A MATHEMATICAL FORMULATION FOR A CLASS OF DISCRETE EVENT SYSTEMS

R. L. Aveyard

Bell Telephone Laboratories

In this paper a discrete event system is a dynamical system in which the term event is used to describe the occurence of a discontinuous change in the elements of the system at a point in time. Events occur at discrete intervals of time, which are not necessarily uniformly separated, and logical relationships govern the discontinuosu changes when events occur. The state space is restricted to being discrete and finite.

The types of discrete event systems, which satisfy the above requirements, include the following example. In many applications of computers to the control of complex processes, various program modules are used to perform specific tasks. These modules must be controlled, so that the logical decisions in using them and the interactions between them are coordinated to achieve the desired control. Similar control problems including start-up and shut-down, which can be represented as discrete event systems, exist in manufacturing processes, warehousing, handling of bulk materials and computer aided diagnosis. In designing a controller for such discrete event systems it is important to know whether the system is deterministic, and if it will 'hang' or cycle.

The state of a discrete event system can be represented by elements of the set $\{Y,R\}$, where Y is the set of states of all units in the system, and R is the set of schedules of future events. Because a change of state takes place when an event occurs, and the change is deterministic, there exists a state transition function p', which maps the state set $\{Y,R\}$ and the event set F into the state $\{Y,R\}$.

$$p': \quad \{Y,R\} \times F \rightarrow \{Y,R\} \tag{1}$$

This work was supported in part by the Industrial Sponsor's Group of the Control Complex Systems Research Program, Case Western Reserve University.

Not all events are able to occur at any instant in time. The events that can occur are termed enabled events, and they are determined by the current description of all the units in the system. The enabled events are not time dependent, because the state of a discrete event system remains constant between event occurrences. Denoting the set of enabled events by G, there is a mapping

$$r': \quad Y \rightarrow G \tag{2}$$

Exogenous events are scheduled by external inputs which are termed firing variables. Hence, for an exogenous event to occur it must be fired while it is enabled. The events influenced by elements of the set R (set of schedules of future events) are endogenous. Endogenous events which occur immediately after they are enabled are called transient events, while those that do not are called delayed events. Associated with each delayed event is a delay time τ, which specifies the interval of time between the enabling of a delayed event and its scheduled occurrence.

In modelling delayed events time is assumed to be quantized in terms of a basic unit (clock time), Δ seconds. As a result, the delay time associated with each delayed event can be expressed as $\tau = N\Delta$, where N is an integer. Time quantization is not applied to events, which are not delayed events. Δ should be chosen small enough to resolve any conflicts or race conditions between delayed events.

When modelling a discrete event system containing delayed events, a counting unit, which can count the passage of time increments, is incorporated into the model for each delayed event. Some exogenous events would be associated with each counting unit, and all such exogenous events are fired by a pulse every Δ seconds. Each counter is started by a transient event, which has the same enabling conditions as the associated delayed event. Another transient event, which produces the same state change as the occurrence of the delayed event would, is enabled when the associated counter has counted N ($N\Delta = \tau$). Each counting unit is disabled and reset to zero by any event occurrence which would disable its associated delayed event.

In this manner, delayed events can be replaced in a model by adding some counting units, transient events and exogenous events to the model. All exogenous events corresponding to counters would be fired by one clocked input pulse (firing variable). The necessity for a schedule of future events has been eliminated by the introduction of counting units, and so the elements of set R in the state description can be eliminated. Let X be the new set of states of the model, and let E be the new set of events, such that an element of E corresponds to a single event occurrence. The state transition function for this representation is

$$p: \quad X \times E \rightarrow X \qquad (3)$$

If H is the set of enabled events, there exists a mapping

$$r: \quad X \rightarrow H \qquad (4)$$

Now, an endogenous event occurs as soon as it is enabled, whereas an exogenous event occurs only if it is fired while it is enabled. Denoting the set of firing variables by S, the events which occur are defined by the mapping

$$q': \quad H \times S \rightarrow E \qquad (5)$$

Combining Equations (4) and (5), a new function q is defined

$$q: \quad X \times S \rightarrow E \qquad (6)$$

From Equations (3) and (6) the following model for discrete event systems is obtained

$$\Phi: \quad X' = p(X,E), \quad E = q(X,S) \qquad (7)$$

where X' represents the next state.

The finite state set X of the model Φ, can be represented by the finite dimensional space $X \subset D^k$, where $D = \{0,1\}$. If there are n states in the set X, then $k \geq \log_2 n$. Let the number of events in the model be m, that is, E has m elements. E can be represented by the space $E = D^m$, so that each dimension of E corresponds to an event. If an element of a vector in the space E has the value 1, the corresponding event has occurred, otherwise it has not occurred. In a manner similar to the representation of the state set, the set of firing variables S, can be efficiently represented by direct binary encoding.

Having defined the spaces X, E and S, a Boolean difference equation model can be developed from Equation (7). The state change for each event E_j can be represented by a Boolean vector C_j, and all the state changes for a model can be incorporated in a matrix C, where each row of C represents the state change of an event. The enabling and firing conditions for each event E_j can be embodied in the matrix expression

$$E = B_1X + B_2\overline{X} + B_3S + B_4\overline{S}$$

such that if a row of this expression has the value 0, the corresponding event occurs. B_1, B_2, B_3 and B_4 are matrices of Boolean variables. If A is a Boolean matrix and X a vector of appropriate dimension, then AX is defined by the normally implied multiplication and addition operations interpreted as 'AND' and 'OR' operations, respectively. $\overline{X}$ is the negation of X. By matching the rows of matrix C with those of expression E, a formulation for the model equivalent to Equation (7) is obtained

$$X' = C(E)\oplus X \qquad (8)$$
$$E = B_1X + B_2\overline{X} + B_3S + B_4\overline{S}$$

'EXCLUSIVE OR' operations are represented by "$\oplus$".

In this formulation, if X and S cause the ith row of E to have the value 0, then the event corresponding to that row occurs and the next state is determined by $X' = C_i \oplus X$, where C_i is the ith row of C. When a transient and an exogenous event, or two or more transient events are enabled by a given state X, two or more rows of E can have the value 0. If E does not have a row with value 0 for a given X and S, then no event occurs and the state X remains unchanged.

Boolean matrix equation models of the above form, can be investigated for determinacy, zero states and cycles. The Boolean matrix equations are quite compact and can be efficiently programmed on a digital computer. Algorithms have been derived for determining whether transient events interact, and whether transient event cycles exist. The zero states of a model have been shown to be solutions of a simple Boolean matrix equation.

BIBLIOGRAPHY

1. Aveyard, R. L. "A Theory for a Class of Discrete Event Systems", Systems Research Center, Case Western Reserve University, Cleveland, Ohio, Rep. SRC-71-14, 1971.