State estimation algorithms for non-linear stochastic sequential machines

S. G. Tzafestas

Computer Control Section, Computer Centre, Nuclear Research Centre, 'Democritus', Athens, Greece

The problem of estimating the state sequence for a number of stochastic sequential machine models is considered. Basically, all models are assumed to have the structure commonly used in the stochastic control field, i.e. that of a deterministic system corrupted by random disturbances. A Bayesian sequential information processing approach is followed which is most convenient for dealing with finite-state systems. The resulting estimators are given in the form of sequential algorithms which are suitable for a digital computer. However, these estimators can be reduced in the form of finite machines with the aid of well-known state equivalence/reduction techniques. An example is presented which illustrates the effectiveness and usefulness of the theory. The results of the paper are applicable to a large variety of digital communication systems involving noisy channels and also to quantised stochastic control processes.

(Received July 1971)

1. Introduction

Much research has been devoted in recent years to the problems of designing optimal filters, predictors and smoothing filters for conventional dynamic systems, lumped or distributed, corrupted by random disturbances and noises (Bucy, 1968; Cox, 1964; Handschin, 1970; Kailath, 1968; Kalman, 1963; Meditch, 1967 and 1970; Raugh, 1963; Tzafestas and Nightingale, 1969). However, not much analogous theory exists for stochastic finite automata and sequential machines (Booth, 1970). The research in the automata field was mainly devoted to the design of realisable models (Gelenbe, 1971), state assignment and decomposition (Bacon, 1964; Stearns and Hartmanis, 1961), and state reduction (Even, 1965; Carlyle, 1963; Nieh, 1970).

The present paper is the first part of a work dealing with the application of modern stochastic estimation and control techniques to finite-state systems such as automata and sequential machines. The results will find application to the optimal design of time-sharing computing systems, of digital data processors, of digital noisy communication systems, of sampled data processes with quantised states etc. This paper treats the filtering, prediction and smoothing problem of several, hardware-realisable, stochastic sequential machine models.* A Bayesian approach is adopted which is most convenient for treating finite-state systems.

The results are given in the form of sequential computational algorithms, i.e. in the form of growing-state machines. However, using well-known state-reduction techniques (Booth, 1967, 1970) these infinite-state machines can be reduced to equivalent finite-state ones which then can be realised with conventional digital hardware equipment by using the results of Gelenbe (1971).

An example is studied which illustrates all aspects of the present theory namely the realisability of the stochastic sequential machines considered, the usefulness of the estimation results and finally the possibility of reducing the estimators into realisable finite-state machines.

2. Problem formulation

The basic model to be first treated in the present paper is a stochastic sequential machine (SSM) M_1 of the type:

$$M_1 : \left. \begin{aligned} x(k+1) &= f_1(x(k), w(k), k) \\ y(k) &= g_1(x(k), v(k), k) \end{aligned} \right\} k = 0, 1, 2, \ldots \qquad (1)$$

where k denotes discrete time, $x(k)$ is the state vector of the machine at time k taking values from a finite vector set $\mathbf{X} = \{x_1, x_2, \ldots, x_n\}$, called the state set; $y(k)$ is the output vector of the machine at time k taking values from a finite vector set $\mathbf{Y} = \{y_1, y_2, \ldots y_m\}$, called the output set; $w(k)$ is the input random vector disturbance at time k taking values from a finite set $\mathbf{W} = \{w_1, w_2, \ldots w_\alpha\}$, called the input disturbance set; $v(k)$ is the output random disturbance at time k taking values from a finite set $\mathbf{V} = \{v_1, v_2, \ldots v_\beta\}$, called the output disturbance set; and f_1, g_1 are known vector-valued functions, usually described by transition tables or state diagrams, which define the behaviour of the machine. Any known input functions, such as test sequence signals or control inputs are accounted for by the explicit dependence of f_1 and g_1 on the time argument k.

It is assumed that the random disturbances $w(k)$ and $v(k)$ have known discrete probability distributions

$$\left. \begin{aligned} \{p(w)\} &= \{p(w_i) : w_i \in \mathbf{W}\} \text{ with } \sum_{w_i \in \mathbf{W}} p(w_i) = 1 \\ \{p(v)\} &= \{p(v_i) : v_i \in \mathbf{V}\} \text{ with } \sum_{v_i \in \mathbf{V}} p(v_i) = 1 \end{aligned} \right\} \qquad (2)$$

and possess the following properties:

(i) $w(k)$ and $v(k)$ are stationary
(ii) $w(k)$ and $v(k)$ are white
(iii) $w(k)$ and $v(\sigma)$ are independent for all k, $\sigma = 0, 1, 2, \ldots$

This is actually a Mealy type machine with random inputs and is denoted here as;

$$M_1 = \langle \mathbf{W}, \mathbf{V}, \mathbf{X}, \mathbf{Y}, \{p(w)\}, \{p(v)\}, f_1, g_1 \rangle \qquad (3)$$

where

$$f_1 : \mathbf{X} \times \mathbf{W} \to \mathbf{X} \quad \text{(next-state mapping)}$$
$$g_1 : \mathbf{X} \times \mathbf{V} \to \mathbf{Y} \quad \text{(output mapping)}$$

A block diagram of this machine is shown in **Fig. 1.** Since the disturbances $w(k)$ and $v(k)$ are white, the machine M_1 actually represents a finite-state Markovian process. The problem to be studied here is:

Given a sequence of observed outputs

$$Y_k = \{y(0), y(1), \ldots, y(k)\} \qquad (4)$$

of the SSM M_1, and an *a priori* probability distribution

$$\{p(x(0))\} = \{p_i(0) = \text{prob}\,[x(0) = x_i] : x_i \in \mathbf{X}\}$$
$$\sum_{i=1}^{n} p_i(0) = 1 \qquad (5)$$

Find an optimal estimate $\hat{x}(k+\theta|k)$ of $x(k+\theta)$ for $\theta = 0$, $\theta > 0$ and $\theta < 0$.

Note that when $\theta = 0$ we have the 'filtering' problem in which an optimal estimate of the current state is sought, when $\theta > 0$ we have the 'prediction' problem in which an optimal estimate of a future state is sought, and finally when $\theta = -\sigma < 0$

*These models cover most of the situations encountered in practice.

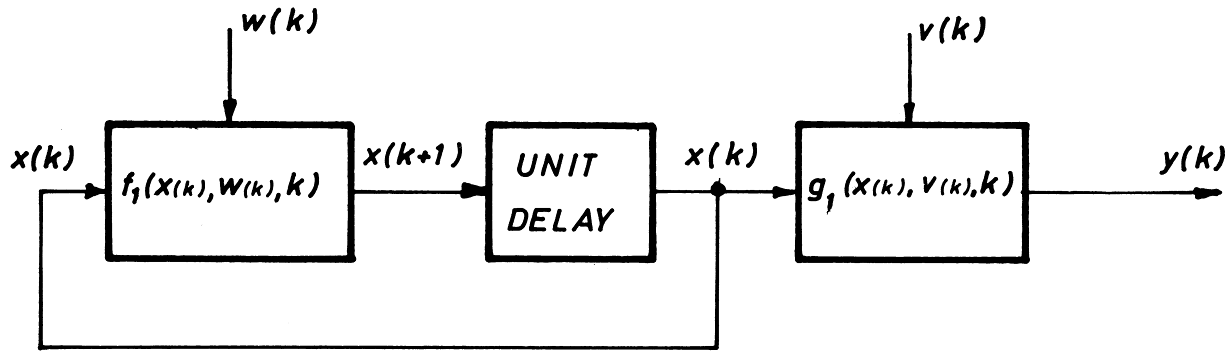

Fig. 1 Block diagram of SSM M_1

we have the 'smoothing' problem in which an estimate of a past state is sought.

In the above formulation one can include in the state vector $x(k)$ unknown parameters either constant or random (machine identification problem), and also unknown input signals (messages) generated by noisy sources (digital communication problem). For completeness, it is briefly illustrated here how the SSM M_1 may represent a digital communication system composed by a noisy digital message source followed by a digital message processor (coder, modulator, etc.) and by a noisy channel with memory.

It is assumed that the vector-valued message $x_m(k)$ generated by the source, is described by the stochastic sequential machine equation

$$x_m(k + 1) = f_m(x_m(k), w_m(k), k)$$

where the function f_m is known and the random disturbance $w_m(k)$ is white. The message processor unit receives $x_m(k)$ and gives the signal $y_p(k)$. Its equations are

$$x_p(k + 1) = f_p(x_p(k), x_m(k), k)$$
$$y_p(k) = g_p(x_p(k), x_m(k), k)$$

where $x_p(k)$ is the state vector of the unit. Finally the channel receives $y_p(k)$ and transmits the observed signal $y(k)$. The channel equations are

$$x_c(k + 1) = f_c(x_c(k), y_p(k), w_c(k), k)$$
$$y(k) = g_c(x_c(k), y_p(k), v(k), k)$$

Defining the overall state vector $x(k)$ and the next state function f_1 as:

$$x(k) = \begin{bmatrix} x_m(k) \\ \hline x_p(k) \\ \hline x_c(k) \end{bmatrix},$$

$$f_1(x(k), w(k), k) = \begin{bmatrix} f_m(x_m(k), w_m(k), k) \\ \hline f_p(x_p(k), x_m(k), k) \\ \hline f_c\{x_c(k), g_p(x_p(k), x_m(k), k), w_c(k), k\} \end{bmatrix}$$

the overall communication system equations take the form (1) with

$$g_1(x(k), v(k), k) = g_c\{x_c(k), g_p(x_p(k), x_m(k), k), v(k), k\}$$

and

$$w(k) = \begin{bmatrix} w_m(k) \\ \hline w_c(k) \end{bmatrix}$$

3. Filtering problem

In this case we seek the estimate $\hat{x}(k|k)$. Let us define the conditional probabilities $S_i(k|k)$ and $S_i(k + 1|k)$ as:

$$S_i(k|k) = \text{prob}\,\{x(k) = x_i | Y_k\} \qquad i = 1, 2, \ldots n \tag{6}$$
$$S_i(k + 1|k) = \text{prob}\,\{x(k + 1) = x_i | Y_k\} \quad k = 0, 1, 2, \ldots$$

It is well known that $S_i(k|k) : i = 1, 2, \ldots, n$ contain all the necessary information for estimating $x(k)$, i.e. for computing $\hat{x}(k|k)$.

A recursive equation for computing $\{S_i(k|k) : k = 1, 2, \ldots\}$ can be found by making use of Baye's rule as follows:

$$S_i(k + 1|k + 1) = \text{prob}\,\{x(k + 1) = x_i | Y_{k+1}\}$$
$$= \frac{\text{prob}\,\{x(k + 1) = x_i,\ Y_k, y(k + 1)\}}{\text{prob}\,\{Y_k, y(k + 1)\}} \tag{7}$$
$$= \frac{\text{prob}\,\{y(k + 1)|x(k + 1) = x_i\}\ \text{prob}\,\{x(k + 1) = x_i | Y_k\}}{\sum_{i=1}^{n} \text{prob}\,\{y(k + 1)|x(k + 1) = x_i\}\ \text{prob}\,\{x(k + 1) = x_i | Y_k\}}$$

Now using the output equation $y(k) = g_1(x(k), v(k), k)$ we see that given the observation $y(k)$ at time k, the probabilities

$$Z_i(k) = \text{prob}\,\{y(k)|x(k) = x_i\} = \sum_{v \in \mathbf{V}_k} p(v) \tag{8}$$
$$k = 0, 1, 2, \ldots; i = 1, 2, \ldots, n$$

where

$$\mathbf{V}_k = \{v \in \mathbf{V} : y(k) = g_1(x_i, v, k\} \tag{9}$$

can be easily computed. Hence $Z_i(k) : i = 1, 2, \ldots, n$; $k = 0, 1, 2, \ldots$ are assumed to be known.

Introducing the following vector notation:

$$\mathbf{S}(k|k) = \begin{bmatrix} S_1(k|k) \\ S_2(k|k) \\ \vdots \\ S_n(k|k) \end{bmatrix}, \ \mathbf{S}(k + 1|k) = \begin{bmatrix} S_1(k + 1|k) \\ S_2(k + 1|k) \\ \vdots \\ S_n(k + 1|k) \end{bmatrix},$$

$$\mathbf{Z}(k) = \begin{bmatrix} Z_1(k) \\ Z_2(k) \\ \vdots \\ Z_n(k) \end{bmatrix} \tag{10}$$

$$\mathbf{Z}(k + 1)^*\mathbf{S}(k + 1|k) = \begin{bmatrix} Z_1(k + 1)S_1(k + 1|k) \\ Z_2(k + 1)S_2(k + 1|k) \\ \vdots \\ Z_n(k + 1)S_n(k + 1|k) \end{bmatrix},$$

$$\langle \mathbf{Z}(k + 1), \mathbf{S}(k + 1|k) \rangle$$
$$= \{Z_1(k + 1), \ldots, Z_n(k + 1)\} \begin{bmatrix} S_1(k + 1|k) \\ \vdots \\ S_n(k + 1|k) \end{bmatrix} \tag{11}$$

equation (7) can be written in vectorial form as:

$$\mathbf{S}(k + 1|k + 1) = \frac{\mathbf{Z}(k + 1)^*\mathbf{S}(k + 1|k)}{\langle \mathbf{Z}(k + 1), \mathbf{S}(k + 1|k) \rangle}\ ; k = 0, 1, 2 \ldots \tag{12}$$

The probability $S_i(k + 1|k)$ can be computed as follows:

$$S_i(k + 1|k) = \text{prob}\,\{x(k + 1) = x_i | Y_k\}$$

$$= \sum_{j=1}^{n} \text{prob}\,\{x(k+1) = x_i, x(k) = x_j | Y_k\}$$

$$= \sum_{j=1}^{n} \text{prob}\,\{x(k+1) = x_i | x(k) = x_j\} \times \text{prob}\,\{x(k) = x_j | Y_k\}$$

$$= \sum_{j=1}^{n} M_{ji}(k)\, S_j(k|k) = \mathbf{M}^T(k)\, \mathbf{S}(k|k) \qquad (13)$$

where $M_{ij}(k)$ is defined as:

$$\left.\begin{aligned} M_{ji}(k) &= \text{prob}\,\{x(k+1) = x_i | x(k) = x_j\} = \sum_{w \in \mathbf{W}_k} p(w) \\ \mathbf{W}_k &= \{x \in \mathbf{W} : x_i = f_1(x_j, w, k)\} \end{aligned}\right\} \qquad (14)$$

and is the (j, i)th submatrix of the transition probability matrix

$$\mathbf{M}(k) = [M_{ij}(k)] = \begin{bmatrix} M_{11}(k) & \cdots\cdots & M_{1n}(k) \\ M_{21}(k) & \cdots\cdots & M_{2n}(k) \\ \cdots & \cdots & \cdots \\ \cdots & \cdots & \cdots \\ M_{n1}(k) & \cdots\cdots & M_{nn}(k) \end{bmatrix} \qquad (15)$$

Here $\mathbf{M}^T(k)$ is the transpose of $\mathbf{M}(k)$.

Combining equations (12) and (13) yields:

$$\mathbf{S}(k+1|k+1) = \frac{\mathbf{Z}(k+1)^*\{\mathbf{M}^T(k)\,\mathbf{S}(k|k)\}}{\langle \mathbf{Z}(k+1), \mathbf{M}^T(k)\,\mathbf{S}(k|k)\rangle} \triangleq \mathbf{F}_f(y(k+1), \mathbf{S}(k|k), k) \qquad (16)$$

which is the desired recursive scheme for computing the probabilities $\{S(k|k) : k = 0, 1, 2, \ldots\}$. The function $\mathbf{F}_f$ is a mapping of the set $\mathbf{Y} \times \mathbf{S}$ onto the set $\mathbf{S}$ where

$$\mathbf{S} = \{S(k|k) : k = 0, 1, 2, \ldots\}$$

which obviously is a growing set with a countable number of elements as k increases. That is:

$$\mathbf{F}_f : \mathbf{Y} \times \mathbf{S} \rightarrow \mathbf{S} \qquad (17)$$

The initial condition $\mathbf{S}(0|0)$ required for initiating the scheme (17) can be found in a similar way and is:

$$\mathbf{S}(0|0) = \frac{\mathbf{Z}(0)^*\mathbf{p}(0)}{\langle \mathbf{Z}(0), \mathbf{p}(0)\rangle},\ \mathbf{p}(0) = \begin{bmatrix} p_1(0) \\ \vdots \\ p_n(0) \end{bmatrix} \qquad (18)$$

The optimal estimate $\hat{x}(x|k)$ of $x(k)$ can be defined as

$$\hat{x}(k|k) = \{x_i \in \mathbf{X} : S_i(k|k) = \max\} \triangleq \mathbf{G}_f\{\mathbf{S}(k|k)\} \qquad (19)$$

i.e. as the *mode* of $\mathbf{S}(k|k)$. The function G_f maps the set $\mathbf{S}$ into the set $\hat{\mathbf{X}} \subseteq \mathbf{X}$, i.e.:

$$\mathbf{G}_f : \mathbf{S} \rightarrow \hat{\mathbf{X}}$$

Equations (17)-(19) define a deterministic sequential machine (filter) M_{f_1} with input set $\mathbf{Y}$, growing countable state set $\mathbf{S}$, and output set $\hat{\mathbf{X}}$, i.e.

$$M_{f_1} = \langle \mathbf{Y}, \mathbf{S}, \hat{\mathbf{X}}, \mathbf{F}_f, \mathbf{G}_f \rangle \qquad (20)$$

The block diagram of this machine is shown in **Fig. 2.**

The computations required by the machine M_{f_1} can easily be performed by a general purpose digital computer. In practice, however, it should be desirable to reduce M_{f_1} to a finite-state physically-realisable sequential machine $M_{f_1}^*$ which is equivalent to M_{f_1} in a certain predefined sense. A discussion of such a reduction was given by Booth (1970), and will not be repeated here.

4. Prediction problem

The optimal estimate $\hat{x}(k+\theta|k)$, $\theta > 0$ is defined as:

$$\hat{x}(k+\theta|k) = \{x_i \in \mathbf{X} : S_i(k+\theta|k) = \max\} \triangleq \mathbf{G}'_p\{\mathbf{S}(k+\theta|k) \qquad (21)$$

To determine the density $S_i(k+\theta|k)$ we use the relation

$$S_i(k+\theta|k) = \text{prob}\,\{x(k+\theta|k) = x_i | Y_k\}$$

$$= \sum_{l_0=1}^{n} \text{prob}\,\{x(k+\theta|k) = x_i | x(k) = x_{l_0}\}\ \text{prob}\,\{x(k) = x_{l_0} | Y_k\}$$

$$= \sum_{l_0=1}^{n} \sum_{l_1=1}^{n} \text{prob}\,\{x(k+\theta|k) = x_i | x(k+1) = x_{l_1}\} \times \text{prob}\,\{x(k+1) = x_{l_1} | x(k) = x_{l_0}\}\ S_{l_0}(k|k)$$

....

....

....

....

$$= \sum_{l_0=1}^{n} \sum_{l_1=1}^{n} \cdots \sum_{l_\theta=1}^{n} M_{(l_{\theta-1})i}\,(k+\theta-1) \cdots \times M_{l_1 l_2}\,(k+1) M_{l_0 l_1}(k) S_{l_0}(k|k) \qquad (23)$$

Equation (23) can be written in vectorial form as:

$$\mathbf{S}(k+\theta|k) = \mathbf{M}^T(k+\theta-1)\,\mathbf{M}^T(k+\theta-2)\ldots\mathbf{M}^T(k+1) \times \mathbf{M}^T(k)\,\mathbf{S}(k|k) \qquad (24)$$

Equations (21), (24) and (16) define a sequential predictor machine M_{p_1} with input set $\mathbf{Y}$, growing state set

$$\mathbf{S} = \{S(k|k) : k = 0, 1, 2, \ldots\}$$

and output set $\hat{\mathbf{x}}_\theta \subseteq \mathbf{X}$, i.e.:

$$M_{p_1} = \langle \mathbf{Y}, \mathbf{S}, \hat{\mathbf{X}}_\theta, \mathbf{F}_f, \mathbf{G}_p \rangle \qquad (25)$$

where the output mapping $\mathbf{G}_p$ is defined by equation (21) as:

$$\begin{aligned} \hat{x}(k+\theta|k) &= \mathbf{G}'_p\{\mathbf{S}(k+\theta|k) \\ &= \mathbf{G}'_p\{\mathbf{M}^T(k+\theta-1)\ldots\mathbf{M}^T(k+1)\,\mathbf{M}^T(k)\,\mathbf{S}(k|k)\} \\ &= \mathbf{G}_p\{\mathbf{S}(k|k)\} \end{aligned} \qquad (26)$$

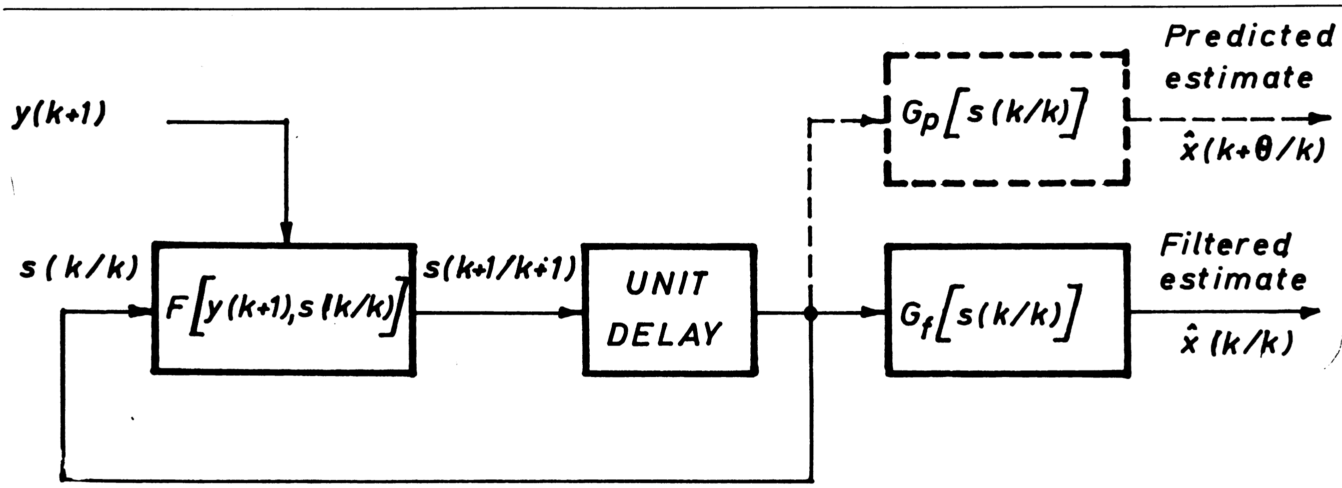

Fig. 2 **Block diagram of optimal filter M_{f_1} and smoothing filter M_{p_1} given by equations (20) and (25)**

We observe that the machines M_{f_1} and M_{p_1} have the same input set, the same state set, the same next state function $\mathbf{F}_f$, and differ only in the output mapping. This is illustrated by the dotted part of Fig. 2.

5. Smoothing problem

The problem is to determine an estimate $\hat{x}(k-\sigma|k)$ of $x(k-\sigma)$ on the basis of the measured set Y_k.

For simplicity let us find first an estimate $\hat{x}(k|L)$ of $x(k)$ on the basis of measured data Y_L.

Define the smoothed probability density.

$$S_i(k|L) = \text{prob}\,\{x(k) = x_i|\,Y_L\} \tag{27}$$

Then,

$$\begin{aligned} S_i(k|L) &= \sum_{j=1}^{n} \text{prob}\,\{x(k) = x_i|x(k+1) = x_j,\, Y_L\} \\ &\qquad \times \text{prob}\,\{x(k+1) = x_j|\,Y_L\} \\ &= \sum_{j=1}^{n} \text{prob}\,\{x(k) = x_i|x(k+1) = x_j,\, Y_k\}\, S_j(k+1|L) \end{aligned} \tag{28}$$

Now,

$$\begin{aligned} &\text{prob}\,\{x(k) = x_i|x(k+1) = x_j,\, Y_k\} \\ &= \frac{\text{prob}\,\{x(k+1) = x_j|x(k) = x_i,\, Y_k\}\ \text{prob}\,\{x(k) = x_i|\,Y_k\}}{\text{prob}\,\{x(k+1) = x_j|\,Y_k\}} \\ &= \frac{\text{prob}\,\{x(k+1) = x_j|x(k) = x_i\}\ \text{prob}\,\{x(k) = x_i|\,Y_k\}}{\sum_{i=1}^{n} \text{prob}\,\{x(k+1) = x_j|x(k) = x_i\}\ \text{prob}\,\{x(k) = x_i|\,Y_k\}} \\ &= \frac{M_{ij}(k)\, S_i(k|k)}{\sum_{i=1}^{n} M_{ij}(k)\, S_i(k|k)} \end{aligned} \tag{29}$$

Thus finally,

$$S_i(k|L) = \sum_{j=1}^{n} \left[\frac{S_i(k|k) M_{ij}(k) S_j(k+1|L)}{\sum_{i=1}^{n} M_{ij}(k) S_i(k|k)} \right] \tag{30}$$

or in vectorial form

$$\mathbf{S}(k|k) = \sum_{j=1}^{n} \frac{\{\mathbf{S}(k|k)^*\mathbf{M}_j(k)\} S_j(k+1|L)}{\langle \mathbf{M}_j^T(k), \mathbf{S}(k|k)\rangle} \tag{31}$$

where:

$$\mathbf{S}(k|L) = \begin{bmatrix} S_1(k|L) \\ S_2(k|L) \\ \vdots \\ S_n(k|L) \end{bmatrix},\ \mathbf{M}_j(k) = \begin{bmatrix} M_{1j}(k) \\ M_{2j}(k) \\ \vdots \\ M_{nj}(k) \end{bmatrix} \tag{32}$$

Using equation (31) for computing $\mathbf{S}(k-r|k) : r = 1, 2, \ldots, \sigma$ yields

$$\mathbf{S}(k-1|k) = \sum_{j=1}^{n} \frac{\{\mathbf{S}(k-1|k-1)^*\mathbf{M}_j(k)\}\ S_j(k|k)}{\langle \mathbf{M}_j^T(k), \mathbf{S}(k-1|k-1)\rangle}$$

$$\mathbf{S}(k-2|k) = \sum_{j=1}^{n} \frac{\{\mathbf{S}(k-2|k-2)^*\mathbf{M}_j(k)\}\ S_j(k-1|k)}{\langle M_j^T(k), \mathbf{S}(k-2|k-2)\rangle}$$

....

....

....

....

$$\mathbf{S}(k-\sigma|k) = \sum_{j=1}^{n} \frac{\{\mathbf{S}(k-\sigma|k-\sigma)^*\mathbf{M}_j(k)\}\ S_j(k-\sigma+1|k)}{\langle \mathbf{M}_j^T(k), \mathbf{S}(k-\sigma|k-\sigma)\rangle} \tag{33}$$

By consecutive substitutions it is easily seen that $\mathbf{S}(k-\sigma|k)$ is a function of $S(k|k)$, $S(k-1|k-1)$, $S(k-2|k-2), \ldots$ $S(k-\sigma|k-\sigma)$, i.e. the smoothed probability density $\mathbf{S}(k-\sigma|k)$ is a function of the σ previous filtered densities:

$$\mathbf{S}(k-\sigma|k) = \mathbf{F}_S\{\mathbf{S}(k|k), \mathbf{S}(k-1|k-1), \ldots, \mathbf{S}(k-\sigma|k-\sigma)\} \tag{34}$$

where by equation (16):

$$\left.\begin{aligned} \mathbf{S}(k|k) &= \mathbf{F}_f\{y(k), \mathbf{S}(k-1|\,k-1), k\} \\ \mathbf{S}(k-1|k-1) &= \mathbf{F}_f\{y(k-1), \mathbf{S}(k-2|k-2, k\} \\ &\ldots\ldots\ldots\ldots\ldots \\ &\ldots\ldots\ldots\ldots\ldots \\ &\ldots\ldots\ldots\ldots\ldots \\ \mathbf{S}(k-\sigma|k-\sigma) &= \mathbf{F}_f\{y(k-\sigma), \\ &\qquad \mathbf{S}(k-\sigma-1|k-\sigma-1), k\} \end{aligned}\right\} \tag{35}$$

Defining the augmented state vectors $\mathbf{S}_\sigma(k|k)$ and $\mathbf{F}_{f\sigma}$ as:

$$\mathbf{S}_\sigma(k|k) = \begin{bmatrix} \mathbf{S}(k|k) \\ \mathbf{S}(k-1|k-1) \\ \vdots \\ \mathbf{S}(k-\sigma|k-\sigma) \end{bmatrix},\ \mathbf{F}_{f\sigma} = \begin{bmatrix} \mathbf{F}_f \\ \mathbf{F}_f \\ \vdots \\ \mathbf{F}_f \end{bmatrix} \tag{36}$$

equations (34) and (35) can be written in vectorial form as:

$$\mathbf{S}(k-\sigma|k) = \mathbf{F}_S\{\mathbf{S}_\sigma(k|k)\} \tag{37}$$

$$\mathbf{S}_\sigma(k|k) = \mathbf{F}_{f\sigma}\{\mathbf{S}_\sigma(k-1|k-1, Y_{k,\sigma}\} \tag{38}$$

where

$$Y_{k,\sigma} = \{y(k-\sigma), y(k-\sigma+1), \ldots, y(k-1), y(k)\} \tag{39}$$

The optimal estimate $\hat{x}(k-\sigma|k)$ is given by:

$$\begin{aligned} \hat{x}(k-\sigma|k) &= \{x_i \in \mathbf{X} : S_i(k-\sigma|k) = \max\} \\ &\triangleq \mathbf{G}'_{sm}\{\mathbf{S}(k-\sigma|k)\} \\ &= \mathbf{G}'_{sm}\{F_S(\mathbf{S}_\sigma(k|k))\} \\ &\triangleq \mathbf{G}_{sm}\{\mathbf{S}_\sigma(k|k)\} \end{aligned} \tag{40}$$

Equations (37)-(40) define a sequential smoother machine M_{sm} with state

$$\mathbf{S}_\sigma(k|k) \in \mathbf{S}_\sigma = \overbrace{\mathbf{S} \times \mathbf{S} \times \ldots \times \mathbf{S}}^{\sigma}$$

input

$$Y_{k,\sigma} \in \mathbf{Y}_\sigma = \overbrace{\mathbf{Y} \times \mathbf{Y} \times \ldots \times \mathbf{Y}}^{\sigma}$$

and output $\hat{x}(k-\sigma|k) \in \hat{\mathbf{X}}_\sigma \subseteq \mathbf{X}$. That is

$$M_{sm} = \langle \mathbf{Y}_\sigma, \mathbf{S}_\sigma, \hat{\mathbf{X}}_\sigma, \mathbf{F}_{f\sigma}, \mathbf{G}_{sm}\rangle \tag{41}$$

It is useful to note that the filtered densities $\mathbf{S}(k|k)$, $\mathbf{S}(k-1|k-1), \ldots, \mathbf{S}(k-\sigma|k-\sigma)$ involved in $\mathbf{S}(k|k)$, which is used in equation (40) for calculating $\hat{x}(k-\sigma|k)$, are computed in the process of determining $\hat{x}(k|k)$ as described in Section 3. From equation (34) it is clear that to initiate the smoothing process, i.e. determine the initial condition $\mathbf{S}(0|\sigma)$ the smoother machine M_{sm} must wait for a time of σ clock periods during which the filter machine M_f calculates $S(0|0)$, $S(1|1), \ldots, S(\sigma|\sigma)$.

6. Treatment of other sequential-machine models

The purpose of the present section is to show how the results of the previous sections can be applied for treating other sequential-machine models.

SSMM_2

This is a Moore SSM model of the type:

$$M_2 : \left.\begin{aligned} x(k+1) &= f_2(x(k), w(k), k) \\ y(k) &= g_2(x(k), k) \end{aligned}\right\} \tag{42}$$

where $x \in \mathbf{X}$, $w \in \mathbf{W}$, $y \in \mathbf{Y}$, The stochastic disturbance $\{w(k)\}$ is assumed to be white and has a known probability distribution $\{p(w)\}$. Equivalently M_2 is denoted as:

$$\left.\begin{aligned} &M_2 = \langle \mathbf{W}, \mathbf{X}, \mathbf{Y}, \{p(w)\}, f_2, g_2\rangle \\ &f_2 : \mathbf{X} \times \mathbf{W} \to \mathbf{X} \\ &g_2 : \mathbf{X} \to \mathbf{Y} \end{aligned}\right\} \tag{43}$$

All the results derived for the machine M_1 are valid for M_2 as well, with the understanding that:

$$Z_i(k) = \text{prob}\,\{y(k)|x(k) = x_i\} = \begin{cases} 1 \text{ for } i = i^* \text{ such that } y(k) = g_2(x_i^*) \\ 0 \text{ for } i \neq i^* \end{cases} \quad (44)$$

The machine M_2 covers all cases where the disturbances appear only within the dynamics, i.e. only in the next state equation of the machine.

SSMM_3

This model has the form of SSMM_1, i.e.

$$M_3: \begin{rcases} x(k+1) = f_3(x(k), w(k), k), x \in \mathbf{X}, w \in \mathbf{W} \\ y(k) = g_3(x(k), v(k), k), \quad v \in \mathbf{V}, y \in \mathbf{Y} \end{rcases} \quad (45)$$

or

$$\begin{rcases} M_3 = \langle \mathbf{W}, \mathbf{V}, \mathbf{X}, \mathbf{Y}, \{p(w)\}, \{p(w|v)\}, f_3, g_3 \rangle \\ f_3 = \mathbf{X} \times \mathbf{W} \to \mathbf{X}, g_3 = \mathbf{X} \times \mathbf{V} \to \mathbf{Y} \end{rcases} \quad (46)$$

but the white disturbances $w(k)$ and $v(k)$ are correlated with known conditional probability distribution:

$$p(v(k) = v_i|w(k) = w_j), v_i \in \mathbf{V}, w_i \in \mathbf{W}$$

In this case the probabilities $p(v(k) = v_i)$ are given by

$$p(v(k) = v_i) = \sum_{w_j \in \mathbf{W}} p(v(k) = v_i, w(k) = w_j) \quad (47)$$

$$= \sum_{w_j \in \mathbf{W}} p(v(k) = v_i|w(k) = w_j)\, p(w(k) = w_j)$$

and

$$Z_i(k) = p(y(k)|x(k) = x_i) = \sum_{v \in \mathbf{V}_k} p(v) \quad (48)$$

$$\mathbf{V}_k = \{v \in \mathbf{V} : y(k) = g_3(x_i, v, k)\}$$

Using equations (47) and (48) all the results concerning M_1 are valid for M_3 as well.

SSMM_4

This model, in which the stochastic disturbances $w(k)$ and $v(k)$ are assumed to be Markovian and independent of each other, is described by the equations:

$$M_4: \begin{rcases} x(k+1) = f_4(x(k), w(k), k), x \in \mathbf{X}, w \in \mathbf{W} \\ y(k) = g_4(x(k), v(k), k), y \in \mathbf{Y}, v \in \mathbf{V} \end{rcases} \quad (49)$$

$$\begin{rcases} w(k+1) = h_w(w(k), m(k)), m \in \Sigma_m \\ v(k+1) = h_v(v(k), n(k)), \quad n \in \Sigma_n \end{rcases} \quad (50)$$

where the disturbances $m(k)$ and $n(k)$ are white, independent of each other, and assume values in the finite sets

$$\begin{rcases} \Sigma_m = \{m_1, m_2, \ldots, m_\gamma\} \\ \Sigma_n = \{n_1, n_2, \ldots, n_\delta\} \end{rcases} \quad (51)$$

The probability distributions $\{p(m)\}$ and $\{p(n)\}$ are assumed to be known.

This machine is denoted as:

$$M_4 = \langle \Sigma_m, \Sigma_n, \mathbf{W}, \mathbf{V}, \mathbf{X}, \mathbf{Y}, \{p(m)\}, \{p(n)\}, f_4, g_4, h_w, h_v \rangle \quad (52)$$

with

$$\begin{rcases} f_4 : \mathbf{W} \times \mathbf{V} \to \mathbf{X}, \; g_4 : \mathbf{V} \times \mathbf{X} \to \mathbf{Y} \\ h_w : \Sigma_m \times \mathbf{W} \to \mathbf{W}, h_v : \Sigma_n \times \mathbf{V} \to \mathbf{V} \end{rcases} \quad (53)$$

Using the probabilities $p(m)$ and $p(n)$ one can find the disturbance transition probabilities as follows:

$$\Phi_{ij} = \text{prob}\,\{w(k+1) = w_j|w(k) = w_i\}$$
$$= \sum_{m \in \Sigma_m^*} p(m), \Sigma_m^* = \{m \in \Sigma_m : w_j = h_w(w_i, m)\} \quad (54)$$

$$\Psi_{ij} = \text{prob}\,\{v(k+1) = v_j|v(k) = v_i\}$$
$$= \sum_{n \in \Sigma_n^*} p(n), \Sigma_n^* = \{n \in \Sigma_n : v_j = h_v(v_i, n)\} \quad (55)$$

The probabilities $\{p(w(k) = w_i)\}$ and $\{p(v(k) = v_i)\}$ are recursively computed by the equations:

$$\begin{rcases} p(w(k+1) = w_j) = \sum_{w_i \in \mathbf{W}} \text{prob}\,\{w(k+1) = w_j|w(k) = w_i\} \\ \times \text{prob}\,\{w(k) = w_i\} \\ = \sum_{w_i \in \mathbf{W}} \Phi_{ij} p(w(k) = w_i), p(w(0) = w_i) \text{ given for all } w_i \in \mathbf{W} \end{rcases} \quad (56)$$

$$\begin{rcases} p(v(k+1) = v_j) = \sum_{v_i \in \mathbf{V}} \text{prob}\,\{v(k+1) = v_j|v(k) = v_i\} \\ \times \text{prob}\,\{v(k) = v_i\} \\ = \sum_{v_i \in \mathbf{V}} \Psi_{ij} p(v(k) = v_i), p(v(0) = v_i \text{ given for all } v_i \in \mathbf{V} \end{rcases} \quad (57)$$

The transition probabilities $M_{ji}(k)$ are again given by

$$M_{ji}(k) = \sum_{w \in \mathbf{W}_k} p(w), \mathbf{W}_k = \{w \in \mathbf{W} : x_j = f_4(x_i, w, k)\} \quad (58)$$

where $\{p(w)\}$ is determined by equation (56), and the data probabilities $Z_i(k)$ are given by

$$Z_i(k) = \text{prob}\,\{y(k)|\, x(y) = x_i\}$$
$$= \sum_{v \in \mathbf{V}_k} p(v), \mathbf{V}_k = \{v \in \mathbf{V} : y(k) = g_4(x_i, v, k)\} \quad (59)$$

where $\{p(v)\}$ is determined by equation (57).

It is now clear that the results concerning the machine M_1 can equally well be applied to the machine M_4 with the understanding that in the later case $M_{ji}(k)$ and $Z_i(k)$ are determined using equations (58) and (59).

SSMM_5

This model has the equations

$$M_5: \begin{rcases} x(k+1) = f_5(x(k), u(k), w(k), k) \\ y(k) = g_5(x(k), x(k+1), u(k), v(k), k) \\ u(k) = h_5(x(k)) \end{rcases} \quad (60)$$

or equivalently

$$M_5 = \langle \mathbf{W}, \mathbf{V}, \mathbf{U}, \mathbf{X}, \mathbf{Y}, \{p(w)\}, \{p(v)\}, f_5, g_5, h_5 \rangle \quad (61)$$

with

$$f_5 : \mathbf{X} \times \mathbf{U} \times \mathbf{W} \to \mathbf{X}, g_5 : \mathbf{X} \times \mathbf{X} \times \mathbf{U} \times \mathbf{V} \to \mathbf{Y},$$
$$h_5 : \mathbf{X} \to \mathbf{U} \quad (62)$$

where $u(k)$, the input of the machine, assumes its values in the finite set

$$\mathbf{U} = \{u_1, u_2, \ldots, u_q\} \quad (63)$$

and h_5 is a mapping of $\mathbf{X}$ onto $\mathbf{U}$ that associates a unique element $u \in \mathbf{U}$ to each $x \in \mathbf{X}$. The properties of $\{w(k)\}$ and $\{v(k)\}$ are the same as in the SSMM_1. The main difference between M_5 and M_1 is that the output $y(k)$ of M_5 depends not only on the present state $x(k)$ but also on the next state $x(k+1)$. The machine M_5 is the realisable model derived by Gelenbe (1971) for a Mealy machine with random disturbances.

In the present case one knows the probabilities

$$M_{rij} = \text{prob}\,\{x(k+1) = x_j|x(k) = x_i, u(k) = u_r\}$$
$$= \sum_{w \in \mathbf{W}_k} p(w), \mathbf{W}_k = \{w \in \mathbf{W} : x_j = f_5(x_i, u_r, w)\} \quad (64)$$

$$Z_{rij}(k) = \text{prob}\,\{y(k)|x(k) = x_i, x(k+1) = x_j, u(k = u_r\}$$
$$= \sum_{v \in \mathbf{V}_k} p(v), \mathbf{V}_k = \{v \in \mathbf{V} : y(k) = g_5(x_i, x_j, u_r, v)\} \quad (65)$$

and

$$\pi_{ij}(k) = \text{prob}\,\{u(k) = u_j|x(k) = x_i\} = \begin{cases} 1 \text{ for } i = i^* \text{ and } u_j = h_5(x_i^*) \\ \text{for } i \neq i^* \end{cases} \quad (66)$$

Equation (13) which relates $S_i(k+1|k)$ and $S_i(k|k)$ takes the form:

$$S_i(k+1|k) = \text{prob}\,\{x(k+1) = x_i|\, Y_k\}$$
$$= \sum_{j=1}^{n} \sum_{r=1}^{q} \text{prob}\,\{x(k+1) = x_i, x(k) = x_j, u(k) = u_r|\, Y_k\}$$
$$= \sum_{j=1}^{n} \sum_{r=1}^{q} \text{prob}\,\{x(k+1) = x_i|x(k) = x_j, u(k) = u_r\}$$
$$\times \text{prob}\,\{u(k) = u_r|x(k) = x_j\}\, \text{prob}\,\{x(k) = x_j|\, Y_k\}$$
$$= \sum_{j=1}^{n} \sum_{r=1}^{q} M_{rji}\pi_{jr}(k)\, S_j(k|k) \quad (67)$$

or in vectorial form

$$\mathbf{S}(k+1|k) = \mathbf{M}_0^T(k)\, \mathbf{S}(k|k) \quad (68)$$

where the (j, i)th element of the matrix $\mathbf{M}_0(k)$ is defined as:

$$\mathbf{M}_{0ji}(k) = \sum_{r=1}^{q} M_{rji}\pi_{jr}(k) \quad (69)$$

Similarly equation (7) must be replaced by:

$$S_i(k+1|k+1) = \text{prob}\,\{x(k+1) = x_i | Y_{k+1}\}$$

$$= \sum_{j=1}^{n} \sum_{r=1}^{q} \text{prob}\,\{x(k+1) = x_i, x(k+2) = x_j, u(k+1) = u_r | Y_k, y(k+1)\}$$

$$= \sum_{j=1}^{n} \sum_{r=1}^{q} \frac{\text{prob}\,\{y(k+1)|x(k+1) = x_i, x(k+2) = x_j, u(k+1) = u_r\} \times \text{prob}\,\{x(k+1) = x_i, x(k+2) = x_j, u(k+1) = u_r | Y_k\}}{\text{prob}\,\{Y_k, y(k+1)\}}$$

$$= \sum_{j=1}^{n} \sum_{r=1}^{q} \frac{Z_{rji}(k+1)\,\text{prob}\,\{x(k+2) = x_j | x(k+1) = x_i, u(k+1) = u_r\}\,\text{prob}\,\{x(k+1) = x_i, u(k+1) = u_r | Y_k\}}{\text{prob}\,\{Y_k, y(k+1)\}}$$

$$= \sum_{j=1}^{n} \sum_{r=1}^{q} \left[\frac{Z_{rji}(k+1)\, M_{rij}\pi_{ir}(k+1)\, S_i(k+1|k)}{\sum_{i=1}^{n}\sum_{j=1}^{n}\sum_{r=1}^{q} Z_{rji}(k+1)\, M_{rij}\pi_{ir}(k+1)\, S_i(k+1|k)}\right] \quad (70)$$

Defining a vector $\boldsymbol{\Omega}(k+1)$ as:

$$\boldsymbol{\Omega}(k+1) = \begin{bmatrix} \Omega_1(k+1) \\ \Omega_2(k+1) \\ \vdots \\ \Omega_n(k+1) \end{bmatrix},$$

$$\Omega_i(k+1) = \sum_{j=1}^{n}\sum_{r=1}^{q} Z_{rji}(k+1)\, M_{rij}\pi_{ir}(k+1) \quad (71)$$

equation (70) can be written in vectorial form as:

$$\mathbf{S}(k+1|k+1) = \frac{\boldsymbol{\Omega}(k+1)^*\mathbf{S}(k+1|k)}{\langle \boldsymbol{\Omega}(k+1), \mathbf{S}(k+1|k)\rangle} = \frac{\boldsymbol{\Omega}(k+1)^*\{\mathbf{M}_0^T(k)\,\mathbf{S}(k|k)\}}{\langle \boldsymbol{\Omega}(k+1), \mathbf{M}_0^T(k)\,\mathbf{S}(k|k)\rangle} \quad (72)$$

It is easy to verify that the initial condition $\mathbf{S}(0|0)$ required for initiating (72) is given by

$$\mathbf{S}(0|0) = \frac{\boldsymbol{\Omega}(0)^*\mathbf{P}(0)}{\langle \boldsymbol{\Omega}(0), \mathbf{P}(0)\rangle} \quad (73)$$

where

$$\left.\begin{aligned} \boldsymbol{\Omega}(0) &= \begin{bmatrix} \Omega_1(0) \\ \vdots \\ \Omega_n(0) \end{bmatrix}, \Omega_i(0) = \sum_{j=1}^{n}\sum_{r=1}^{q} Z_{rji}(0)\, M_{rji}\pi_{ir}(0) \\ \mathbf{P}(0) &= \begin{bmatrix} P_1(0) \\ \vdots \\ P_n(0) \end{bmatrix}, P_i(0) = \text{prob}\,\{X(0) = x_i\}, x_i \in \mathbf{X} \end{aligned}\right\} \quad (74)$$

The prediction and smoothing problems can be treated in the same way.

An important problem concerning SSMM_5 is that of determining an optimal estimate $u(k+\theta|k)$ of the input $u(k+\theta)$ on the basis of the data Y_k. A similar problem was considered by Booth (1970).

Consider the case $\theta = 0$. We first compute the probability

$$\Phi_r(k|k) = \text{prob}\,\{u(k) = u_r | Y_k\}$$

$$= \sum_{j=1}^{n}\sum_{i=1}^{n} \text{prob}\,\{u(k) = u_r, x(k+1) = x_j, x(k) = x_i | Y_k\}$$

$$= \sum_{j=1}^{n}\sum_{i=1}^{n} \text{prob}\,\{x(k+1) = x_j | x(k) = x_i, u(k) = u_r\} \times \text{prob}\,\{u(k) = u_r | x(k) = x_i\}\,\text{prob}\,\{x(k) = x_i | Y_k\} \quad (75)$$

$$= \sum_{j=1}^{n}\sum_{i=1}^{n} M_{rij}\pi_{ir}(k)\, S_i(k|k) \quad (r = 1, 2, \ldots, q)$$

and then choose $\hat{u}(k|k)$ as:

$$\hat{u}(k|k) = \{u_r \in \mathbf{U} : \Phi_r(k|k) = \max\} \quad (76)$$

The predicted estimate $\hat{u}(k+\theta|k) : \theta > 0$ and the smoothing estimate $\hat{u}(k-\sigma|k) : \sigma > 0$ can be found in a similar way.

Clearly, $u(k) = h_5(x(k))$ represents a state feedback law, and since the function h_5 is nonlinear the expression $h_5(\hat{x}(k|k))$ does not always give the best estimate of $u(k)$. Equation (76) provides the best estimate of the feedback control input $u(k)$ on the basis of the available data Y_k and thus it is useful in all state feedback applications of stochastic sequential machines with inaccessible state variables.

7. Example

Consider the SSM

$$M = \langle \mathbf{W}, \mathbf{V}, \mathbf{U}, \mathbf{X}, \mathbf{Y}, \{p(w)\}, \{p(v)\}, f_5, g_5, h_5\rangle \quad (77)$$

where

$$\left.\begin{aligned} \mathbf{W} &= \{w_1, w_2\}, p(w_1) = 1/5, p(w_2) = 4/5 \\ \mathbf{V} &= \{v_1, v_2\}, p(v_1) = 6/7, p(v_2) = 1/7 \\ \mathbf{U} &= \{0, 1\}, \mathbf{X} = \{x_1, x_2\}, \mathbf{Y} = \{0, 1\} \end{aligned}\right\} \quad (78)$$

and the functions f_5, g_5 and h_5 are described by the following tables:

Table 1

$f_5 : \mathbf{U} \times \mathbf{X} \times \mathbf{W} \to \mathbf{X}$

	X	
U × W	x_1	x_2
$(0, w_1)$	x_1	x_2
$(0, w_2)$	x_2	x_1
$(1, w_1)$	x_2	x_1
$(1, w_2)$	x_1	x_2

Table 2

$h_5 : \mathbf{X} \to \mathbf{U}$

X	**U**
x_1	0
x_2	1

Table 3

$g_5 : \mathbf{X} \times \mathbf{X} \times \mathbf{U} \times \mathbf{V} \to \mathbf{Y}$

	V	
X × X × U	V_1	V_2
$(x_1, x_1, 0)$	0	1
$(x_1, x_2, 0)$	0	1
$(x_2, x_1, 0)$	1	0
$(x_2, x_2, 0)$	1	0
$(x_1, x_1, 1)$	1	1
$(x_1, x_2, 1)$	1	1
$(x_2, x_1, 1)$	1	1
$(x_2, x_2, 1)$	1	1

This machine (without the mapping h_5) was studied by Gelenbe (1971) who showed that it is equivalent to a stochastic Moore machine of the type

$$M' = \langle \mathbf{U}, \mathbf{X}, \mathbf{Y}, \{\mathbf{M}(y|u)\}\rangle$$

where

$$\mathbf{U} = \{0, 1\}, \mathbf{X} = \{x_1, x_2\}, \mathbf{Y} = \{0, 1\}$$

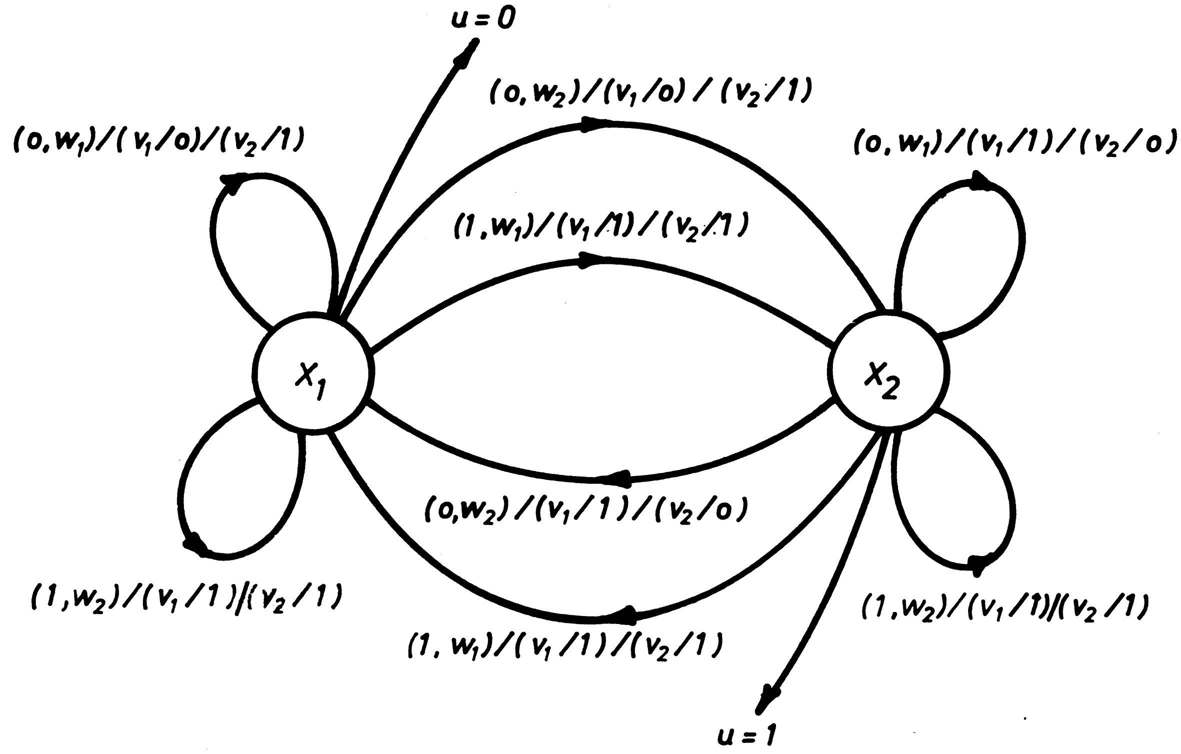

Fig. 3 State diagram of SSM M

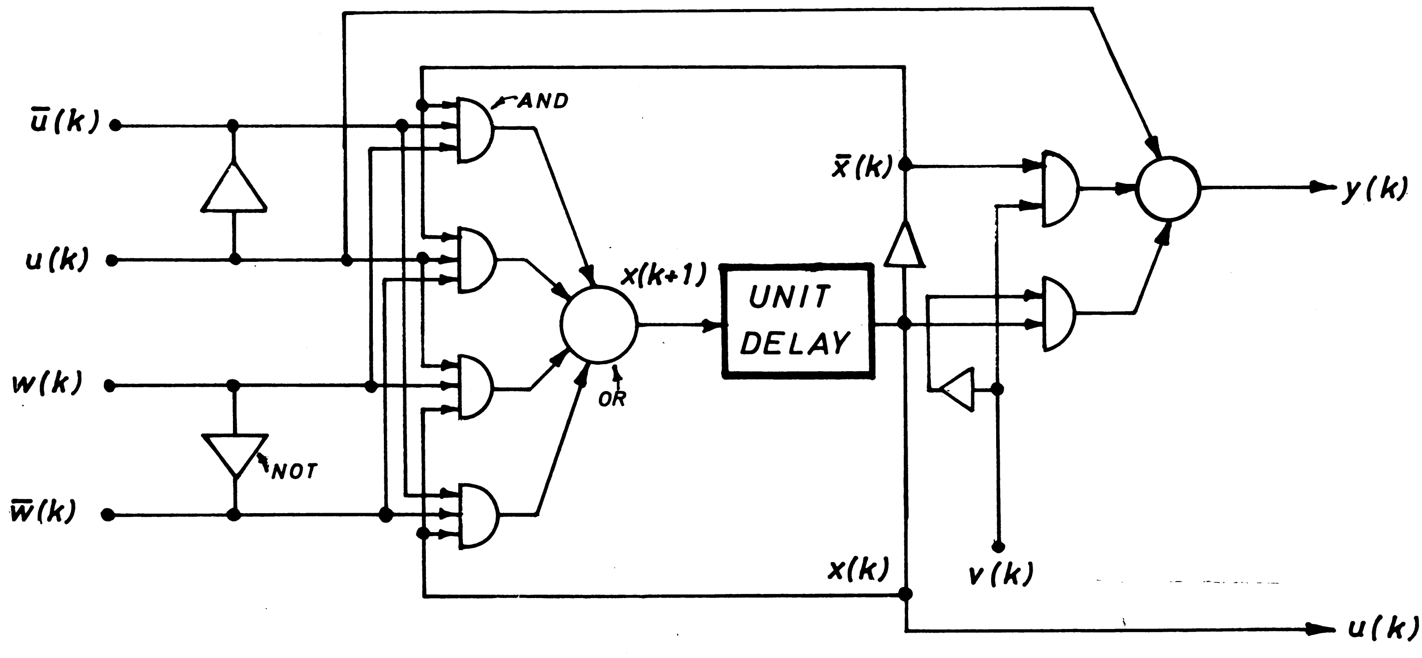

Fig. 4 Logic circuit for the realisation of SSM M

$$\mathbf{M}(0|0) = \begin{bmatrix} 6/35 & 24/35 \\ 4/35 & 1/35 \end{bmatrix}, \mathbf{M}(1|0) = \begin{bmatrix} 1/35 & 4/35 \\ 24/35 & 6/35 \end{bmatrix}$$

$$\mathbf{M}(0|1) = \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix}, \mathbf{M}(1|1) = \begin{bmatrix} 4/5 & 1/5 \\ 1/5 & 4/5 \end{bmatrix}$$

The element $m_{ij}(y|u)$ of $\mathbf{M}(y|u)$ in the above relations is interpreted as the probability that the machine M' started in state x_i will go to state x_j and produce output y for input u.

The state diagram of M is shown in **Fig. 3.**

Coding the values x_i, w_i, $v_i : i = 1, 2$ as follows

$$x_1 \to 0, x_2 \to 1; w_1 \to 0, w_2 \to 1; v_1 \to 0, v_2 \to 1 \quad (79)$$

we find from **Tables 1** and **3** that the logical equations of M are:

$$x(k+1) = \bar{u}(k)\, w(k)\, \bar{x}(k) + u(k)\, \bar{w}(k)\, \bar{x}(k) + \bar{u}(k)\, \bar{w}(k)\, x(k) + u(k)\, w(k)\, x(k)$$

$$y(k) = u(k) + x(k)\, \bar{v}(k) + \bar{x}(k)\, v(k)$$

Table 4

$u = 0$	$u = 1$
$M_{011} = p(w_1) = 1/5$	$M_{111} = p(w_2) = 4/5$
$M_{012} = p(w_2) = 4/5$	$M_{112} = p(w_1) = 1/5$
$M_{021} = p(w_2) = 4/5$	$M_{121} = p(w_1) = 1/5$
$M_{022} = p(w_1) = 1/5$	$M_{122} = p(w_2) = 4/5$

and so the machine M can be realised by the logic circuit shown in **Fig. 4**, in which the input disturbances $w(k)$ and $v(k)$ are generated by two independent random pulse sequence generators with probabilities as required by equation (73).

Using (78) and Table 1 we find Table 4

Similarly from the Tables 2 and 3 it follows that:

Table 5

$y = 0$	$y = 1$
$Z_{011} = p(v_1) = 6/7$	$Z_{011} = p(v_2) = 1/7$
$Z_{012} = p(v_1) = 6/7$	$Z_{012} = p(v_2) = 1/7$
$Z_{021} = p(v_2) = 1/7$	$Z_{021} = p(v_1) = 6/7$
$Z_{022} = p(v_2) = 1/7$	$Z_{022} = p(v_1) = 6/7$
$Z_{111} = 0$	$Z_{111} = 1$
$Z_{112} = 0$	$Z_{112} = 1$
$Z_{121} = 0$	$Z_{121} = 1$
$Z_{122} = 0$	$Z_{122} = 1$

and

$$\pi_{10} = 1, \pi_{11} = 0, \pi_{20} = 0, \pi_{21} = 1$$

Consequently applying equation (69) yields

$$\mathbf{M}_0 = \begin{bmatrix} 1/5 & 4/5 \\ 1/5 & 4/5 \end{bmatrix}$$

The *a priori* distribution is assumed to be

$$P_1(0) = P(x_1(0)) = 1, P_2(0) = P(x_2(0)) = 0 \quad (80)$$

and the data are given in **Table 6.**

Table 6

k	0	1	2	3	4	5
$y(k)$	0	1	0	0	0	1

With the aid of equation (71) we find that $\Omega_i(k + 1)$ are as follows:

Table 7

	$y(k + 1) = 0$	$y(k + 1) = 1$
$\Omega_1(k + 1)$	10/35	25/35
$\Omega_2(k + 1)$	0	1

Equation (68) reduces to:

$$\begin{bmatrix} S_1(k + 1|k) \\ S_2(k + 1|k) \end{bmatrix} = \begin{bmatrix} 1/5 & 1/5 \\ 4/5 & 4/5 \end{bmatrix} \cdot \begin{bmatrix} S_1(k|k) \\ S_2(k|k) \end{bmatrix}$$

from which it follows that

$$S_2(k + 1|k) = 4S_1(k + 1|k) \quad (81)$$

Similarly equation (73) yields

$$\begin{bmatrix} S_1(0|0) \\ S_2(0|0) \end{bmatrix} = \begin{bmatrix} 1 \\ 0 \end{bmatrix} \quad (82)$$

and equation (72) gives

$$\begin{bmatrix} S_1(k + 1|k + 1) \\ S_2(k + 1|k + 1) \end{bmatrix} = \begin{bmatrix} 1 \\ 0 \end{bmatrix} \text{ when } y(k + 1) = 0 \quad (83)$$

$$\begin{bmatrix} S_1(k + 1|k + 1) \\ S_2(k + 1|k + 1) \end{bmatrix} = \frac{\begin{bmatrix} \frac{25}{35} S_1(k + 1|k) \\ S_2(k + 1|k) \end{bmatrix}}{(25/35)S_1(k + 1|k) + S_2(k + 1|k)}$$

$$= \frac{\begin{bmatrix} 25/35 \\ 4 \end{bmatrix}}{(25/35) + 4} = \begin{bmatrix} 5/33 \\ 28/33 \end{bmatrix} \text{ when } y(k + 1) = 1 \quad (84)$$

Consequently using (19) we find that

$$\hat{x}(k|k) = \begin{cases} x_1 \text{ when } y(k) = 0 \\ x_2 \text{ when } y(k) = 1 \end{cases} \quad \text{i.e. } \hat{x}(k|k) = y(k) \quad k = 0, 1, 2, \ldots \quad (85)$$

and so the optimal filtered estimates corresponding to the data given in Table 6 are:

Table 8

k	0	1	2	3	4	5
$\hat{x}(k|k)$	0	1	0	0	0	1

Finally, using equation (75) we find that

$$\Phi_0(k|k) = S_1(k|k), \Phi_1(k|k) = S_2(k|k)$$

and so, by equation (76),

$$\hat{\mathrm{u}}(k|k) = \hat{x}(k|k), k = 0, 1, 2, \ldots$$

8. Conclusions

In this paper an attempt has been made to apply modern filtering techniques for estimating the state of stochastic sequential machines. Several sequential machine models were considered which cover the majority of practical situations.

It is believed that the results of the present paper should initiate more research in this field which is valuable in many engineering applications particularly in the digital computer, digital control and digital communication areas. The present paper is the first part of a project, the aim of which is to extend the conventional optimal control theory to finite-state machines.

Our next step would be to develop general methods for designing realisable finite-state filters amenable to LSI technology.

Of particular interest is the class of linear sequential machines over a field $GF(p)$ for which a well-defined algebra exists (Tzafestas, 1972, 1973).

It is useful to remark that the control results should find applications to discrete-time processes with quantised states, optimal design of time sharing systems, etc. In general we believe that stochastic estimation and control of finite-state machines is a versatile area for further research.

References

Bacon, G. (1964). The Decomposition of Stochastic Automata, *J. Inf. Control*, Vol. 7, pp. 320-339.

Booth, T. (1967). *Sequential Machines and Automata Theory*, John Wiley, N.Y.

Booth, T. (1970). Estimation, Prediction, and Smoothing in Discrete Parameter Systems, *IEEE Trans.*, Vol. C-19, pp. 1193-1203.

Bucy, R. (1968). Recent Results in Linear and Non-linear Filtering, Symp. of Amer. Auto. Control Council on *Stochastic Problems in Control*, Univ. of Michigan, Ann Arbor.

CARLYLE, J. (1963). Reduced Forms for Stochastic Sequential Machines, *J. Math. Anal. Appl.*, Vol. 7, pp. 167-175.
COX, H. (1964). On the Estimation of State Variables and Parameters for Noisy Dynamic Systems, *IEEE Trans.* Vol. AC-9, pp. 5-12.
EVEN, S. (1965). Comments on the Minimization of Stochastic Machines, *IEEE Trans.*, Vol. EC-14, pp. 634-637.
GELENBE, S. (1971). A Realizable Model for Stochastic Sequential Machines, *IEEE Trans.*, Vol. C-20, pp. 199-204.
HANDSCHIN, J. (1970). Monte Carlo Techniques for Prediction and Filtering of Non-linear Stochastic Processes, *Automatica*, Pergamon Press, pp. 555-563.
KAILATH, T. (1968). An Innovations Approach to Least-Squares Estimation (part I), *IEEE Trans.*, Vol. AC-13, pp. 646-660.
KALMAN, R. (1963). New Methods in Wiener Filtering Theory, Proc. 1st Symp. on *Engrg. Applications of Random Function Theory and Probability*, Bogdanoff, J. and Kozin, F. Eds., John Wiley, N.Y.
MEDITCH, J. (1967). On Optimal Linear Smoothing Theory, *J. Inf. Control*, Vol. 10, pp. 598-615.
MEDITCH, J. (1970). Newton's Method in Discrete-Time Nonlinear Data Smoothing, *The Computer Journal*, Vol. 13, pp. 387-391.
NIEH, T. (1970). On the Uniqueness of Minimal-State Stochastic Sequential Machines, *IEEE Trans*, Vol. C-19, pp. 164-166.
RAUGH, H. (1963). Solutions to the Linear Smoothing Problem, *IEEE Trans.*, Vol. AC-8, pp. 371-372.
STEARNS, R. and HARTMANIS, J. (1961). On the State Assignment Problem for Sequential Machines, *IRE Trans.*, Vol. EC-10, pp. 593-603.
TZAFESTAS, S. and NIGHTINGALE, J. (1969). Maximum-Likelihood Approach to the Optimal Filtering of Distributed-Parameter Systems, *Proc. IEE*, Vol. 116, pp. 1085-1093.
TZAFESTAS, S. (1972). Concerning controllability and observability of linear sequential machines, *Internat. J. Systems Science*, Vol. 3, pp. 197-208.
TZAFESTAS, S. (1973) State observer design for linear sequential machines, *Internat. J. Systems Science* Vol. 4. pp. 13-25

Book reviews

Representation and Meaning, by Herbert A. Simon and Laurent Siklossy, 1972; 440 pages. (*Prentice-Hall, New York*, £7·00)

This book is a collection of five Ph.D. theses from Carnegie-Mellon University on topics in artificial intelligence ranging over natural language processing, game playing, inductive generalisation and special purpose programming languages. All the theses were supervised by Simon, and three papers of his own, one jointly with Siklossy, are included. The book is thus comparable with *Semantic Information Processing* edited by Minsky from MIT, although not of the same degree of interest. The theses were completed between 1965 and 1969, and one can see how some of the ideas have contributed to more recent developments in natural language processing and programming languages. They read quite well, although as theses they inevitably have a little more padding than Journal articles would.

Simon's own contributions are very lucid, but the one on his 'Heuristic Compiler' is just a slightly extended version of a paper he published in the Journal of the ACM in 1963, and the one with Siklossy is a rather out-of-date review of natural language processing programs. I enjoyed his third paper 'On Reasoning about Actions' which manages in a dozen or so pages to raise several interesting speculations about methods of tackling 'state-action' problems.

Altogether I would say that this book provides some useful material for research work or course projects in artificial intelligence, but is rather unrewarding for the general reader.

Let me describe briefly the content of the theses.

T. G. Williams developed a 'general' game-playing program, trying it on a number of card and board games. It seems to be mainly a collection of useful subroutines to enable one to program game-playing, alleviating the tedious intricacies of programming in IPL-V. D. S. Williams developed an inductive program for taking intelligence tests of the sequence-extrapolation variety.

A program by Coles applies well-known syntax directed compiling techniques to the problem of interpreting natural language statements about pictures on a CRT display. Thus given 'Each polygon smaller than a triangle which is black is a square' it can test whether this is true. This is a substantial piece of work and has quite a lot in common with Winograd's recent celebrated program. It uses predicate calculus as an underlying language and is rather preoccupied with the English use of 'each' and 'all'. Siklossy, in his thesis, is more ambitious and has a program which learns to produce Russian or German sentences from a sort of parenthesised English with grammatical markers, thus (BE BOY [MOD THIS] HERE). Roughly, it learns to pick out foreign equivalents of English words from examples, and also deduces the surface structure equivalents in the foreign language of some deep structure trees. Ingenious, but we need to know a lot more before this could be more than a limited exercise.

On finishing the book I felt some admiration for the efforts of Simon and his students, but also disappointment that so little of clear general principle comes out of it all. There are approaches and feasibility demonstrations; we need definite discoveries which can generally be accepted as valuable.

R. M. BURSTALL (Edinburgh)

A Collection of Programming Problems and Techniques, by H. A. Maurer and M. R. Williams, 1972; 256 pages. (*Prentice-Hall Inc*, £3·50, paper)

One of the difficulties that faces anyone involved in teaching programming is the provision of a good supply of graded exercises for the student to tackle. Ideally they should have their sources in a wide range of topics, they should illustrate particular techniques, there should be specimen solutions to guide the student. All these are present to some extent in this book, yet I do not find it wholly satisfactory.

There can be no complaint about the variety of problems that are offered, ranging over simple mathematics, methods of sorting, strategies for playing simple games, statistics and networks (to name a few) in subject and from trivial to very difficult in standard. The specimen solutions exist only in the form of hints and numerical answers that could be expected for certain values of input parameters, there are no actual programs to show how a particular problem would be coded.

Each section has a brief introduction to the topic, followed by its selection of exercises. What I find disturbing is the way particular techniques are described, with little or no assessment of how they compare with others for the same problem, nothing about limitations on their use. For example, the section on solving equations gives the Newton-Raphson method, but not straightforward iteration, without even mentioning that the iteration may not always converge. Solution of linear equations recommends Gauss-Seidel iteration as the best method for more than 50 equations, again without comment that the matrix may not give guaranteed convergence. Cramer's rule, incidentally, is 'not recommended' only because it is said to be tricky to program, not because of inefficiency.

Altogether, a useful book for the alert teacher who will supplement and expand as necessary, not quite so good for the student on his own.

P. A. SAMET (London)