form and as each step preserves ambiguity G_1 is unambiguous if G is unambiguous. By Corollary 1.2, the result then follows.

Appendix 3

Consider the effect of a particular $a_{ij} = \phi$ in the left cycle removal construction. Without loss of generality choose $a_{31} = \phi$ in the third-order situation. Our original equations are:

$$\begin{aligned} X_1 &\to X_1a_{11}|X_2a_{21}|b_1 \\ X_2 &\to X_1a_{12}|X_2a_{22}|X_3a_{32}|b_2 \\ X_3 &\to X_1a_{13}|X_2a_{23}|X_3a_{33}|b_3. \end{aligned}$$

Under the construction we obtain:

$$\begin{aligned} X_1 &\to b_1Y_{11}|b_2Y_{21}|b_3Y_{31} \\ X_2 &\to b_1Y_{12}|b_2Y_{22}|b_3Y_{32} \\ X_3 &\to b_1Y_{13}|b_2Y_{23}|b_3Y_{33} \end{aligned}$$

where

$$\begin{aligned} Y_{11} &\to a_{11}Y_{11}|a_{12}Y_{21}|a_{13}Y_{31}|\epsilon \\ Y_{12} &\to a_{11}Y_{12}|a_{12}Y_{22}|a_{13}Y_{32} \\ Y_{13} &\to a_{11}Y_{13}|a_{12}Y_{23}|a_{13}Y_{33} \\ Y_{21} &\to a_{21}Y_{11}|a_{22}Y_{21}|a_{23}Y_{31} \\ Y_{22} &\to a_{21}Y_{12}|a_{22}Y_{22}|a_{23}Y_{32}|\epsilon \\ Y_{23} &\to a_{21}Y_{13}|a_{22}Y_{23}|a_{23}Y_{33} \\ Y_{31} &\to a_{31}Y_{11}|a_{32}Y_{21}|a_{33}Y_{31} \\ Y_{32} &\to a_{31}Y_{12}|a_{32}Y_{22}|a_{33}Y_{32} \\ Y_{33} &\to a_{31}Y_{13}|a_{32}Y_{23}|a_{33}Y_{33}|\epsilon. \end{aligned}$$

If we now applied the simple rule $a_{ij} = \phi$ implies $Y_{ij} = \phi$, we find that X_1 could not begin with b_3 after the construction, whereas before the construction it could begin with b_3. Therefore carrying through the consequence of $a_{31} = \phi$ we delete the alternatives $a_{31}Y_{1i}$, $i = 1, 2, 3$. If we now also assume $a_{32} = \phi$, we also delete $a_{32}Y_{2i}$, $i = 1, 2, 3$. However we are then left with two non-terminating rules, namely

$$Y_{31} \to a_{33}Y_{31}$$

and

$$Y_{32} \to a_{33}Y_{32}.$$

In this situation, however, we must delete these rules by defining Y_{31} and Y_{32} to be ϕ. If we did not do this X_1 and X_2 would have non-terminating alternatives, corresponding to the fact that X_1 and X_2 cannot begin with b_3 if $a_{31} = a_{32} = \phi$. This leads to the following rule:

Reduction rule: For any Y_{ij} which is non-terminating define $Y_{ij} = \phi$ and repeat the reduction until no further Y_{ij} is non-terminating.

Book Review

A Mathematical Theory of Systems Engineering, by A. Wayne Wymore, 1967; 353 pages. (London and New York: *John Wiley and Sons Limited*, 150s.)

As a theoretical text on systems engineering this book has a relatively unique feature: it uses set theory rather than classical algebra. It aims at a general theory of systems and succeeds in so far as it treats data processing automata, control and man-machine systems between the same covers. This is a refreshing standpoint for control engineers, but disappointing for those who look for a direct application. Set theory provides a weak mathematical structure which, due to its weakness, embraces many systems while allowing little scope for useful manipulation. Indeed, the author makes no claims in this respect and is happy to provide the tools, perhaps merely a language, whereby systems may be described.

The *introductory chapter* makes the point that rigour is important for systems that involve information and its communication. These cannot be tackled intuitively in the same way as systems that have physically tangible parameters. The chapter also presents an introduction to the set theory that is used in the rest of the book. Chapter 2 defines the *elements of the general theory*. Here control concepts of state space appear amid tools of automata theory such as admissible input sequences, semigroups and Turing machines.

The third chapter is on *modelling* and I consider it to be a highlight of the book. Proceeding from the premise that '. . . modelling is an art . . .' the author asserts his artistry by producing models for a wide variety of systems. These include computing elements, a widget factory, a watershed and a human-tracking experiment. The introduction of a pseudo-computer language is interesting since it underlines the fact that the aim of modelling is, usually, the design of a computer simulation program. Chapter 4 is on the *comparison of systems*. Laws of homomorphic mapping are used to generate equivalence classes as is customary in the minimisation of sequential machines. Chapter 5, *coupling of systems*, deals with cascading and feedback. Here the author is led to the assertion that a system with no input is no system. This is a pity since it excludes autonomous machines which are generally interesting systems.

Chapter 6, *subsystems and components*, uses concepts developed in connection with sequential machines by Gill, Hartmanis and Stearns for the decomposition of a system into elements. The discussion on duality in systems and the meaning of observability and controllability is worth noting. Chapter 7 is on *discrete systems* and provides an interesting link between programming and state transition tables.

The book is well organised. The mathematical and the discussion sections are clearly set out. However, even though the jacket claims that all the necessary mathematical foundations are included, a reader with no knowledge of set jargon may find it difficult to follow.

IGOR ALEKSANDER (Canterbury)