State Space Models and Minimization of Finite Automata

Bulletin of Osaka Prefecture University
Series A, Vol. 46, No. 1, 1997, pp. 39−45

State Space Models and Minimization of Finite Automata

Koji WATANABE*, Takeo IKAI** and Kunio FUKUNAGA**

(Received July 11, 1997)

Regarding finite automata (FAs) as discrete time dynamical systems, state space models of FAs are obtained from vectorization of states and symbols over B (= {0, 1}) and parameterization of system characteristics (state transition function etc.).

In this paper, we define the similarity relations between FAs expressed by state space models under which input-output responses of FAs are invariant. Based on the similarity relations, we first derive the subset construction and then provide an algorithm to minimize a given arbitrary deterministic finite automaton (FA).

1. Introduction

Finite Automata (FAs) are discrete time dynamical systems to alter the internal states by input symbols. For such dynamical systems over the real field R, the expression and analysis on the state space models are established in the field of dynamical systems and controls[1]. As against it, in the theory of FAs, FAs are only treated by diagrams, tables, and functions which define state transitions. But state space models of FAs as dynamical systems and analysis of FAs based on state space models are not found in the theory of FAs.

By vectorization of the states and symbols and parameterization of the state transitions and output function over $B=(\{0, 1\})$, FAs can be formulated as bilinear discrete time systems, and are represented by state space models over the boolean semiring$(B, +, \cdot)$.

As a result, the concepts such as reachability, observability can be defined for the state space models of FAs as well as linear dynamical systems over R[2,3].

Based on the state space model representation, we introduce in this paper a state transformation and its applications which conserve input-output responses.

In chapter 2, we shall show the way to construct state space models of FAs over B. Next we define the similarity relations between FAs which are derived by a state transformation[4]. We then perform by the similarity relations the subset construction[5] which is the method to transform a given nondeterministic FA (NFA) to an equivalent deterministic FA (DFA).

Minimization algorithms for DFAs are well known in the theory of FAs[5,6], but a novel minimization algorithm over the state space models is found using canonical decomposition and distinguishability[7].

In chapter 3, we shall propose a method to minimize a given DFA by the similarity relations instead of the canonical decomposition.

2. State Space Models of Finite Automata

2.1 State space models

We first introduce an algebraic system called the boolean semiring $(B, +, \cdot)$ (where $B=\{0, 1\}$) to derive state space models of FAs. Table 1 shows the addition and multiplication of boolean semiring. These are ordinary addition and multiplication in integers except $1+1=1$.

Table 1 Boolean semiring

addition	multiplication
$0+0=0$	$0 \cdot 0=0$
$0+1=1$	$0 \cdot 1=0$
$1+0=1$	$1 \cdot 0=0$
$1+1=1$	$1 \cdot 1=1$

A FA is formally represented by 5-tuple,

$$M=(Q, \Sigma, \delta, q_0, F) \tag{1}$$

where Q is the set of states, Σ is the set of input symbols, δ is the state transition function, q_0 is the initial state, and F is the set of accepting states. The number of states denoted by $|Q|$ is n and that of symbols $|\Sigma|$ is m.

* Graduate Student, Department of Computer and Systems Sceices, College of Engineering.
** Department of Computer and Systems Sceices, College of Engineering.

States and symbols are expressed as vectors over B and the parameterization of system characteristics (δ and F) are performed as follows.

1. The state $q_i \in Q$ $(i=0, \cdots, n-1)$ is vectorized to a n-dimensional unit vector $\boldsymbol{e}_i$ (only i-th component is 1). The initial state q_0 is represented by $\boldsymbol{x}_0(=\boldsymbol{e}_0)$, and the zero vector, if necessary, represents the dead state.
2. The state transition function $\delta(\bullet, a_k)$ for a_k $(\in \Sigma)$ is parameterized as a square matrix A_{a_k} of order n which is called a state transition matrix and abbreviated as A_k.
   The (i, j) element $a_{ij}^{(k)}$ of A_k is determined as,

$$a_{ij}^{(k)} = \begin{cases} 1 & (q_i \in Q_j^{(k)}) \\ 0 & (q_i \notin Q_j^{(k)}) \end{cases} \tag{2}$$

   where

$$Q_j^{(k)} = \delta(q_j, a_k) \quad (i, j=0, \cdots, n-1) . \tag{3}$$

3. The set F of accepting states is expressed by n-dimensional vector $\boldsymbol{c}$ with the element

$$c_i = \begin{cases} 1 & (q_i \in F) \\ 0 & (q_i \notin F) \end{cases} \quad (i=0, \cdots, n-1). \tag{4}$$

4. The input symbol a_k is encoded to $u_k(t)$ such that

$$u_k(t) = \begin{cases} 1 & \text{if } a_k \text{ is entered at time } t \\ 0 & \text{otherwise} \end{cases} . \tag{5}$$

According to these parameterizations, the state space model of a FA M, which is regarded as a bilinear dynamical system over B, is obtained as follows,

$$\begin{cases} \boldsymbol{x}(t+1) = \sum_{k=0}^{m-1} u_k(t)\, A_k \boldsymbol{x}(t) \\ y(t) = \boldsymbol{c}^t \boldsymbol{x}(t) \end{cases} \tag{6}$$

where $\boldsymbol{x}(0) = \boldsymbol{x}_0$, and $\boldsymbol{c}^t$ means the transposed vector of $\boldsymbol{c}$.

If the state vector $\boldsymbol{x}(t)$ over B includes unit vectors representing accepting states, $y(t)$ becomes 1, which means the acceptance of a input string. The parameter representations of M is denoted by $(\{A_k\}, \boldsymbol{c}, \boldsymbol{x}_0)$ corresponding to (1) and A_k's, $\boldsymbol{c}$ and $\boldsymbol{x}_0$ are called system matrices or system parameters.

Example 1:

To clarify the above-mentioned, we give an example to construct the state space model of FA M_1 given as state transition diagram of Fig. 1.

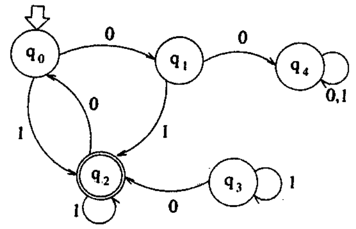

Fig. 1 DFA M_1

1. Since M_1 is 5 states, each state is vectorized to 5-dimensional unit vector.

$$q_0 \Leftrightarrow \begin{pmatrix} 1 \\ 0 \\ 0 \\ 0 \\ 0 \end{pmatrix}, q_1 \Leftrightarrow \begin{pmatrix} 0 \\ 1 \\ 0 \\ 0 \\ 0 \end{pmatrix}, \ldots, q_4 \Leftrightarrow \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 1 \end{pmatrix}$$

2. State transitions shown in the diagram(Fig. 1) are parameterized as follows:

$$A_0 = \begin{pmatrix} 0 & 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 1 \end{pmatrix}$$

$$A_1 = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{pmatrix}$$

   where A_0 is a transition matrix for input symbol 0 and A_1 is for 1.
   If there is a transition from q_j to q_i by the input 0, the (i, j) element of A_0 is 1, and otherwise 0.

3. The accepting state of M_1 is the state q_2, then $\boldsymbol{c}$ is as follows:

$$\boldsymbol{c}^t = (0 \;\; 0 \;\; 1 \;\; 0 \;\; 0).$$

4. The input symbol $a_0(=0)$ is encoded as follows:

$$u_0(t)=\begin{cases} 1 & \text{if 0 is entered at time } t \\ 0 & \text{otherwise} \end{cases}.$$

The input symbol $a_1(=1)$ is encoded similarly.

In this way, the state space model of M_1 is obtained.

2.2 Reachability and observability

Let an input string $w=a_{k_{t-1}}\cdots a_{k_0}(\in\Sigma^*)$ be added to a FA from right side symbol, then the corresponding transition matrix $A(w)$ for w is written as

$$A(w)=A_{k_{t-1}}\cdots A_{k_0} \tag{7}$$

where a_{k_t} $(\in\Sigma)$ is a input symbol at time t and A_{k_t} $(\in\{A_1,\cdots,A_m\})$ is the transition matrix for a_{k_t}.

We now define the reachability matrix R as follows:

$$R=[\boldsymbol{x}_0,\ A_0\boldsymbol{x}_0,\ \cdots,\ A(w)\boldsymbol{x}_0] \tag{8}$$

where $w\in\Sigma^*$, $|w|\leq n'$ and

$$n'=\begin{cases} n-1 & \text{for DFA} \\ 2^n-2 & \text{for NFA} \end{cases}.$$

The column $A(w)\boldsymbol{x}_0$ of R is denoted by the column label w. The i-th row of R corresponds to the state q_i. Namely, the $(i,\ w)$ element of R is 1 if there are transitions from the initial state q_0 to the state q_i by the input string w. When the i-th row is a zero row vector, q_i is called the unreachable state since there is no input string to cause transitions from q_0 to q_i. The system parameters $(\{A_k\},\boldsymbol{x}_0)$ is called completely reachable if there is no zero row vector in R.

Similarly the observability matrix O is defined as follows:

$$O=\begin{pmatrix} \boldsymbol{c}^t \\ \boldsymbol{c}^tA_0 \\ \vdots \\ \boldsymbol{c}^tA(w) \end{pmatrix} \tag{9}$$

where $w\in\Sigma^*$, $|w|\leq n'$.

The row $\boldsymbol{c}^tA(w)$ of O is denoted by the row label w. The i-th column corresponds to the state q_i. The $(w,\ i)$ element of O is 1 if there are transitions from the state q_i to the accepting states by the input string w. When the i-th column is a zero vector, q_i is called the unobservable state and there is no input string to cause transitions from q_i to accepting states.

The states which have an identical vector in the corresponding columns of O are equivalent and called indistinguishable. $(\{A_k\},\boldsymbol{c})$ is called completely observable if there is no zero column in O.

2.3 Characteristic responses

The general solution of (6) for a input string w of (7) is written as follows:

$$\boldsymbol{x}(t)=A(w)\boldsymbol{x}_0,\ \ y(t)=\boldsymbol{c}^tA(w)\boldsymbol{x}_0 \tag{10}$$

where $A(\varepsilon)=E_n$(E_n: the unit matrix of order n) and ε is the empty string.

Let

$$h(w)=\boldsymbol{c}^tA(w)\boldsymbol{x}_0, \tag{11}$$

the output sequence :

$$\{h(\varepsilon),\ \cdots,\ h(w),\ \cdots\}(\forall w\in\Sigma^*) \tag{12}$$

is called the characteristic responses for the state space model (6).

We next define the Hankel matrix H as follows:

$$H=OR. \tag{13}$$

It is clear from definition of R, O that H consists of the characteristic responses $h(w)$.

FAs which have identical characteristic responses (or Hankel matrix) are called equivalent FAs.

2.4 Similarity relations

In the linear system theory[1], the similarity transformations of system matrices,under which input-output responses of systems are invariant, are well known method for canonical decomposition.

We now introduce a somewhat generalized version of similarity transformations to state space models of FAs.

Let $M_a=(\{A_k^{(a)}\},\ \boldsymbol{c}^{(a)},\ \boldsymbol{x}_0^{(a)})$ and $M_b=(\{A_k^{(b)}\},\ \boldsymbol{c}^{(b)},\ \boldsymbol{x}_0^{(b)})$ where M_a and M_b have n_a and n_b states respectively, and let T be an $n_b\times n_a$ matrix over B.

We then define similarity relations between M_a and M_b by a transformation matrix T as follows:

$$TA_k^{(a)}=A_k^{(b)}T \tag{14}$$

$$Tx^{(a)}=x^{(b)} \tag{15}$$

$$c^{(a)\,t}=c^{(b)\,t}\,T. \tag{16}$$

Thus, T shows the state transformation between M_a and M_b such that,

$$T=[t_0,\ t_1,\ \cdots,\ t_{n_a-1}] \tag{17}$$

where t_i is the state vector of M_b which corresponds to the state q_i of M_a.

If T is a square and nonsingular matrix such as a permutation matrix, similarity relations (14)~(16) are rewritten as

$$TA_k^{(a)}\,T^{-1}=A_k^{(b)} \tag{18}$$

$$Tx^{(a)}=x^{(b)} \tag{19}$$

$$c^{(a)\,t}T^{-1}=c^{(b)\,t} \tag{20}$$

which are the similarity transformation and M_b can be obtained from M_a by a similarity matrix T.

2.5 Subset construction by similarity relations

The subset construction is a well known method to transform a given NFA to an equivalent DFA.

As an application of similarity relations, we shall show the way to perform the subset construction by similarity relations.

For a given NFA $M=(\{A_k\},\ c,\ x_0)$ of n states, let DFA $D(M)=(\{\hat{A}_k\},\ \hat{c},\ \hat{x}_0)$ be the transformed DFA by the subset construction.

Then an algorithm to derive $D(M)$ equivalent to a NFA M is as follows.

1. From the reachability matrix R_M of M, construct R'_M by taking out different columns from R_M except zero columns.

$$R'_M=[r_{a_0},\ r_{a_1},\ \cdots,\ r_{a_{\bar{n}-1}}] \tag{21}$$

2. By regarding R'_M as a transformation matrix T, system parameters $(\{\hat{A}_k\},\ \hat{c},\ \hat{x}_0)$ of $D(M)$ is obtained from $(\{A_k\},\ c,\ x_0)$ of M by use of similarity relations (14)~(16) in which M_a and M_b are replaced by $D(M)$ and M respectively.

3. Minimization of Finite Automata

There are innumerable DFAs which have identical input-output responses (namely Hankel matrix) for a given DFA. It therefore becomes necessary to construct a minimal or reduced DFA which has the fewest states. Some algorithms of minimization are well known in the theory of FAs[2,4,5,7]. In this chapter, we propose an algorithm to minimize a given DFA M of n states using similarity relations.

3.1 Construction of the completely reachable deterministic finite automata

The completely reachable DFA can be constructed by elimination of unreachable states of M.

Let DFA M_R of n_R states be the completely reachable subsystem of M. The reachability matrix R_R of M_R is obtained by removing zero rows from the reachability matrix R of M.

Construct an $n_R\times n$ matrix T_R satisfying the following equation

$$T_RR=R_R. \tag{22}$$

Then T_R is a transformation matrix from M to M_R such that,

$$T_R=[t_{R_0},\ t_{R_1},\ \cdots,\ t_{R_{n-1}}] \tag{23}$$

$$t_{R_i}=\begin{cases} e_j & \text{if the state } q_i \text{ of } M \text{ is reachable} \\ & \text{and transformed to the state } q_j \text{ of } M_R. \\ 0 & \text{otherwise} \end{cases}$$

Since distinct states of M except unreachable states are transformed to different states of M_R, there is no identical column in T_R except the zero vector.

The observability matrix O_R of M_R is obtained by means of T_R as follows:

$$O_R=OT_R^t. \tag{24}$$

Thus the columns of O corresponding to unreachable states of M are removed to produce O_R.

The reachability matrix R_R and the observability matrix O_R of M_R were obtained in the way described above.

3.2 Construction of the minimal deterministic finite automata

A minimal DFA $\overline{M}$ of $\bar{n}$ states can be constructed by eliminating unobservable states and merging indistinguishable states of M_R. States corresponding to the zero columns of O_R are unobservable, and states which have an identical vector in the corresponding column of O_R are indistinguishable, so that the observability matrix $\overline{O}$ of $\overline{M}$ is obtained by removing zero columns from O_R and merging identical columns of O_R.

Construct an $\overline{n}\times n_R$ matrix T_O satisfying the following equation

$$\overline{O}T_O=O_R. \tag{25}$$

Then T_O is a transformation matrix from M_R to $\overline{M}$ such that,

$$T_O=[\boldsymbol{t}_{O_0},\ \boldsymbol{t}_{O_1},\ \cdots,\ \boldsymbol{t}_{O_{n_R-1}}] \tag{26}$$

$$\boldsymbol{t}_{O_i}=\begin{cases}\boldsymbol{e}_j & \text{if the state } q_i \text{ of } M_R \text{ is observable} \\ & \text{and transformed to the state } q_j \text{ of } \overline{M}. \\ \boldsymbol{0} & \text{otherwise}\end{cases}$$

Since indistinguishable states are transformed to a same state of $\overline{M}$, there are some identical columns in T_O. That is, if the states q_i and q_j of M_R are indistinguishable, then, $\boldsymbol{t}_{O_i}$ and $\boldsymbol{t}_{O_j}$ are identical.

The reachability matrix $\overline{R}$ of $\overline{M}$ is obtained as follows:

$$\overline{R}=T_OR_R. \tag{27}$$

Let $T=T_OT_R$, then an $\overline{n}\times n$ matrix T satisfies next equations.

$$\overline{R}=TR \tag{28}$$

$$\overline{O}=OT^t \tag{29}$$

T is a transformation matrix from M to $\overline{M}$ such that,

$$T=[\boldsymbol{t}_0,\ \boldsymbol{t}_1,\ \cdots,\ \boldsymbol{t}_{n-1}]$$

where $\boldsymbol{t}_i$ is a state vector of $\overline{M}$ corresponding to the state q_i of M. If the state q_i is unreachable or unobservable, $\boldsymbol{t}_i$ is zero vector and if the states q_i and q_j of M are indistinguishable, then $\boldsymbol{t}_i=\boldsymbol{t}_j$.

System parameters $(\{\overline{A_k}\},\ \overline{\boldsymbol{c}},\ \overline{\boldsymbol{x}_0})$ of $\overline{M}$ is obtained from $(\{A_k\},\ \boldsymbol{c},\ \boldsymbol{x}_0)$ of M using similarity relations (14)~ (16) for T of (30), in which M_a and M_b are replaced by M and $\overline{M}$ respectively.

To prove that the above algorithm provides an equivalent DFA for a given DFA, we show in Appendix that the Hankel matrix is invariant under minimization procedure.

Example 2 :

We perform the minimization of FA M_1 shown in Fig. 1.

To simplify the computation of matrices,we use R' (O') which is constructed by taking out different columns (rows) from $R(O)$.

Then we have R',O' as follows:

$$R'=\begin{pmatrix}1&0&0&0\\0&1&0&0\\0&0&1&0\\0&0&0&0\\0&0&0&1\end{pmatrix} \tag{31}$$

$$O'=\begin{pmatrix}0&0&1&0&0\\0&0&0&1&0\\1&1&1&0&0\\0&0&0&0&0\\1&0&1&1&0\\0&0&1&1&0\\1&1&1&1&0\end{pmatrix} \tag{32}$$

By removing zero rows from R', R'_R is obtained as follows:

$$R'_R=\begin{pmatrix}1&0&0&0\\0&1&0&0\\0&0&1&0\\0&0&0&1\end{pmatrix}. \tag{33}$$

Then T_R is solved by use of Eqs. (22), (31) and (33) as

$$T_R=\begin{pmatrix}1&0&0&0&0\\0&1&0&0&0\\0&0&1&0&0\\0&0&0&0&1\end{pmatrix}. \tag{33}$$

Next O'_R is obtained on referring to Eq. (24).

$$O'_R=\begin{pmatrix}0&0&1&0\\0&0&0&0\\1&1&1&0\\0&0&0&0\\1&0&1&0\\0&0&1&0\\1&1&1&0\end{pmatrix} \tag{35}$$

We have $\overline{O}'$ from O'_R.

$$\overline{O}'=\begin{pmatrix}0&0&1\\0&0&0\\1&1&1\\0&0&0\\1&0&1\\0&0&1\\1&1&1\end{pmatrix} \tag{36}$$

T_O from Eq. (25) and $T=T_OT_R$ are obtained as follows:

$$T_O=\begin{pmatrix}1&0&0&0\\0&1&0&0\\0&0&1&0\end{pmatrix} \tag{37}$$

$$T=\begin{pmatrix}1&0&0&0&0\\0&1&0&0&0\\0&0&1&0&0\end{pmatrix}. \tag{38}$$

System parameters $(\{\overline{A_k}\}, \overline{c}, \overline{x_0})$ of $\overline{M_1}$ is then obtained by similarity relations for T as

$$\overline{A_0}=\begin{pmatrix}0&0&1\\1&0&0\\0&0&0\end{pmatrix}, \overline{A_1}=\begin{pmatrix}0&0&0\\0&0&0\\1&1&1\end{pmatrix}. \tag{38}$$

$$\overline{c}^t=\begin{pmatrix}0&0&1\end{pmatrix}$$

$$\overline{x_0}=\begin{pmatrix}1\\0\\0\end{pmatrix}.$$

Figure 2 is the transition diagram of $\overline{M_1}$.

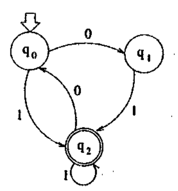

Fig. 2 Minimal DFA $\overline{M_1}$

4. Conclusion

By introducing state space models for FAs regarding as bilinear systems,FAs can be treated algebraically based on linear algebra over the boolean semiring.

As a result,the similarity relations can be defined to FAs. As its applications, we showed the methods of subset construction and minimization of FAs on state space models.

Appendix

We prove that Hankel matrices of M and M_R correspond to each other.

Let H_R be the Hankel matrix of M_R. Then the following equation is obtained from Eqs(22) and (24).

$$H_R=O_RR_R=OT_R^tT_RR \tag{42}$$

$T_R^tT_R$ is a square matrix of order n and the (i, j) element of $T_R^tT_R$ is $t_{Ri}^tt_{Rj}$ such that

$$t_{Ri}^tt_{Rj}=\begin{cases}1 & i=j \text{ and } t_{Ri}\neq 0\\ 0 & \text{otherwise}\end{cases}.$$

Let $R=[r_0, r_1, \ldots, r_{n-1}]^t$, then

$$T_R^tT_RR=\begin{pmatrix}t_{R0}^tt_{R0}r_0^t\\ t_{R1}^tt_{R1}r_1^t\\ \vdots\\ t_{R_{n-1}}^tt_{R_{n-1}}r_{n-1}^t\end{pmatrix} \tag{43}$$

where $t_{Ri}^tt_{Ri}$ is 0 iff the state q_i of M is unreachable. Then, clearly $r_i=0$, so the next equation is obtained.

$$T_R^tT_RR=R \tag{44}$$

Thus, we obtain the next equations,

$$H_R=O_RR_R \tag{45}$$

$$=OT_R^tT_RR \tag{46}$$

$$=OR \tag{47}$$

$$=H, \tag{48}$$

and H_R consists with H.

Though we should prove that Hankel matrix of $\overline{M}$ and H are consistent, it is omitted since the proof is almost the same as the above-mentioned.

References

1) C. Chi-Tsong, "Linear System Theory and Design", CBS College Publishing, New York, USA(1984).
2) T. Ikai, M. Masumoto and K. Fukunaga, Technical Report of IEICE, COMP95-62(1995-11).
3) T. Ikai, T. Shintani and K. Fukunaga, Technical Report of IEICE, COMP95-63(1995-11).
4) K. Watanabe, T. Ikai and K. Fukunaga, Technical Report of IEICE, CST96-32(1997-01).
5) J. E. Hopoft, J. D. Ullman, "Introduction to Automata Theory, Languages and Computation", Addison-Wesley Publishing Co. Inc. Reading, Massachusetts, USA(1979).
6) J.E.Hopcroft, "An *n log n* algorithm for minimizing the states in a finite automaton", in *The Theory of Machines and Computations*(Kahavi, ed.), pp.189-196, Academic Press, New York, USA(1971).
7) T. Ikai, K. Furusawa and K. Fukunaga, Technical Report of IEICE, CST96-31(1997-01).