}$$

The concept of an orthogonal admissible partition on a transformation semigroup can be defined in an analogous fashion and it is then a straightforward matter to deduce that if π is orthogonal on $\mathcal{A} = (Q, S)$ then

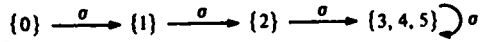
$$\mathcal{A} \leq \mathcal{A}/\langle\pi\rangle \times \mathcal{A}/\langle\tau\rangle.$$

Example 3.2

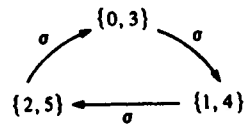
Returning to our previous example (3.1) we see that

$$\mathcal{M} \leq \mathcal{M}/\pi \times \mathcal{M}/\tau$$

where $\mathcal{M}/\pi$ is given by



and $\mathcal{M}/\tau$ is given by



In the notation of transformation semigroups

$$\mathrm{TS}(\mathcal{M}) = \mathscr{C}_{(3,3)}$$

$$\mathrm{TS}(\mathcal{M}/\pi) = \mathscr{C}_{(1,3)}$$

and

$$\mathrm{TS}(\mathcal{M}/\tau) = \mathbf{Z}_3$$

and so

$$\mathscr{C}_{(3,3)} \leq \mathscr{C}_{(1,3)} \times \mathbf{Z}_3.$$

Example 3.3

Let $\mathcal{M} = (Q, \Sigma, F)$ be a state machine with the property that $|(Q)F_\sigma| \leq 1$ for all $\sigma \in \Sigma$. Such machines are called *reset machines*. The transformation semigroup of $\mathcal{M}$ is covered by $(\overline{Q, \varnothing})$ which is closed. Now let $\pi = \{H_i\}_{i \in I}$ be any partition of Q, then π is admissible, for if $\sigma \in \Sigma$, $i \in I$, then $|(H_i)F_\sigma| \leq 1$ and so $(H_i)F_\sigma \subseteq H_j$ for some $j \in I$. Consequently all partitions of Q are admissible. If $|Q| > 1$ let $q_1, q_2 \in Q$ and consider the partition $\pi = \{\{q_1, q_2\}, Q \backslash \{q_1, q_2\}\}$ which has two blocks. It is admissible and orthogonal; choose any partition τ such that $\pi \cap \tau = 1_Q$. Then

$$\mathcal{M} \leq \mathcal{M}/\pi \times \mathcal{M}/\tau.$$

Now $\mathcal{M}/\pi$ has two states and is a reset machine. The state machine $\mathcal{M}/\tau$ is also a reset machine and has fewer states than $\mathcal{M}$. We can therefore apply the process again to the state machine $\mathcal{M}/\tau$ by choosing a partition with two blocks and continuing as before. Eventually this process will cease since $|Q|$ is finite. We will then have established that $\mathcal{M} \leq \mathcal{N}_1 \times \mathcal{N}_2 \times \ldots \times \mathcal{N}_n$ where each $\mathcal{N}_i$ is a two-state reset machine. For the transformation semigroup case we have

$$\mathbf{TS}(\mathcal{M}) \leq (\overline{Q, \varnothing}) \leq \mathbf{TS}(\mathcal{N}_1) \times \mathbf{TS}(\mathcal{N}_2) \times \ldots \times \mathbf{TS}(\mathcal{N}_n).$$

Each $\mathbf{TS}(\mathcal{N}_i)$ can be covered by $\bar{\mathbf{2}}$ and so

$$\mathbf{TS}(\mathcal{M}) \leq (\overline{Q, \varnothing}) \leq \bar{\mathbf{2}} \times \bar{\mathbf{2}} \times \ldots \times \bar{\mathbf{2}} = \prod^k \bar{\mathbf{2}}$$

where $k = |Q| - 1$ and $\prod^k \bar{\mathbf{2}}$ means the direct product of k copies of the transformation semigroup $\bar{\mathbf{2}}$. In fact better decompositions exist for this type of state machine. (See example 3.2.)

The results in theorems 3.1.1, 3.1.2, 3.1.3 have the obvious extensions to transformation monoids and so we have

$$\mathbf{TM}(\mathcal{M}) \leq (\overline{Q, \varnothing})^* \leq \prod^k \bar{\mathbf{2}}^*$$

where $k = |Q| - 1$.

3.3 General admissible partitions

The next step is to examine the situation where π is an admissible partition which is not necessarily orthogonal. First let $\mathbf{max}(\pi)$ indicate the maximum size of a π-block.

Theorem 3.3.1

Let $\mathcal{M} = (Q, \Sigma, F)$ be a state machine and $\pi = \{H_i\}_{i \in I}$ a non-trivial admissible partition on Q. There exists a state machine $\mathcal{N} = (Q', \Sigma', F')$ such that

$$\mathcal{M} \leq \mathcal{N} \omega \mathcal{M}/\pi$$

for some $\omega : \pi \times \Sigma \to \Sigma'$ and $|Q'| = \mathbf{max}(\pi)$.

Proof Again let $\tau = \{K_j\}_{j \in J}$ be a partition of Q satisfying $\pi \cap \tau = 1_Q$. Construct a state machine $\mathcal{N} = (Q', \Sigma', F')$ by putting

$$Q' = \tau$$

$$\Sigma' = \pi \times \Sigma$$

and defining

$$(K_j)F'_{(H_i,\sigma)} = [(H_i \cap K_j)F_\sigma]_\tau$$

for each $\sigma \in \Sigma$, $H_i \in \pi$, $K_j \in \tau$. Define $\omega = 1_{\Sigma'}$ and consider the state machine

$$\mathcal{N} \omega \mathcal{M}/\pi = (\tau \times \pi, \Sigma, \bar{F})$$

where

$$(K_j, H_i)\bar{F}_\sigma = ((K_j)F'_{(H_i,\sigma)}, [(H_i)F_\sigma]_\pi)$$

for $\sigma \in \Sigma$, $K_j \in \tau$ and $H_i \in \pi$. Put $L = \{(K_j, H_i) \mid K_j \in \tau, H_i \in \pi, K_j \cap H_i \neq \varnothing\}$. Define $\phi : \tau \times \pi \to Q$ to be the surjective partial function with domain L given by

$$\phi(K_j, H_i) = q \Leftrightarrow K_j \cap H_i = \{q\}.$$

Now consider $(K_j, H_i) \in L$, $\sigma \in \Sigma$ and note that

$$(\phi(K_j, H_i))F_\sigma = qF_\sigma$$

where $K_j \cap H_i = \{q\}$.

Now $(K_j \cap H_i)F_\sigma = \{qF_\sigma\} \subseteq [(H_i)F_\sigma]_\pi$ and since $\{qF_\sigma\} \subseteq [(K_j \cap H_i)F_\sigma]_\tau$ we have

$$[(K_j \cap H_i)F_\sigma]_\tau \cap [(H_i)F_\sigma]_\pi = \{qF_\sigma\}.$$

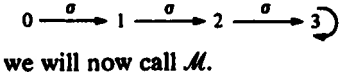
Thus $(\phi(K_j, H_i))F_\sigma \subseteq \phi((K_j, H_i)\bar{F}_\sigma)$ in general. □

Corollary 3.3.2

$\mathbf{TS}(\mathcal{M}) \leq \mathbf{TS}(\mathcal{N}) \circ (\mathbf{TS}(\mathcal{M}))/\langle \pi \rangle$.

Example 3.4

In a previous example (3.2) we obtained the state machine

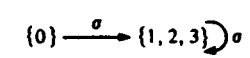


which we will now call $\mathcal{M}$.

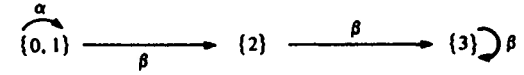
This has an admissible partition

$$\pi = \{\{0\}, \{1, 2, 3\}\}.$$

Putting $\tau = \{\{0, 1\}, \{2\}, \{3\}\}$ we see that $\mathcal{M}/\pi$ is given by



and $\mathcal{N}$ is given by



where $\alpha = (\sigma, \{0\})$, $\beta = (\sigma, \{1, 2, 3\})$. For the transformation semigroup situation $\mathbf{TS}(\mathcal{M}/\pi) \cong \mathcal{C}$ and $\mathbf{TS}(\mathcal{N}) \leq (\mathcal{C}_{(1,2)})^{\cdot}$. Now $\mathcal{C}_{(1,3)} \leq (\mathcal{C}_{(1,2)})^{\cdot} \circ \mathcal{C}$ and $\mathcal{C}_{(1,2)} \leq \bar{\mathbf{2}} \circ \mathcal{C}$, so $\mathcal{C}_{(1,3)} \leq \bar{\mathbf{2}}^{\cdot} \circ \mathcal{C} \circ \mathcal{C}$.

The result in 3.3.2 is an indication both of the possibilities of finding useful decompositions using an admissible partition π and of the limitations of this approach, because of the difficulties of determining the semigroup of $\mathbf{TS}(\mathcal{N})$. The choice of the partition τ will have a major influence on the ease of determining the semigroup of $\mathbf{TS}(\mathcal{N})$. In some of our later results we will, in effect, make a suitable choice of τ so that $\mathbf{TS}(\mathcal{N})$ has a particularly desirable form. It is then often easier to construct the covering of 3.3.2 directly rather than to calculate $\mathbf{TS}(\mathcal{N})$ and this is the approach that we will usually take.

Let $\mathcal{M} = (Q, \Sigma, F)$ be a state machine, an admissible partition $\pi = \{H_i\}_{i \in I}$ is called *maximal* if π is non-trivial and if τ is any admissible partition with $\pi \leq \tau \leq \{Q\}$ then either $\tau = \pi$ or $\tau = \{Q\}$. So a maximal partition is an admissible partition such that no strictly larger non-trivial admissible partitions exist.

A state machine $\mathcal{N} = (Q', \Sigma', F')$ is called *irreducible* if $|Q'| > 1$ and the only admissible partitions on Q' are trivial (i.e. $1_{Q'}$ and $\{Q'\}$). (See exercise 2.13.)

Theorem 3.3.3

Let $\mathcal{M} = (Q, \Sigma, F)$ be a state machine. π is a maximal admissible partition on Q if and only if $\mathcal{M}/\pi$ is irreducible.

Proof See exercise 3.3.

Theorem 3.3.1 can now be applied to produce the *irreducible decomposition.*

Theorem 3.3.4

Let $\mathcal{M} = (Q, \Sigma, F)$ be any state machine, $|Q| = m \geq 2$, then

$$\mathcal{M} \leq \mathcal{N}_1 \omega_1 \mathcal{N}_2 \omega_2 \ldots \omega_{n-1} \mathcal{N}_n$$

where $\mathcal{N}_1, \ldots, \mathcal{N}_n$ are irreducible state machines each with state sets of order less than m.

Proof Choose a maximal admissible partition π of Q, since $|Q| \neq 1$, and apply theorem 3.3.1, then $\mathcal{M} \leq \mathcal{N} \omega \mathcal{M}/\pi$ for suitable $\mathcal{N}$ and ω. The state set of $\mathcal{N}$ is by construction of order equal to the size of the largest π-block, which is less than m. Similarly $\mathcal{M}/\pi$ has state set equal to the number of distinct π-blocks, which is also less than m. Furthermore $\mathcal{M}/\pi$ is irreducible by 3.3.3. Now apply 3.3.1 to the state machine $\mathcal{N}$, having first found a maximal admissible partition π' for $\mathcal{N}$. Then

$$\mathcal{N} \leq \mathcal{N}' \omega' \mathcal{N}/\pi'$$

and so

$$\mathcal{M} \leq \mathcal{N}' \omega' \mathcal{N}/\pi' \omega \mathcal{M}/\pi.$$

Continuing in this way the finiteness of m forces a halt and clearly all the state machines in the decomposition are irreducible. If $|Q| = 2$ then $\mathcal{M}$ is already irreducible. □

Corollary 3.3.5

Let $\mathcal{A} = (Q, S)$ be any transformation semigroup with $|Q| = m \geq 2$, then

$$\mathcal{A} \leq \mathcal{B}_1 \circ \mathcal{B}_2 \circ \ldots \circ \mathcal{B}_n$$

where $\mathcal{B}_i = (Q_i, S_i)$ are transformation semigroups satisfying

(i) $|Q_i| < m$

(ii) $\mathcal{B}_i$ have no non-trivial admissible partitions.

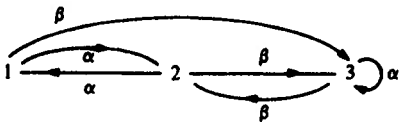
Proof Apply the transformation semigroup process to the decomposition of 3.3.4 noting that irreducible state machines give rise to transformation semigroups satisfying condition (ii). □

This decomposition is not as useful as it may appear, principally because we do not have a clear idea of what irreducible state machines look like. This problem will be examined in section 3.7 and the exercises. However the study of irreducible state machines may well be of some importance since many seem to arise naturally in applications. The example in chapter 2 of a state machine arising from a metabolic pathway

is irreducible (and it may be that biological examples are generally irreducible for reasons of stability).

Example 3.5

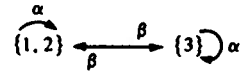
Consider the state machine $\mathcal{M} = (Q, \Sigma, F)$ given by



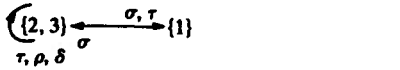
The partition $\pi = \{\{1, 2\}, \{3\}\}$ is admissible and maximal. Choose $\tau = \{\{2, 3\}, \{1\}\}$, then $\pi \cap \tau = 1_Q$ but τ is not admissible. Applying 3.3.1 we obtain

$$\mathcal{M} \leq \mathcal{N} \omega \mathcal{M}/\pi$$

where $\mathcal{M}/\pi$ is given by



and $\mathcal{N}$ is given by



where $\sigma = (\alpha, \{1, 2\})$, $\tau = (\beta, \{1, 2\})$, $\rho = (\alpha, \{3\})$, $\delta = (\beta, \{3\})$. Both $\mathcal{N}$ and $\mathcal{M}/\pi$ are irreducible. Converting to transformation semigroups we have

$$\mathbf{TS}(\mathcal{M}) \leq \mathbf{TS}(\mathcal{N}) \circ \mathbf{Z}_2$$

and since

$$\mathbf{TS}(\mathcal{N}) \leq \bar{\mathbf{Z}}_2$$

we obtain

$$\mathbf{TS}(\mathcal{M}) \leq \bar{\mathbf{Z}}_2 \circ \mathbf{Z}_2.$$

(Compare this with example 4.8.)

3.4 Permutation–reset machines

An important class of state machines are the permutation–reset state machines. Let $\mathcal{M} = (Q, \Sigma, F)$ be a state machine with $|Q| > 1$ and suppose that for each $\sigma \in \Sigma$ either $(Q)F_\sigma = Q$ or $|(Q)F_\sigma| = 1$, then we call $\mathcal{M}$ a *permutation–reset machine*. (Each input either defines a permutation of Q or a reset.) We shall see shortly how these arise naturally, but in the meantime we will examine a method of decomposing them.

First we call a state machine $\bar{\mathcal{M}} = (\bar{Q}, \bar{\Sigma}, \bar{F})$ a *permutation machine* if $(\bar{Q})\bar{F}_{\bar{\sigma}} = \bar{Q}$ for all $\bar{\sigma} \in \bar{\Sigma}$. Thus each input gives a permutation of the state machine.

Theorem 3.4.1

Let $\mathcal{M}$ be a permutation–reset machine then

$$\mathcal{M} \leq \mathcal{N}^{\cdot} \omega \mathcal{P}$$

where $\mathcal{N}$ is a reset machine and $\mathcal{P}$ is a permutation machine.

Proof Let $\mathcal{M} = (Q, \Sigma, F)$ and put

$$\Theta = \{\sigma \in \Sigma \mid (Q)F_\sigma = Q\}$$

and

$$\Xi = \{\sigma \in \Sigma \mid |(Q)F_\sigma| = 1\}.$$

Define G to be the subgroup of $\mathbf{S}(\mathcal{M})$ generated by Θ and put $\mathcal{P} = (G, \Sigma, \bar{F})$ where

$$[\alpha]\bar{F}_\theta = [\alpha\theta] \quad \text{for } \theta \in \Theta, \alpha \in \Theta^*$$

$$[\alpha]\bar{F}_\xi = [\alpha] \quad \text{for } \xi \in \Xi, \alpha \in \Theta^*.$$

Let $\mathcal{N} = (Q, G \times \Sigma, F^*)$ where $qF^*_{(g,\xi)} = qF_\alpha F_\xi (F_\alpha)^{-1}$ if $g = [\alpha] \in G$, $\alpha \in \Theta^*$, $\xi \in \Xi$, $q \in Q$. Now F_α is a permutation of Q and so $(F_\alpha)^{-1}$ is defined, furthermore $|(Q)F^*_{(g,\xi)}| = |(Q)F_\alpha F_\xi (F_\alpha)^{-1}| = 1$ since $(Q)F_\alpha = Q$ and $|(Q)F_\xi| = 1$, and thus $\mathcal{N}$ is a reset machine. The state machine $\mathcal{N}^{\cdot}$ consists of the state machine $\mathcal{N}$ with the identity map 1_Q adjoined. We thus adjoin a *new* symbol Λ to the set $G \times \Sigma$ and $\mathcal{N}^{\cdot} = (Q, (G \times \Sigma) \cup \{\Lambda\}, F^{**})$ where

$$qF^{**}_{(g,\xi)} = qF^*_{(g,\xi)}$$

for $q \in Q$, $g \in G$, $\xi \in \Xi$ and

$$qF^{**}_\Lambda = q$$

for $q \in Q$. Now define

$$\omega : G \times \Sigma \to G \times \Sigma \cup \{\Lambda\}$$

by

$$\omega(g, \sigma) = \begin{cases} \Lambda & \text{if } \sigma \in \Theta \\ (g, \sigma) & \text{if } \sigma \in \Xi. \end{cases}$$

We may now form the cascade product $\mathcal{N}^{\cdot} \omega \mathcal{P}$; the state mapping of this machine will be denoted by F^ω. The covering map $\phi : Q \times G \to Q$

is defined by

$$\phi(q, g) = qF_\alpha$$

where $g = [\alpha] \in G$, $q \in Q$.

We must now establish the covering properties for ϕ. First ϕ is clearly surjective as $G \neq \emptyset$ and F_α is a permutation of Q. Now let $\sigma \in \Theta$ and $(q, g) \in Q \times G$. If $g = [\alpha]$ where $\alpha \in \Theta^*$, then

$$(\phi(q, [\alpha]))F_\sigma = (qF_\alpha)F_\sigma = qF_{\alpha\sigma} = \phi(q, [\alpha\sigma])$$

since $\alpha\sigma \in \Theta^*$. Hence

$$\begin{aligned}\phi((q, [\alpha])F^\omega_\sigma) &= \phi((qF^*_{([\alpha],\sigma)}, [\alpha]\bar{F}_\sigma) \\ &= \phi((q, [\alpha\sigma])).\end{aligned}$$

If $\sigma \in \Xi$ and $(q, g) \in Q \times G$ with $g = [\alpha]$ for $\alpha \in \Theta^*$ then

$$(\phi(q, [\alpha]))F_\sigma = (qF_\alpha)F_\sigma = qF_\sigma.$$

Also

$$\begin{aligned}\phi((q, [\alpha])F^\omega_\sigma) &= \phi(qF^{**}_{([\alpha],\sigma)}, [\alpha]\bar{F}_\sigma) \\ &= \phi(qF^*_{([\alpha],\sigma)}, [\alpha]) \\ &= \phi(qF_\alpha F_\sigma (F_\alpha)^{-1}, [\alpha]) \\ &= qF_\alpha F_\sigma (F_\alpha)^{-1} F_\alpha \\ &= qF_\alpha F_\sigma = qF_\sigma.\end{aligned}$$

Hence in all cases $(\phi(q, [\alpha]))F_\sigma \subseteq \phi((q, [\alpha])F^\omega_\sigma)$. □

This result can be interpreted in the language of transformation semigroups and it is then possible to generalize it slightly. First we have:

Corollary 3.4.2
$\mathbf{TS}(\mathcal{M}) \leq \mathbf{TS}(\mathcal{N})^c \circ \mathcal{G}$

where

$$G = \mathbf{S}(\mathcal{M}).$$

Proof This follows from 3.4.1 using the fact that $\mathcal{P} \leq \mathbf{SM}(\mathcal{G})$. (*See* 3.4.4.) □

Theorem 3.4.3
Let $\mathcal{A} = (Q, S)$ be a transformation monoid and suppose that G is the maximal subgroup of S. Then

$$\mathcal{A} \leq (Q, S\backslash G)^c \circ \mathcal{G}.$$

Proof We use the same covering map, namely $\phi : Q \times G \to Q$ defined in 3.4.1, so that $\phi(q, g) = qg$ $(q \in Q, g \in G)$. Now let $s \in S$ and we have two cases. If $s \in G$ define $f_s : G \to (S\backslash G) \cup 1_Q$ by $f_s(g) = 1_Q$ for $g \in G$. If $s \in S\backslash G$ define $f_s : G \to (S\backslash G) \cup 1_Q$ by $f_s(g) = gsg^{-1}$ for $g \in G$. We note that $gsg^{-1} \in S\backslash G$ if $s \in S\backslash G$, for letting $gsg^{-1} = h \in G$ gives $s = g^{-1}hg \in G$.

Next we form the element $k_s \in G$ defined by

$k_s = s$ if $s \in G$

$k_s = 1$ if $s \in S\backslash G$.

Now we show that the pair of elements (f_s, k_s) will cover s with respect to ϕ. Choose any $q \in Q$, $g \in G$; then

$$\phi(q, g) \cdot s = qgs$$

and

$$\begin{aligned}\phi((q, g) \cdot (f_s, k_s)) &= \begin{cases}\phi(q, gs) & \text{if } s \in G \\ \phi(qgsg^{-1}, g) & \text{if } s \in S\backslash G\end{cases} \\ &= qgs \quad \text{in both cases.}\end{aligned}$$

Thus $\phi(q, g) \cdot s \subseteq \phi((q, g) \cdot ((f_s, k_s))$ for all $q \in Q$, $g \in G$, $s \in S$. □

If we now turn our attention to permutation machines we have the following result.

Theorem 3.4.4
Let $\mathcal{M} = (Q, \Sigma, F)$ be a permutation machine, then $\mathcal{M} \leq \mathbf{SM}(\mathcal{G})$ where G is a finite group.

Proof Let G be the group of all permutations on the set Q. Consider the state machine $\mathbf{SM}(\mathcal{G}) = (G, G, F')$ where

$$g_1 F'_g = g_1 g \quad \text{for } g, g_1 \in G.$$

Define a covering function $\phi : G \to Q$ as follows, let $q_0 \in Q$ be a fixed element of Q, put $\phi(g) = q_0 F_\alpha$ where $g = [\alpha] \in G$. Now ϕ is surjective. Let $\rho : \Sigma \to G$ be defined by $\rho(\sigma) = [\sigma]$ for $\sigma \in \Sigma$.

Given

$$\begin{aligned}\sigma \in \Sigma, g \in G, \phi(g)F_\sigma = q_0 F_\alpha F_\sigma \quad &\text{where } g = [\alpha] \\ &= q_0 F_{\alpha\sigma} \\ &\subseteq \phi([\alpha\sigma]) \\ &= \phi([\alpha]F'_{[\sigma]}) \\ &= \sigma(gF'_{\rho(\sigma)})\end{aligned}$$

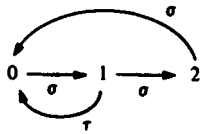
as required. □

Since the group G may be rather larger than $\mathbf{S}(\mathcal{M})$ we may find the next result more useful.

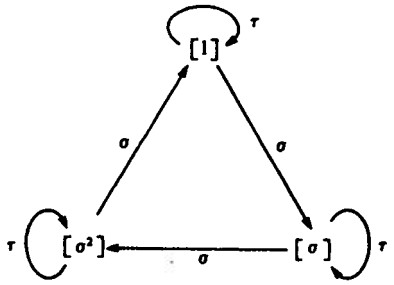
Theorem 3.4.5
Let $\mathcal{M}$ be a permutation machine, then
$\mathbf{TS}(\mathcal{M}) \leq \mathcal{Q}^{\cdot} \circ \mathcal{G}$ where $\mathcal{M} = (Q, \Sigma, F)$ and $G = \mathbf{S}(\mathcal{M})$.

Proof Apply corollary 3.4.2; then $\mathbf{TS}(\mathcal{N}) = (Q, \varnothing)$ and $\mathbf{TS}(\mathcal{N})^{\cdot} = \mathcal{Q}^{\cdot}$. □

Example 3.6
Consider the state machine $\mathcal{M} = (Q, \Sigma, F)$ defined by



where $\Sigma = \{\sigma, \tau\}$, $Q = \{0, 1, 2\}$. This is a permutation–reset machine. In the notation of 3.4.1 $\mathcal{P} = (\mathbf{Z}_3, \Sigma, \bar{F})$ is given by



where $\mathbf{Z}_3 = \{[1], [\sigma], [\sigma^2]\}$.
$\mathcal{N}^{\cdot} = (Q, (\mathbf{Z}_3 \times \Sigma) \cup \{\Lambda\}, F^{**})$ is a reset machine given by the table

	0	1	2
Λ	0	1	2
$([1], \sigma)$	∅	∅	∅
$([1], \tau)$	∅	0	∅
$([\sigma], \sigma)$	∅	∅	∅
$([\sigma], \tau)$	∅	2	∅
$([\sigma^2], \sigma)$	∅	∅	∅
$([\sigma^2], \tau)$	∅	1	∅

In transformation semigroups

$$\mathbf{TS}(\mathcal{M}) \leq \mathbf{TS}(\mathcal{N})^{\cdot} \circ \mathbf{TS}(\mathcal{P}) \leq (\bar{\mathbf{2}} \times \bar{\mathbf{2}})^{\cdot} \circ \mathbf{Z}_3$$

since $\mathbf{TS}(\mathcal{N}) \leq \bar{\mathbf{2}} \times \bar{\mathbf{2}}$ and $\mathbf{TS}(\mathcal{P}) \leq \mathbf{Z}_3$.

3.5 Group machines

The last result brings us into the world of group machines. As we have seen, given any finite group G we can construct a transformation semigroup $\mathcal{G} = (G, G)$ and a state machine $\mathbf{SM}(\mathcal{G}) = (G, G, F')$ where $g_1 F'_g = g_1 g$ for $g, g_1 \in G$. This state machine is called the *state machine defined by* G.

For the moment it is more natural to use the transformation group terminology. Suppose that H is a subgroup of G, we define a partition on G by using the set of distinct right cosets $\{Hg \mid g \in G\}$. It is immediate that this is an admissible partition since $Hg \cdot g_1 = Hgg_1 \in \pi$ for $Hg \in \pi$ and $g_1 \in G$. Consequently we can construct the quotient transformation semigroup $\mathcal{G}/\pi$. This has state set equal to π or in another terminology G/H (although we should not assume that H is a *normal* subgroup of G/H: we are just regarding G/H as a set, the set of right cosets of H in G, and not as a group). The semigroup of G/π consists of all the distinct mappings of G/H into G/H induced by elements of G. Define a relation $\sim$ on G by

$$g_1 \sim g_2 \Leftrightarrow Hgg_1 = Hgg_2 \quad \text{for all } Hg \in G/H.$$

So

$$g_1 \sim g_2 \Rightarrow g_1 \in g^{-1}Hgg_2 \quad \text{for all } g \in G$$

$$\in \bigcap_{g \in G} g^{-1}Hg \cdot g_2.$$

Put $H^G = \bigcap_{g \in G} g^{-1}Hg$ then H^G is a subgroup of G and is clearly normal in G.

Lemma 3.5.1
If $H \subseteq G$ and $\sim$ is defined on G by

$$g_1 \sim g_2 \Leftrightarrow Hgg_1 = Hgg_2 \quad \text{for all } Hg \in G/H$$

then $\sim$ is an equivalence relation and the partition of $\sim$ equals the partition consisting of the right cosets of $H^G = \bigcap_{g \in G} g^{-1}Hg$ in G.

Proof Clearly if $g_1, g_2 \in G$ and $g_1 \sim g_2$ then $g_1 \in H^G g_2$ and so $H^G g_1 = H^G g_2$. Now let $H^G g_1 = H^G g_2$, then $g_1 \in H^G g_2$, so $g_1 \in g^{-1}Hgg_2$

for all $g \in G$ and thus $gg_1 \in Hgg_2$. Therefore $gg_1 = hgg_2$ for some $h \in H$ and $Hgg_1 = Hhgg_2 = Hgg_2$, giving $g_1 \sim g_2$. □

We can now write $\mathcal{G}/\pi$ as the transformation group $(G/H, G/H^G)$ with the action of the *group* G/H^G on the *set* G/H given by

$$Hg * H^G g_1 = Hgg_1 \quad \text{for } g, g_1 \in G.$$

It is clear that this action is well-defined and faithful. We now proceed to the central result of this section, namely:

Theorem 3.5.2
Let H be a subgroup of the finite group G, then
$\mathcal{G} \leq \mathcal{H} \circ (G/H, G/H^G)$.

Proof Let us fix the coset representatives so that

$$G = Hg_1 \cup Hg_2 \cup \ldots \cup Hg_n \quad \text{(say)}.$$

Define a function $\phi : H \times G/H \to G$ by $\phi(h, Hg_i) = hg_i$ for $h \in H$, $Hg_i \in G/H$; this is clearly surjective. Given $g \in G$ we must find a pair $(f_g, s_g) \in H^{G/H} \times G/H^G$, that is a map $f_g : G/H \to H$ and an element $s_g \in G/H^G$, which covers g.

First we put $s_g = H^G g$.

If $Hg_i \in G/H$ then $Hg_i g = Hg_k$ for some $1 \leq k \leq n$ and $g_i g = h' g_k$ for some $h' \in H$. Define $f_g(Hg_i) = h'$ where $g_i g = h' g_k$. The choice of h is unique because we have fixed the representatives $g_1, \ldots, g_n$, consequently we can define the function $f_g : G/H \to H$ as required.

Now for $(h, Hg_i) \in H \times G/H$ and $g \in G$ we get

$$\phi(h, Hg_i)g = hg_i g$$

and

$$\begin{aligned} \phi((h, Hg_i)(f_g, s_g)) &= \phi(hf_g(Hg_i), Hg_i H^G g) \\ &= \phi(hh', Hg_k) \end{aligned}$$

where $g_i g = h' g_k$ and $Hg_i g = Hg_k$,

$$= hh' g_k = hg_i g.$$

Therefore ϕ is a covering. □

Notice that if $H = \{h_1, \ldots, h_m\}$ and $K = \{g_1, \ldots, g_n\}$ then the collection of subsets $\tau = \{h_1 K, h_2 K, \ldots, h_m K\}$ is a partition, for if $h_i K \cap h_j K \neq \varnothing$ then $x \in h_i K \cap h_j K \Rightarrow x = h_i g_l = h_j g_p$ for $g_l, g_p \in K$ so $Hg_l \cap Hg_p \neq \varnothing$ and thus $g_l = g_p$ and $h_i = h_j$.

This partition τ has the property that $\pi \cap \tau = 1_G$ for $h_i K \cap Hg_k = \{h_i \cdot g_k\}$ and there are $n \cdot m$ such distinct singletons. The order of

$G = |H| \cdot |G/H| = m \cdot n$ and so the partitions intersect in the identity. Consequently we could also use theorem 3.3.1 and corollary 3.3.2 which would eventually lead to the above result.

Corollary 3.5.3
If H is a normal subgroup of G then
$\mathcal{G} \leq \mathcal{H} \circ \mathcal{G}/\mathcal{H}$.

Proof This follows because G/H is now a group and $H^G = H$, therefore

$$(G/H, G/H^G) = (G/H, G/H) = \mathcal{G}/\mathcal{H}. \qquad \square$$

Let G be a finite group, then G possesses a composition series $G = G_n \supset G_{n-1} \supset \ldots \supset G_1 \supset G_0 = \{1\}$ where G_{i-1} is a normal subgroup of G_i and G_i/G_{i-1} is a simple group, for $i = 1, \ldots, n$. Using this fact and applying corollary 3.5.3 repeatedly we obtain the following important result:

Theorem 3.5.4
Let G be a finite group. There exist simple groups $K_1, \ldots, K_n$ such that

$$\mathcal{G} \leq \mathcal{K}_1 \circ \ldots \circ \mathcal{K}_n$$

and

$$\mathcal{K}_i \leq \mathcal{G} \quad \text{for } i = 1, \ldots, n.$$

Proof Clearly $\mathcal{G} \leq \mathcal{G}_{n-1} \circ \mathcal{G}_n/\mathcal{G}_{n-1}$ where $\mathcal{G}_n/\mathcal{G}_{n-1}$ is simple, so we put $\mathcal{K}_n = \mathcal{G}_n/\mathcal{G}_{n-1}$ and note that the canonical epimorphism of groups $\phi : G_n \to G_n/G_{n-1}$ is a covering $\mathcal{K}_n \leq \mathcal{G}_n$. Now $\mathcal{G}_{n-1} \leq \mathcal{G}_{n-2} \circ \mathcal{G}_{n-1}/\mathcal{G}_{n-2}$ and as before if we put $\mathcal{K}_{n-1} = \mathcal{G}_{n-1}/\mathcal{G}_{n-2}$ it is clear that $\mathcal{K}_{n-1} \leq \mathcal{G}_{n-1} \leq \mathcal{G}_n$ and $\mathcal{G} \leq \mathcal{G}_{n-2} \circ \mathcal{K}_{n-1} \circ \mathcal{K}_n$. Continuing in this way completes the process. □

Recapping the results of these last two sections we have established:

Theorem 3.5.5
Let $\mathcal{M}$ be a permutation–reset machine, then
$\mathbf{TS}(\mathcal{M}) \leq \mathcal{A} \circ \mathcal{A}_1 \circ \ldots \circ \mathcal{A}_m$
where $\mathcal{A}$ is of the form $\prod^k \tilde{2}$ and $\mathcal{A}_i = \mathcal{K}_i$ with $\mathcal{K}_i$ a simple group such that $\mathcal{K}_i \leq \mathcal{G}$, where $G = \mathbf{S}(\mathcal{M})$.

Proof First we use corollary 3.4.2 to obtain

$$\mathbf{TS}(\mathscr{M}) \leq \mathbf{TS}(\mathscr{N})^{\cdot} \circ \mathscr{G}$$

where $G = \mathbf{S}(\mathscr{M})$. Now $\mathbf{TS}(\mathscr{N})^{\cdot} \leq (\overline{Q, \varnothing})^{\cdot} \leq \prod^{k} \bar{2}^{\cdot}$ where $k = |Q| - 1$ and Q is the state set of $\mathscr{M}$ by example 3.3.

Now $\mathscr{G} \leq \mathscr{K}_1 \circ \ldots \circ \mathscr{K}_n$ where each K_i is a simple group such that $\mathscr{K}_i \leq \mathscr{G}$ for $i = 1, \ldots, n$. □

From chapter 2, the statement $\mathscr{K}_i \leq \mathscr{G}$ is the same as K_i divides G.

3.6 Connected transformation semigroups

A *connected* transformation semigroup $\mathscr{A} = (Q, S)$ is a transformation semigroup such that given q, $q_1 \in Q$ there exists $s \in S$ satisfying $q_1 = qs$. If $\mathscr{A} = (Q, G)$ is a transformation group then $\mathscr{A}$ is connected if and only if G is a transitive permutation group acting on Q.

A state machine $\mathscr{M}$ is *connected* if its transformation semigroup $\mathbf{TS}(\mathscr{M})$ is connected.

Connectedness is a useful property which we will now investigate.

A connected reset machine is closed and a connected permutation machine can be covered by the group machine defined by the group of the original machine rather than the group of all permutations of the states.

Theorem 3.6.1

Let $\mathscr{M} = (Q, \Sigma, F)$ be a connected machine.

(i) If $\mathscr{M}$ is a reset machine then $\mathbf{TS}(\mathscr{M}) = (\overline{Q, \varnothing})$.

(ii) If $\mathscr{M}$ is a permutation machine then

$$\mathscr{M} \leq \mathbf{SM}(\mathscr{G})$$

where $G = \mathbf{S}(\mathscr{M})$ and

$$\mathbf{TS}(\mathscr{M}) \leq \mathscr{G}.$$

(iii) If $\mathscr{A} = (Q, G)$ is a connected transformation group then $\mathscr{A} \leq \mathscr{G}$.

Proof (i) is immediate.

(ii) Consider the function $\phi : G \to Q$ defined by

$$\phi(g) = q_0 F_\alpha$$

where q_0 is a fixed element of Q and $[\alpha] = g \in \mathbf{S}(\mathscr{M})$. (We used this function in the proof of 3.4.4 but with G equal to the group of all permutations of Q.) Then ϕ is surjective because of the connectivity of

$\mathscr{M}$. The rest of the proof of 3.4.4 can be adapted to this situation and we see that ϕ is thus a covering.

(iii) This is similar to (ii). □

We now examine connected transformation groups in more detail. These are transitive permutation groups. Suppose that $\mathscr{A} = (Q, G)$ is a connected transformation group and let $\pi = \{H_i\}_{i \in I}$ be a non-trivial admissible partition on $\mathscr{A}$. Consider a π-block H_i, then, given $g \in G$, we have

$$H_i \cdot g \subseteq H_j \quad \text{for some } j \in I.$$

Suppose that $q \in H_i \cap H_i g$ then $q \in H_i \cap H_j \Rightarrow H_i = H_j$. Therefore $\pi = \{H_i\}_{i \in I}$ is a primitive block system of the permutation group (Q, G). Consequently G is an imprimitive permutation group as G is transitive and π is non-trivial.

Let $H_1 = \{q_1, \ldots, q_r\}$ be a π-block and define $K = \{q \in G \mid H_1 g = H_1\}$. Then K is a subgroup of G. Suppose that $Kg_1, \ldots, Kg_s$ is a set of distinct right cosets of K in G such that

$$G = \bigcup_{j=1}^{s} Kg_j \quad \text{and} \quad Kg_j \cap Kg_l = \varnothing \quad \text{if } j \neq l.$$

Define

$$L_1 = \{q_1 g_1, q_1 g_2, \ldots, q_1 g_s\}$$
$$L_2 = \{q_2 g_1, q_2 g_2, \ldots, q_2 g_s\}$$
$$\vdots$$
$$L_r = \{q_r g_1, q_r g_2, \ldots, q_r g_s\}$$

and put $\tau = \{L_1, L_2, \ldots, L_r\}$. We note that τ is a partition of Q for if $L_i \cap L_j \neq \varnothing$ then we can find $q_i g_l = q_j g_m$ for suitable i, j, l, m. Then $q_i = q_j (g_m g_l^{-1}) \in H_1 \cap H_1 (g_m g_l^{-1})$ and so $H_1 = H_1 (g_m g_l^{-1})$ which implies that $g_m g_l^{-1} \in K$ and so $Kg_m = Kg_l$, $m = l$ and $i = j$. Furthermore if $q \in Q$ then there exists $g \in G$ such that $q = q_1 g$. Now $g = kg_j$ for some $k \in K$ and $j \in \{1, \ldots, s\}$, thus $q = q_1 k g_j$. But $q_1 k = q_i$ for some $i \in \{1, \ldots, r\}$ and so $q = q_i g_j \in L_i$. This establishes that τ is a partition of Q.

Now let $H_i \in \pi$ and suppose that $q \in H_i$. Then there exists $g \in G$ such that $q = q_1 g$ and as $g = kg_m$ for some $m \in \{1, \ldots, s\}$ we see that $q = g_1 k g_m \in H_1 g_m$. Therefore $H_1 g_m \subseteq H_i$ and conversely $H_i \subseteq H_1 g_m$. Hence $H_i = H_1 g_m$ and we may write π as $\{H_1 g_1, \ldots, H_1 g_s\}$. Now $H_1 g_m \cap L_n = \{q_n g_m\}$ and so $\pi \cap \tau = 1_Q$. We can now apply the procedure of 3.3.1 and 3.3.2 to this situation. First we calculate the transformation semigroup $\mathscr{A}/\langle \pi \rangle$. Each permutation $g \in G$ induces a permutation of the π-block

which we will denote by $\bar{g}$. These mappings form a permutation group $\bar{G}$ on the set π and the mapping $\theta: G \to \bar{G}$ defined by $\theta(g) = \bar{g}$ for $g \in G$ is clearly a homomorphism. The kernel, N of θ, is the subgroup of all permutations in G that fix all the π-blocks, that is $g \in N$ if and only if $H_i g = H_i$ for all $i \in I$. It is fairly easy to see that

$$\mathscr{A}/\langle\pi\rangle \cong (\pi, G/N).$$

Instead of now calculating the transformation semigroup corresponding to the state machine $\mathscr{N}$ of 3.3.1 we will deduce the required result directly. First define

$$K_{H_1} = \{g \in K \mid q_i g = q_i \quad \text{for all } i \in \{1, \ldots, r\}\}.$$

Then K_{H_1} is a normal subgroup of K. Consider the transformation group $(\tau, K/K_{H_1})$ with the operation defined by $L_n \cdot K_{H_1} \cdot k = L_p$ where $q_n k = q_p$ and $n, p \in \{1, \ldots, r\}$ and $k \in K$. This operation is faithful for if $L_n K_{H_1} k = L_n K_{H_1} k'$ for all $n \in \{1, \ldots, r\}$ then

$$q_n k = q_n k' \quad \text{for all } n \in \{1, \ldots, r\}$$

and so

$$k(k')^{-1} \in K_{H_1} \quad \text{and} \quad K_{H_1} k = K_{H_1} k'.$$

We now establish the following result:

$$(Q, G) \leq (\tau, K/K_{H_1}) \circ (\pi, G/N).$$

Let $\phi: \tau \times \pi \to Q$ be defined by

$$\phi(L_n, H_1 g_m) = q_n g_m \quad \text{for } L_n \in \tau, H_1 g_m \in \pi.$$

Given $g \in G$ define the pair $(f_g, Ng) \in (K/K_{H_1})^{\pi} \times G/N$ by putting $f_g(H_1 g_m) = K_{H_1} k$ where $g_m g = k g_i$ for a unique $k \in K$ and $i \in \{1, \ldots, s\}$. Then

$$\begin{aligned}\phi(L_n f_g(H_1 g_m), H_1 g_m N g) &= \phi(L_n K_{H_1} k, H_1 g_m g) \\ &= \phi(L_p, H_1 g_i) \\ &\quad \text{where } q_p = q_n k \text{ and } g_m g = k g_i \\ &= q_p g_i \\ &= q_n k g_i \\ &= q_n g_m g.\end{aligned}$$

Furthermore $\phi(L_n, H_1 g_m) \cdot g = q_n g_m g$ and so ϕ is a covering map. We restate this result in the following way.

Theorem 3.6.2

(Dilger [1976]) Let $\mathscr{A} = (Q, G)$ be a connected transformation group and π a non-trivial admissible partition on Q. Then

$$\mathscr{A} \leq (\tau, K/K_{H_1}) \circ (\pi, G/N).$$

Corollary 3.6.3

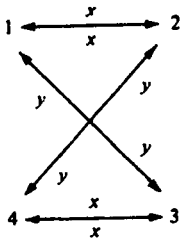
$$\mathscr{A} \leq \mathscr{K}/\mathscr{K}_{H_1} \circ \mathscr{G}/\mathscr{N}.$$

Proof Notice that both $(\tau, K/K_{H_1})$ and $(\pi, G/N)$ are connected transformation groups and so we may apply theorem 3.6.1(iii). □

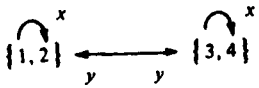
Another useful fact is that both $\mathscr{G}/\mathscr{N} \leq \mathscr{G}$ and $\mathscr{K}/\mathscr{K}_{H_1} \leq \mathscr{G}$ by 2.4.2.

Example 3.7

Consider the following state machine



This yields a transformation group which is connected and possesses a non-trivial admissible partition, namely $\pi = \{\{1, 2\}, \{3, 4\}\}$. We will apply theorem 3.6.2. First we construct $\mathscr{A}/\langle\pi\rangle$, this is given by



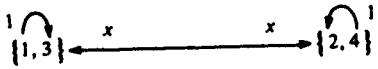
which is isomorphic to $\mathbf{Z}_2$.

Now let $H_1 = \{1, 2\}$ and consider $K = \{g \in G \mid H_1 g = H_1\}$. The only inputs that can preserve H_1 in this way are the identity and powers of x. Since $x^2 = 1$ we have $K = \{1, x\}$. Put $G = Kg_1 \cup Kg_2 \cup \ldots \cup Kg_s$, with $g_1 = 1$. Clearly $g_2 = y$ is another coset representative and so

$$L_1 = \{1 \cdot 1, 1 \cdot y, 1 \cdot g_3, \ldots, 1 \cdot g_s\} = \{1, 3, \ldots.\}$$

$$L_2 = \{2 \cdot 1, 2 \cdot y, 2 \cdot g_3, \ldots, 2 \cdot g_s\} = \{2, 4, \ldots.\}$$

and this exhausts $Q = \{1, 2, 3, 4\}$ and so there are two cosets of K in G and $\tau = \{\{1, 3\}, \{2, 4\}\}$. Further $K_{H_1} = \{1\}$ and $(\tau, K/K_{H_1})$ is given by



again isomorphic to $\mathbf{Z}_2$. Thus $(Q, G) \leq \mathbf{Z}_2 \circ \mathbf{Z}_2$.

We have not actually calculated G but it is easily seen to be isomorphic to the Klein four-group V. Using a combination of theorems 3.6.1(ii) and 3.5.5 we can obtain the same decomposition, namely

$$(Q, G) = (4, V) \leq \mathscr{V}$$

and

$$\mathscr{V} \leq \mathscr{Z}_2 \circ \mathscr{Z}_2$$

since $\{1\} \subset \mathbf{Z}_2 \subset V$ is a composition series with $V/\mathbf{Z}_2 \cong \mathbf{Z}_2$. However this approach does require a knowledge of the group G of the original machine. This may be large and cumbersome to evaluate and there are distinct advantages in using theorem 3.6.2. In Dilger [1976] an example illustrating this point can be found. If either of the transformation groups $(\tau, K/K_{H_1})$ or $(\pi, G/N)$ are imprimitive the theorem can be applied further.

3.7 Automorphism decompositions

Let $\mathscr{A} = (Q, S)$ be a transformation semigroup and let $\mathbf{Aut}_S Q$ denote the set of all automorphisms of $\mathscr{A}$. Thus $\gamma : Q \to Q$ belongs to $\mathbf{Aut}_S Q$ if and only if γ is bijective and $\gamma(qs) = \gamma(q) \cdot s$ for all $q \in Q$, $s \in S$. The identity function $1_Q : Q \to Q$ always belongs to $\mathbf{Aut}_S Q$, it may be the only element. The pair $\mathscr{A}^* = (Q, \mathbf{Aut}_S Q)$ is a transformation group under the action defined by

$$q\gamma = \gamma(q) \quad \text{for } q \in Q, \gamma \in \mathbf{Aut}_S Q,$$

where the set $\mathbf{Aut}_S Q$ is a group under the operation $*$ defined by $(\gamma * \gamma_1)(q) = \gamma_1(\gamma(q))$ for all $q \in Q$; $\gamma, \gamma_1 \in \Gamma$.

Theorem 3.7.1

Let $\mathscr{A} = (Q, S)$ be a transformation semigroup and suppose that $\Gamma = \mathbf{Aut}_S Q$. If $\Gamma \neq \{1\}$ and $\pi = \{H_i\}_{i \in I}$ is a set of distinct orbits of Q under Γ then π is an admissible partition.

Proof Let $H_i \in \pi$, and suppose that $q \in H_i$. If $s \in S$ then $qs \in H_j$ for some $j \in I$. Now let $q' \in H_i$ then $q' = \gamma(q)$ for some $\gamma \in \Gamma$ and $q's = \gamma(q)s = \gamma(qs)$ which shows that $q's \in H_j$. Hence $H_i s \subseteq H_j$ as required. □

Corollary 3.7.2

If $\mathscr{A} = (Q, S)$ is an irreducible transformation semigroup and $\mathbf{Aut}_S Q \neq \{1\}$ then $\mathscr{A}^*$ is a connected transformation group.

Proof The set of distinct orbits must consist of just one orbit, that is $\mathbf{Aut}_S Q$ is transitive on Q. □

From exercise 3.5, $\mathbf{Aut}_S Q$ is a primitive permutation group on Q.

Theorem 3.7.3

Let $\mathscr{A} = (A, S)$ be an irreducible transformation semigroup with $\mathbf{Aut}_S Q \neq \{1\}$ then $\mathscr{A} \leq \mathbf{Z}_p$ where p is a prime number and $|Q| = p$.

Proof We have established that $\mathscr{A}^* = (Q, \Gamma)$ is a transitive permutation group where $\Gamma = \mathbf{Aut}_S Q$. Let H_1 be a subgroup of Γ and suppose that $\{1\} \neq H_1 \subsetneq \Gamma$. Define the relation $\sim$ on Q by $q \sim q'$ if and only if $q' = h(q)$ for some $h \in H_1$. This equivalence relation defines an admissible partition π on $\mathscr{A}$ since $q \sim q'$ and $s \in S$ implies that $q's = h(q)s = h(qs)$ and so $qs \sim q's$. But $\mathscr{A}$ is irreducible and so $\pi = 1_Q$ or $\pi = \{Q\}$. The first conclusion leads to $H_1 = \{1\}$ and is excluded. Either Γ has no proper subgroups apart from $\{1\}$ and is thus cyclic of prime order or $\mathscr{A}_1^* = (Q, H_1)$ is a transitive permutation group. Now let H_2 be a proper non-trivial subgroup of H_2 and repeat the process. Again either $H_2 = \{1\}$ or $\mathscr{A}_2^* = (Q, H_2)$ is a transitive permutation group. Eventually we reach the position where there exists a transitive permutation group $\mathscr{A}_r^* = (Q, \mathbf{Z}_p)$ where p is a prime number. We construct the covering $\mathscr{A} \leq \mathbf{Z}_p$ as follows. Let $q_0 \in Q$ be fixed and define $\phi : \mathbf{Z}_p \to Q$ by

$$\phi(g) = q_0 g \quad \text{for } g \in \mathbf{Z}_p.$$

This is surjective since $\mathbf{Z}_p$ is transitive on Q. For $s \in S$ we define the element $h_s \in \mathbf{Z}_p$ by $q_0 s = q_0 h_s$ and then $\phi(g) \cdot s = (q_0 g)s = g(q_0 s) = g(q_0 h_s) = g(h_s(q_0)) = \phi(gh_s) = \phi(h_s g) = \phi(g * h_s)$ as $\mathbf{Z}_p$ is abelian.

Thus $\mathscr{A} \leq \mathbf{Z}_p$ with h_s covering s. Finally $|Q| \leq p$ and if $q \in Q$ then $\{g \in \mathbf{Z}_p \mid qg = q\} = \{1\}$ and so $Q = q\mathbf{Z}_p$ implies that $|Q| = p$. (Note that $|Q| \neq 1$ as $\mathbf{Aut}_S Q \neq \{1\}$.) □

This last result is a special case of the following theorem (Krohn, Langer & Rhodes [1967]).

Theorem 3.7.4

Let $\mathscr{A} = (Q, S)$ be a connected transformation semigroup and suppose that $\Gamma = \mathbf{Aut}_S Q$. If $\Gamma \neq \{1\}$ then

$$\mathscr{A} \leq \Gamma \circ \mathscr{A}/\langle \pi \rangle$$