Int. J. Man-Machine Studies (1976) **8**, 337–365

Behaviour/structure transformations under uncertainty

B. R. GAINES
Man–Machine Systems Laboratory, Department of Electrical Engineering Science, University of Essex, Colchester, Essex, U.K.

(*Received 14 November 1975 and in revised form 15 March 1976*)

This paper analyses the problem of determining a structure for an automaton, optimal in some sense, from observations of its behaviour which are themselves uncertain. It is shown that extension of deterministic modelling techniques based on the Nerode equivalence to probabilistic sources gives meaningless results. The problem of approximate modelling with nondeterministic structures is rigorously formulated leading to the concept of a *space of admissible models*. The special case where the observed behaviour may be represented as a symbol string is then analysed in terms of measures of *string approximation*. It is shown that appropriate measures lead to the poorness-of-fit of admissible models of a probabilistic source being an *entropy* for that source. The formulation is consistent with a computational complexity basis for probability theory and leads to natural expressions for the *surprise* at each observation and the *uncertainty* as to the next observation. An implemented algorithm for this modelling process is then described with examples of its application to: probabilistic sources; sampled deterministic sources; grammatical inference; human behaviour; and program derivation from traces.

1. Introduction

The problem of deriving a *structure* for a system from observations of its *behavior* is ancient and fundamental with a vast literature. Philosophically the reality of such derived structures has long been questioned (cf. Plato's "shadows on the walls of a cave") and Hume's inductive scepticism indicates that predictions of future behaviour based on such structures cannot be justified on solely logical grounds. These philophical problems are bypassed in modern system theory by considering the class of possible structures to be prescribed and the problem to be one of *identifying* which member is most like that whose behaviour is observed (Zadeh, 1962; Eykhoff, 1974). A wide variety of classes of possible structures have now been studied and deep results and practical algorithms derived and used for the case where the observed system *does* belong to the postulated class. Recently many of these results have been uniformly expressed in elegant category-theoretic formulations (Arbib & Zeiger, 1969; Arbib & Manes, 1974; Goguen, 1973; Goguen, 1975) that encompass a wide diversity of system structures. However, a major question remains, of great practical importance, as to how these results and methodologies fare when the observed system does *not* belong to the postulated class—e.g. does the identification gracefully degrade, or become immediately absurd—in what sense can one structure be said to *approximate* another (Wharton, 1974).

In this paper I shall be concerned with the identification of systems as one of a class of *automata*, or state-determined machines. In particular I shall consider the problems of

identifying structure when either the behaviour of the system observed is partially acausal, or errors are made in observing it. It will be shown that there is a dramatic discontinuity in the problem of behaviour-structure transformation when the class of systems considered is changed from that of deterministic system to that showing even the slightest acausality, for example, of a probabilistic nature. Techniques that were completely adequate for the deterministic case do not just deteriorate in their utility but give completely meaningless results that can be highly misleading. Examples of this phenomena will be given and its derivation analysed.

The work to be described had its origins in an attempt (Gaines, 1971*a*, 1974) to develop a purely behavioural account of human behaviour as envisaged by Watson (Tolman, 1951) and Hull (1943) but developed by neither. The results are embodied in a suite of computer programs for behaviour-structure transformation and examples will be given of actual data analysis. Apart from their application to the original problem, the techniques developed are relevant to several related problems of general interest as follows.

(a) *Non-linear system identification* (Gaines, 1976*b*). Linear systems theory that has proved so powerful with artificial systems is often completely inapplicable to data from natural systems, particularly biological data—ethological observations are best regarded as a string of symbols to be modelled by an automaton structure (Dawkins & Dawkins, 1973, 1974)—examples are given of the analysis of human behaviour.

(b) *Learning machines.* The acquisition and use of knowledge about unknown sequential environments has been studied in the artificial intelligence and control literature as the problem of "learning machines" (Gaines & Andreae, 1966; Andreae & Cashin, 1969; Andreae & Cleary, 1976)—a major component of this problem is that of modelling the environment through observations of its behaviour—additionally, this is coloured by the "dual-control" (Feldbaum, 1963) or "two-armed bandit" (Witten, 1975) problem of having to control the environment whose behaviour is being observed and modelled—examples are given of the re-analysis of some of Andreae's data.

(c) *Automatic programming.* The technique of programming a computer by observing human beings solving the required problem and modelling their behaviour has now been extensively investigated (Biermann, 1972; Biermann, Baum, Krishdaswamy & Petry, 1973) as problems of inferring a deterministic automaton structure from observed behaviour (Biermann & Feldman, 1972)—however, if the person being modelled makes errors or uses a varying mixture of strategies, even though the overall "algorithm" is perfectly correct, the inference of a deterministic model leads to an unnecessarily massive program—examples are given of inferring program structures from their traces.

(d) *Computational complexity.* The definition of goodness-of-fit criteria for the approximate identification of automata leads to a natural measure of the complexity of sequences of behaviour—this is completely independent of probability theory but in the case where the source is a probabilistic automaton the expected value of the measure is an *entropy* for the source—the complexity measure has the advantage of being based on an ensemble of finite state automata and hence of being essentially computable.

The following section is concerned with modelling based on deterministic automata and the behaviour of the modeller when the source is not deterministic. Section 3 gives a new formulation of the general problem of identification, introducing the concept of an admissible subspace of models. Measures of string approximation are introduced that

enable the technique of identification to be rigorously and completely specified when the observed behaviour is a string of symbols, and the behaviour of the resultant identification scheme with probabilistic sources is analysed. Section 4 describes a computational algorithm based on this technique and illustrates its application to the identification of: a stochastic source; a sampled deterministic source; a deterministic source with unspecified inputs; a grammaticaly inference problem; human game-playing behaviour; and autoprogramming from traces. Section 5 summarizes the current state of this study.

2. Modelling with deterministic structures

The problem of identifying discrete deterministic systems from observations of their behaviour is important, both because it corresponds to a class of models that seem naturally postulated by the human observer and because it has been solved. Nerode (1958) defined an equivalence relation, essentially a congruence on the monoid of observable behaviours, that leads directly from a set of observed behaviours to a minimal-state finite-state automaton that accounts for them. The machine is unique up to an isomorphism and replicates precisely the behaviour observed—it is the "natural" structure for the observed behaviours in some sense, the *simplest* structure *cybernetically equivalent* (in the sense of showing the same input-output behaviour) to the observed system. In category-theoretic terms this result may be expressed (Goguen, 1973) more generally as an *adjunction* between categories of behaviour (in this case the monoids of possible behaviours and their homomorphisms) and structure (in this case finite automata and their dynamorphisms (Arbib & Manes, 1974)).

I have stressed the derivation of structure from *observed* behaviour in the previous paragraph because observation is essentially finite. The nomenclature of regular sets in automata theory allows for finite expressions describing infinite sequences (the "star" operation) and techniques based on the Nerode equivalence cope with this. However, they also demand the specification of a *complete* set of possible behaviours, an unrealistic requirement in practice. Clearly, however, in the actual observation of a system a unique minimal structure may be derived at each stage for the (finite) set of observations to that stage. A "differential calculus" of the variation of structure with successive observations would be of interest. It is clear that some observations will not affect the derived structure at all—they are "expected" on the basis of it and lead to no "surprise". Simple examples also show however, that a single observation may cause a massive change in structure, e.g. consider the sequence of observations consisting of 999 A's followed by a B—up to the thousandth observation a 1-state model suffices but when the B is observed a 1000-state (deterministic) model suddenly becomes necessary.

It is simple to construct an *optimum causal modeller* in terms of the Nerode equivalence, a system that forms a minimal-state deterministic automaton cybernetically equivalent to that generating an observed behaviour. Algorithms have been described in terms of the state-reduction of sequential machines (one can always start with a machine that has one state for each observation and then reduce it) that do this with the fewest possible operations (Hopcroft, 1971). In the following sections I will consider the behaviour of such a modeller, whether human being or computer algorithm, firstly when the observed system is actually a finite state deterministic machine and secondly when it is not.

2.1. BEHAVIOUR OF OPTIMUM CAUSAL MODELLERS WITH DETERMINISTIC SOURCES

An *observed behaviour* of length N is a sequence of N descriptors drawn from a set, D, i.e. a member of the free semigroup, D*, generated by D. For example ABAAB is a sequence of length 5 with initial symbol A and final symbol B. I shall eventually allow for a subset of D to be termed *inputs*, and also allow for non-consecutive sequences of descriptions by terming some elements of D *delimiters*. It will also become necessary to distinguish between modellers using Mealy or Moore type models. However, the following results are obvious for any optimal causal modeller and sequence of observations:

(1) the number of states in the model is a monotonic non-decreasing function of the length of the observed sequence of behaviour—the model cannot become simpler with further observation;
(2) the number of states in the model of a sequence of behaviour generated by an m-state deterministic automaton cannot exceed m—however it cannot become more complex than the observed system;
(3) the number of states cannot exceed the length of the sequence of observations—an obvious bound that we might expect to be ridiculously high;
(4) there are observed behaviours of length s such that the model must have s states (e.g. $A^{999}B$)—however, a bound that can be attained.

The first two properties seem to express the intuitive expectations of a human causal modeller—after a sufficiently long sequence of observations the model should stabilize. The last two properties show that it is possible for the size of model to grow at the highest possible rate, precisely as the number of observations. Clearly such growth cannot be sustained indefinitely if the system being observed is actually a finite state machine. For example, the behaviour, $ABA^2B^2A^3B^3\ldots$, could not be generated by a finite-state system and it would not be realistic to expect a finite-state model of it.

Gold (1967, 1971) considered the problems of modelling sequences generated by simple-recursive automata. He shows (1967) that there is a sense in which a modeller identifying from the class of simple-recursive machines can be correct "in the limit". However, there is no comparable theorem for a modeller drawing from the class of finite-state machines to model the behaviour of a simple-recursive system. Note, in terms of a differential calculus of models how even a sense of "approximation in the limit" is lacking—for example, a finite-state model of the sequence $ABA^2B^2A^3B^3\ldots.$ will essentially only "remember" it to date and in no way reflect its rather obvious structure.

2.2. BEHAVIOUR WITH PROBABILISTIC SOURCES

This lack of meaningfulness in the models of an optimal causal modeller using finite-state automata to model the behaviour of an infinite-state system is probably intuitively reasonable and acceptable. What, however, if the system observed is actually finite-state but not completely deterministic, for example a probabilistic automaton? We might expect, particularly if the source of acausality is slight, that the modeller would perform in a similar way to when modelling a causal system. Clearly a probabilistic source can generate sequences as exemplified in (4) requiring as many states in the model as the length of the sequence, but it is plausible to suppose that such "pathological" sequences might be generated only infrequently by a finite-state automaton.

Hence, a more interesting characteristic of the observer's behaviour would be, not the maximum complexity model it might generate, but instead the *expected* number of states

in the model generated (the average complexity). This might have one of three possible behaviours, as a function of the length of the observation sequence, s. The expected number of states in the model formed by an optimal causal modeller of the behaviour of a finite-state probabilistic automaton might be as follows.

(a) Asymptotic to a finite number, i.e. closely similar to the corresponding situation when the behaviour modelled is generated by a finite-state deterministic automaton.

(b) Grow without limit but slower than the number of observations, s itself, for example as log s. One might hypothesize that at least the ratio:

$$R_s = \frac{\text{expected number of states}}{\text{number of observations}} \tag{1}$$

would tend to zero as the number of observations increased.

(c) Grow without limit at a rate similar to the maximum possible, 1. This would imply that nearly all sequences generated were "pathological" requiring maximum-size models growing as fast as the number of observations.

Theories of probability based on computational complexity (Martin-Löf, 1966; Kolmogorov, 1968; Chaitin, 1969; Willis, 1970; Chaitin, 1975) show that case (c) in fact occurs, and it is probably a measure of the extent to which the result is counter-intuitive that this basis for probability has been so late in developing. We have so long accepted that a random sequence may have any structure, including possibly one that appears highly deterministic, that it comes as a shock when it is suggested that we can apply a test for randomness to an individual sequence rather than a distribution. The paradox is resolved of course because almost all randomly generated sequences may be shown to have high computational complexity (Willis, 1970).

The following additional property shows the significance of these results in the present context—case (c) occurs and ratio of equation (1) can tend to unity:

(5) the expected number of states in the model formed by an optimal causal modeller observing a sequence of behaviour of length s generated by a 1-state stochastic automaton may be at least s-$\log_2 s$—2—that is even a memoryless probabilistic source may require a model that grows on average as rapidly as the number of observations:

$$R_N = 1 - \frac{(\log_2 s + 2)}{s} \tag{2}$$

which is asymptotic to unity as N increases.

Gaines (1976*a*) obtains this result by considering a binary Bernoulli sequence with a generating probability of 1/2. Pearl (1975*b*) has extended this result using Shannon's rate distortion theory to other sequences and modelling situations. Gaines gives computed results of the actual behaviour of an optimal causal modeller for Bernoulli sequences with generating probabilities ranging from 0 through 0·02 to 0·5. It is apparent that even the smallest introduction of probabilistic acausality leads to the expected size of the model growing asymptotically proportional to the number of observations, and Gaines shows that:

(6) the ratio R_s of the expected number of states in the model formed by an optimal causal modeller of a binary Bernoulli sequence is asymptotic to at least 1/2 as s increases whenever the generating probability differs from zero or unity.

2.3. PSYCHOLOGICAL AND SCIENTIFIC ASPECTS OF COMPUTATIONAL COMPLEXITY

I have deliberately phrased the arguments of the previous suggestion to give the maximum impact to results (5) and (6) showing how the assumption of causality leads to large and meaningless models even when the system modelled is extremely simple provided it has the slightest degree of acausality. And yet this result will be of no surprise to those familiar with the computational complexity foundations of probability theory. To realize how far we have come in our attitudes, or at least our intellectual expectations, it is worth looking back to some of the philosophical and scientific literature on the role of probability and the principle of causality (Gaines 1976*c*).

Einstein's famous argument to Born against the probabilistic interpretation of quantum mechanics that "you believe in God playing dice" (Schilpp, 1949) illustrates the depth of feeling behind the assumption of determinism and causality. Nagel (1961, p. 324) argues that the assumption of causality is an "*analytical consequence* of what is commonly meant by "theoretical science" . . . . it is difficult to understand how it would be possible for modern theoretical science to surrender the general ideal expressed by the principle without becoming thereby transformed into something incomparably different". Popper (1972) argues that this in fact is precisely what has happened, at least in particle physics. However, in 1974 Suppes (1974) still finds it necessary to mount a vehement attack on what he calls the "new theology of science" that holds such tenets as: "Every agent has a sufficient determinant source" and "Knowledge must be grounded in certainty".

Philosophical arguments apart, there are also psychological grounds for supposing that the assumption of causality is basic and innate in man. At a perceptual level Michotte's (1963) famous series of illusions are convincing evidence that we cannot avoid *seeing* cause and effect even when we *know* it is not present. Gaines (1976*a*) outlines experiments with human beings in a game playing situation where the introduction of a random opponent leads to the development of complex models of the game—even to the hypothesis that a binary Bernoulli source is a "frustration automaton" (Gaines, 1971*b*) forming a model of its opponent in order to defeat him!

Thus the results of the previous section may have wide currency as an explanation of the pathology of both direct human behaviour and "science" when faced with acausal phenomena. Freud's (1914) complex development of a psychopathology of everyday life and dreaming may be seen as just what will happen if a principle of causality is assumed in modelling observations of random events. Thus the techniques developed in the next section, and indeed the whole methodology of a computational complexity basis for probability theory, may be seen as consistent with a new direction in the philosophy of science that runs counter to our innate preconceptions and intuition.

3. Approximate modelling with non-deterministic structures

The results of the previous section make it clear that we cannot obtain useful or even meaningful results by modelling the observed behaviour of an acausal system as if it were that of a deterministic automaton. However, if a more appropriate class of models is considered, such as that of non-deterministic finite-state automata, then the behaviour of the model is not completely determined and we no longer have the criterion of an exact match between model and observations—some measure of *approximation* is required. This is to be expected since, for example, in modelling the behaviour of a probabilistic

automaton with a structure that is precisely the same probabilistic automaton we are clearly unable to establish an exact match to any finite sample of behaviour. However, it does seem a reasonable objective to hope that, for example, our modelling of the zero-state Bernoulli sequences used as examples in the previous section would, over an increasingly long sequence of observations, converge upon a zero-state stochastic model as the "best" approximation.

There are two points to note in this concept. Firstly, that an $N+1$ state model will in any reasonable semantics be at least as good an approximation as an N state model. We need a criterion by which Ockham's razor may be applied to eliminate more complex models if their improved approximations are insufficient to justify their increased number of states. Secondly, approximation may be used for two quite distinct purposes: (a) to save resources in time and space, e.g. by accepting a sub-optimal, "almost-correct", solution; (b) because it is intrinsically necessary to the solution of the problem, e.g. an automaton exactly replicating the behaviour of a stochastic source can do so only by "memorizing" it—the source as a "correct model" of itself can replicate its own behaviour a second time only approximately. Pearl (1975*a*, *b*, *d*) has shown that the use of approximation in sense (a) is generally non-productive. In particular, if we accept approximate generation of an observed behaviour by a structure modelling it we do *not* gain significantly in the simplicity of the model until the approximation has become so great as to make the results meaningless. It is approximation of type (b) that is being considered here—an acceptance of a less than perfect match between structure and behaviour because a perfect one is intrinsically meaningless and demanding it transfers this meaninglessness to the results.

3.1. A GENERAL FORMULATION OF THE IDENTIFICATION PROBLEM

The general problem of behaviour-structure transformation may be formulated (Gaines, 1975*a*) in terms of: a set of possible observed behaviours, B; a set of models, M; the pointed monoid, $(Ord_{\mathrm{M}}, \leqslant)$, of all order relations on M with one specified relation, $\leqslant$, singled out; and a mapping, f: $\mathrm{B} \rightarrow Ord_{\mathrm{M}}$, from the set of behaviours, B, to the set of order relations on M, Ord_{M}. The quadruple, $(\mathrm{B}, \mathrm{M}, \leqslant, f)$, defines an *identification space:* the relation $\leqslant$ is one of model *simplicity* and if $m, n \in \mathrm{M}$ are such that $m \leqslant n$ we shall say that the model m is simpler than n—other considerations being equal it will be assumed that the simplest possible model is preferred—note, however, that $\leqslant$ may be only a partial order so that, in general, there will be a set of minimal models rather than a unique minimum; the mapping f is determined by the further order relation of *poorness-of-fit* that each behaviour induces on the set of models—we shall write, for $b \in B$, $\leqslant_b = f(b) \in Ord_{\mathrm{M}}$—if $m \in n \in \mathrm{M}$ are such that $m \leqslant_b n$ we shall say that model n is a poorer fit to behaviour b than is model m—the best models for b are thus those minimal in the order relation $\leqslant_b$ which again need not be more than a partial order.

Now we are in a position to define a solution of the identification problem—in terms of the product of the two order relations, $\leqslant$ and $\leqslant_b$, a new relation, $<_b^*$:

$$\forall\, m, n \in \mathrm{M},\ m \leqslant_b^* n \Leftrightarrow m \leqslant n \text{ and } m \leqslant_b n,$$

i.e. $m \leqslant_b^* n$ if and only if n is both simpler and a less poor fit than n. The minimal elements of the new order relation have the property that there are no other models that are both simpler and a better fit than them. Even if both $\leqslant$ and $\leqslant_b$ are total orders it is likely that $\leqslant_b^*$ will be a partial order (we can trade simplicity for improvement of fit) and hence

there will be in general no unique minimum model. The minimal elements are all *admissible* (Weiss, 1961; Kwakernaak, 1965) solutions to the identification problem because they cannot be improved in simplicity without increasing poorness-of-fit and cannot be improved in goodness-of-fit without increasing complexity. Thus we may define the solution of the identification problem for a space $(B, M, \leqslant, f)$ and an observed sequence $b \in B$ to be the *admissible subspace* determined by b and $M_b \subset M$ such that:

$$M_b \equiv \{m: \forall n \subset M, n \leqslant_b^* m \Rightarrow m \leqslant_b^* n\}.$$

This general formulation seems to encompass all practical identification problems. Note that, whereas for example the Nerode equivalence leads to a deterministic automation model of any finite sequence that is unique (up to isomorphism), in general there is no unique "best model" but instead a family of "admissible models". From its construction it is clear that this admissible sub-space is simply ordered both by "simplicity" and by "poorness-of-fit" and that there is an order-inverting transformation between them, i.e. if $m, n \in M_b$ and $m \geqslant n$ then $n \geqslant_b m$—if one admissible model is simpler than another then it is also a poorer fit. This order-inversion suggests a computational basis for determining the admissible models—if the models can be enumerated in order of decreasing simplicity then they may be searched consecutively and marked admissible if there is no model of greater or equal simplicity with lower poorness-of-fit. If the computation is halted at any stage then at least a well-defined set of the simpler admissible models has been determined.

The "solution" to the identification problem thus leaves unresolved the question of whether a simple model that is rather inaccurate in accounting for the observed behaviour is better, or worse, than a more complex model that gives a better fit to the observations. This residual dimension of "trade-off" between simplicity and accuracy is inherent to the problem and may be seen as a fundamental feature of all knowledge acquisition and theory-building under conditions of uncertainty. The general theory cannot resolve it without further postulates about the order relations, $\leqslant$ and $\leqslant_b$, e.g. that they may be represented by ratio scales. In the next section specific order relations are developed for the distances between strings of symbols generated by automata that, in the probabilistic case, may be used to reduce the problem of selecting the appropriate trade-off to a statistical decision as to whether the reduced simplicity of the model is "worth" the better approximation obtained.

3.2. APPROXIMATION OF STRINGS

Our criterion of approximation necessarily involves the string of observed behaviour and the matching of some string, or strings, produced by the model to it. A deterministic model produces a single, well-determined string of behaviour and we can apply a simple binary criterion to the result: either it is, or is not, the same as the observed behaviour. Approximate matches between strings of symbols have been studied in such diverse fields as the analysis of text-editing (Wagner & Fischer, 1974) and the derivation of common ancestors in genetic coding (Fitch & Margoliash, 1967; Sankoff, 1972). Metrics for "distances" between strings have been developed that are both untuitively meaningful and well-behaved mathematically, and computational algorithms for measuring such distances have been given (Sellers, 1974). In this paper all the classes of model considered are such that the model string is the same length as the modelled string, and the individual elements of both may be put in one-to-one correspondence. This is not necessarily

so in the general case and less-restricted approaches may be useful, for example, in automatic speech recognition.

The obvious measure of approximation to use between two strings in 1-to-1 correspondence is the number of differing symbols. This is a true distance measure in the space of possible strings (Sellers, 1974), and has properties of reflexity, positivity and transivity that make it theoretically tractable. The two strings:

Target:	A	B	C	A	A	B	B	C	C
Model:	A	B	C	A	B	C	A	B	C

for example, are 4 units apart. More formally, if we have two strings of s symbols, B^T and B^M, made up of symbols drawn from a set of descriptors, D, with ith elements $B^T(i)$ and $B^M(i)$ respectively, then the distance between them is:

$$E(B^T, B^M) = \sum_{i=1}^{s} \lambda(B^T(i), B^M(i)) \tag{3}$$

where λ is a two-argument function whose value is zero if its arguments are equal and 1 otherwise.

As noted, Pearl (1975*a*, *b*, *d*) has shown that no advantage may be gained by considering deterministic approximate models when the target string has been generated by a probabilistic automaton. However, a non-deterministic model will, in general, generate more than one model string and we have to extend the measure to be one between a target and many model strings. The minimum distance from one object to each of a set of objects is one well-known extension, but fails in this case because a model could then generate all strings and ensure zero minimum distance. The average distance of the target string from the set of model strings is a more promising measure. If we associate with each model string, B^M, a measure, μ^M, such that:

$$\sum_{\text{all M}} \mu^M = 1 \tag{4}$$

then the mean distance of B^T from the set of model strings is:

$$E(B^T, \{B^M, \mu^M\}) = \sum_{\text{all M}, i=1}^{s} \mu^M \lambda(B^T(i), B^M(i)). \tag{5}$$

This distance may be expressed in an alternative form more useful for the later discussion by re-defining the set $\{B^M, \mu^M\}$ in terms of a distribution over individual events. Let

$$\mu_j(i) = \sum_{\text{all M}} \mu^M \lambda(B^M(i), d_j) \tag{6}$$

where $d_j \in D$, the set of descriptors, now assumed indexed by the integer $j(1 \leq j \leq k)$, i.e. $\mu_j(i)$ is the distribution value assigned to the symbol d_j in the ith position in the sequence of model descriptions. We may now write (5) as:

$$E_\mu^T = E(B^T, \{B^M, \mu^M\}) = \sum_{i=1}^{s} \sum_{j=1}^{k} \mu_j(i) \lambda(B^T(i), d_j). \tag{7}$$

The form of (6) shows that, instead of regarding the modeller as proposing a number of different possible sequences, we may regard him as putting forward a single sequence,

not of symbols, but of distributions over possible symbols. If these were actual probability distributions generating the model strings then the measure E_μ^T in (7) would be the expected number of errors to be made by the modeller in matching the target string. Thus the string matching examined earlier might now appear as:

Event	: 1	2	3	4	5	6	7	8	9
Target	: A	B	C	A	A	B	B	C	C
Model	A: 0·1	0·1	0·2	1	0·1	0·4	0·4	1	0·5
distribution	B: 0·2	0·4	0·1	0	0·2	0·6	0·6	0	0·5
	C: 0·7	0·5	0·7	0	0·7	0	0	0	0

with $E = 0{\cdot}9+0{\cdot}6+0{\cdot}3+0+0{\cdot}9+0{\cdot}4+0{\cdot}4+1+1 = 5{\cdot}5$.

This formulation is interesting because it closely resembles the procedures used by Finetti (1972) and Savage (1970) to elicit subjective probabilities from human subjects and to give a rigorous formulation of probability theory based on such procedures. Finetti notes that if the target sequence is generated by a Bernoulli source and the subject gives a vector of numbers representing a distribution over possible symbols at each occurrence, then there is a measure of error that, when minimized by the subject, leads to him giving true probabilities. This is:

$$SE_\mu^T = \sum_{i=1}^{s} \sum_{j=1}^{k} (\lambda(B^T(i), d_j) - \mu_j(i))^2 \tag{8}$$

i.e. the sum of the squares of the differences between the proposed distributions and the actual event "distribution" (1 for the event which occurred and 0 for each of the others). Savage proved the same property for an alternative measure:

$$LE_\mu^T = -\sum_{i=1}^{s} \sum_{j=1}^{k} \lambda(B^T(i), d_j) \log_2(\mu_j(i)) \tag{9}$$

i.e. the sum of minus the logarithms of the components in the distribution of the elements that actually occur in the target sequence.

It has been shown (Aczel & Pfanzagl, 1966; Shuford, Albert & Massengill, 1966) that there is an infinite family of such measures with the property that a subject minimizing them is forced to give true probabilities in a probabilistic situation. Finetti (1972) showed this happened experimentally and the procedure has been used to assess "good probability assessors" in meteorology (Winkler & Murphy, 1968; Winkler, 1974) and to get maximum information about students' knowledge in multi-choice examinations (Shuford & Brown, 1975). Pearl (1975*c*) has recently given more meaning to the various measures that may be used to elicit subjective probabilities by relating them to possible hypotheses that the subject might make about the distribution of future pay-offs in what, to him, is a gambling situation. For example, SE corresponds to an exponential fall in future expected pay-offs and LE corresponds to the slower decay of a Cauchy density. The original measure proposed, E of (7), does not lead to the optimal modeller giving true probabilities, but is instead minimized by the modeller who gives *maximum-likelihood* estimates in a probabilistic situation, i.e. a distribution having the value 1 for the most likely event and 0 for all the others. Hence, it again corresponds to well-defined and well-known pattern of decision-making behaviour.

The most striking difference between SE and LE may be seen by contrasting them on the example given previously where E = 5·5:

$$\begin{aligned}
\text{SE} = {} & (0{\cdot}9^2+0{\cdot}2^2+0{\cdot}7^2)+(0{\cdot}1^2+0{\cdot}6^2+0{\cdot}5^2)+(0{\cdot}2^2+0{\cdot}1^2+0{\cdot}3^2) \\
& +(0^2+0^2+0^2)+(0{\cdot}9^2+0{\cdot}2^2+0{\cdot}7^2)+(0{\cdot}4^2+0{\cdot}4^2+0^2) \\
& +(1^2+0^2+1^2)+(0{\cdot}5^2+0{\cdot}5^2+1^2) \\
= {} & 1{\cdot}34+0{\cdot}62+0{\cdot}14+0+1{\cdot}34+0{\cdot}32+2+1{\cdot}5 = 7{\cdot}26 \\
\text{LE} = {} & -\log_2(0{\cdot}1)-\log_2(0{\cdot}4-\log_2(0{\cdot}7)-\log_2(1)-\log_2(0{\cdot}1) \\
& -\log_2(0{\cdot}6)-\log_2(0{\cdot}6)-\log_2(0)-\log_2(0) \\
= {} & 3{\cdot}32+1{\cdot}32+0{\cdot}51+0+3{\cdot}32+0{\cdot}74+0{\cdot}74+\infty+\infty = \infty.
\end{aligned}$$

The logarithmic measure will not tolerate the situation where an event is given a valuation of zero but then occurs—the error then becomes infinite, whereas both E and SE give large but finite errors in this situation. The logarithmic measure is also distinguished in that it depends only on the valuation given to the event which actually occurred regardless of the distribution over the other components. This has been taken by some writers as a desirable feature although the argument seems dubious and there are more meaningful considerations that make the logarithmic measure attractive (see following section).

One important feature of the reformulation of (5) leading to (6), (7) and (8) is that the poorness-of-fit measure can now be viewed as accumulated event by event in the sequence. The component added for each event may be described as the *surprise* caused by that event. All three measures agree that the surprise caused by an event given the valuation 1 which actually occurs is zero (e.g. event 4 in the example). They give varying weights to events which would occasion little surprise (e.g. event 3) or much surprise (e.g. event 1) and, as noted, the logarithmic rule expresses infinite surprise at an event that occurs when the valuation given to it is zero. This valuation of "surprise" is consistent with the model of decision-making based on "potential surprise" proposed by the economist Shackle (1955, 1961), and is particularly useful in on-line learning algorithms where a marked increase in the rate of surprise may be used to indicate the need for the re-computation of the model.

3.3. PROBABILISTIC MODELLING OF PROBABILISTIC SYSTEMS

I have deliberately avoided the use of terms such as "estimated probabilities", "subjective probabilities", etc., for the distributions over model strings or individual symbols put forward by the modeller in the previous section. The results of Finetti, Savage, *et al.*, indicate that *if* the actual event sequence is probabilistically generated then a modeller that is optimal (in the sense of minimizing the poorness-of-fit measures, SE or LE) will be forced to put forward the actual generating probabilities of events. This result is an important link between "subjective" and "physical" or "frequentist" foundations of probability theory. It is equally important as a link between our general approach to system identification and probabilistic modelling. However, the measures defined in the previous section and the identification techniques based on them do *not* entail a hypothesis of probabilistic acausality. The fact that they behave meaningfully and well when used with probabilistic systems is clearly desirable, even essential, but there is no converse argument that they are based on a hypothesis of probabilistic behaviour in the system modelled.

Clearly, we may now expect to obtain results for probabilistic modelling (optimality of identification techniques, decision criteria for selecting amongst admissible models, etc.) which do not necessarily apply in more general cases—indeed are not meaningful unless further hypotheses are made about the more general case. Clearly also, there are few hypotheses comparable in power and significance to that of a probabilistic generator —we have examined some examples of asynchronous systems modelling where no probabilities are definable but it is possible to obtain weaker, structural rather than numeric, results for identification techniques based on the measures of poorness-of-fit defined. An example of non-probabilistic acausality will be given in section 4.3. where several samples of the behaviour of a deterministic system are identified—the acausality arising through the sampling process and having no numeric, probabilistic significance in the model. In the remainder of this section, however, some results will be derived for the case where both model, and system modelled, are probabilistic automata.

What we would like to derive are comparable results to those of section 2.1 for an optimal causal modeller observing finite-stage deterministic automata, but now for an optimal probabilistic modeller observing finite-state probabilistic automata. It is convenient to split the problem into two parts: (a) considering the successive probability distributions themselves as a sequence of descriptions to be matched; (b) considering under what conditions such a match could be made. Moore type automata (in which outputs are associated with states) will be assumed in the ensuing discussion but the results are readily transferred to Mealy type models (outputs associated with transitions) and the computational algorithms provide for either. Autonomous automata will also be assumed and the extension to automata with inputs will be considered later—again the algorithms to be described provide for certain symbols to be designated as inputs.

Any probabilistic automaton may be regarded as outputting a distribution over possible outputs at each step (this is not identical in concept to an ensemble of all possible transitions, but rather more artificial since we are following the state-distributions of a single automaton). Suppose that the modeller is able to put forward precisely the same distributions and consider the expected values of E, SE and LE—we have:

$$\hat{\mathrm{E}} = \sum_{i=1}^{s} \sum_{j=1}^{k} p_j(i)(1-\mu_j(i)) = \sum_{i=1}^{s} \sum_{j=1}^{k} p_j(i)(1-p_j(i)) \tag{10}$$

$$\hat{\mathrm{SE}} = \sum_{i=1}^{s} \sum_{j=1}^{k} p_j(i)(1-\mu_j(i))^2 - \mu_j(i)^2 + \sum_{h=1}^{k} \mu_h(i)^2)$$

$$= \sum_{i=1}^{s} \sum_{j=1}^{k} p_j(i)(1-p_j)(i)) \tag{11}$$

$$\hat{\mathrm{LE}} = \sum_{i=1}^{s} \sum_{j=1}^{k} p_j(i)(-\log_2(\mu_j(i)))$$

$$= \sum_{i=1}^{s} \sum_{j=1}^{k} -p_j(i) \log_2(p_j(i)) \tag{12}$$

where s is the length of the sequence; k is the number of different symbols; $p_j(i)$ is the probability of symbol j at event i; $\mu_j(i)$ ($= p_j(i)$ by assumption) is the value put forward by the modeller for symbol j at event i.

Equations (10) through (12) show that the expected value of the residual poorness-of-fit when the modeller matches the system probabilities is an *entropy* function for all three measures—in particular LE of equation (12) is the familiar Shannon (1948) entropy function. The coincidence in values of E and SE is interesting but spurious since, as noted previously, the condition $\mu_j(i) = p_j(i)$ gives a minimum for the functions SE and LE in (11) and (12) whereas E is minimized when:

$$\mu_j(i) = \begin{cases} 1 \text{ if } j = h \\ \\ 0 \text{ otherwise} \end{cases} \tag{13}$$

where h is any value such that

$$p_h(i) = \max_j (p_j(i)),$$

a maximum-likelihood estimate. Thus Finett's quadratic SE in (8) has the same expected residual error as the more obvious definition of a poorness-of-fit measure, E in (7), but forces true probability estimation on the optimal modeller which use of E does not.

Now consider the problem of how a modeller might actually come to match a sequence of distributions. The two sources of difficulty are as follows.

(a) Since the sequence is not Bernoulli and the distributions change from event to event the modeller must be able to locate himself in the sequence, i.e. the events with different distributions must be *observable*. This is precisely the same condition as with deterministic modelling where no modeller can discriminate between two different structures if their reduced, observable forms are isomorphic. In practice, for a probabilistic model, this condition implies simply that, even though from a state we can predict only the probability of the next state, after the transition the output must be sufficient to indicate the actual state. Thus in analysing the match between source and model we need only consider the reduced form of the source—this appears in result (2) of section 2.1 as the number of states in a model cannot *exceed* those in the source rather than becomes eventually equal to their number.

(b) The distributions themselves are not the actual outputs an an indefinitely large sample of actual outputs is necessary in order to estimate them. Thus the distributions in any *transient* behaviour of the source automaton cannot be accurately estimated, only those in its recurrent behaviour.

Combining these two factors we can see that it is realistic to consider matching in a model only the recurrent behaviour of the reduced form of a probabilistic automaton. The recurrency is an additional constraint compared with deterministic modelling and clearly a reasonable one in the circumstances. Conceptually the modeller of an observable sequence of distributions is applying a Bernoulli sequence modelling strategy to each distinct distribution. However, he has both to discover the observation algorithm and estimate the distributions.

Consider the formulation of the identification problem given in section 3.1 with the identification space, (B, M, $\leqslant$, f), with: M the space of probabilistic Moore automata in reduced form; $m \leqslant n$ if m has less states than n; the value $f(b) = \leqslant_b$ being defined for a sequence of behaviour b by $m \leqslant_b n$ if $\mathrm{LE}^b{}_{\mu m} \leqslant \mathrm{LE}^b{}_{\mu n} + s\varepsilon$ where μ_m and μ_n are the distributions arising from modelling m and n respectively, s is the length of b and ε is a small tolerance to allow for statistical fluctuations (an alternative f may be similarly

defined based on SE instead of LE). Now consider the behaviour of the admissible subspace for a sequence of observations of increasing length of a finite state probabilistic source.

A result equivalent to (1) of section 2.1 could be that the maximum number of states in the admissible models is monotonic non-decreasing. However, this is not so because in the short term the particular sequence generated may be such as to justify complex models. However, there is a result equivalent to (2) as follows.

(2′) For any ε, for a given probabilistic source, the maximum number of states of an admissible model cannot eventually exceed that of the source as the sequence of observation increases. This follows because the properties of LE are such that averaged over a long sequence of recurrent states any modeller cannot do better than put forward precisely the distributions of the reduced form of the source and hence the source itself will have at least as low a value of LE as any model with a greater number of states.

We still cannot show that the maximum number of states is precisely that of the source, even in reduced form, because the modelling procedure cannot accurately identify the transient behaviour of the source. Our definition of $\leqslant_b$ based on LE and $s\varepsilon$ means that in the long term the transient behaviour will have a decreasingly small effect so that eventually the admissible models neglect it. The maximum number of states in an admissible model will then correspond to the number of states in the recurrent part of the automaton generating the observed behaviour (there is clearly no well-defined "recurrent part" in general since we may enter different recurrent parts after the same initial transient). What of the admissible models with less than the maximal number of states? These correspond to the "lumping" of states in a Markov process and will inevitably give higher values for the entropy of the process and hence LE. They are best approximations to the source in the sense that they minimize the deviation in behaviour from that of the actual source.

4. Identification algorithms and results

The formulation of the identification problem given in 3.1 is applicable to a very wide class of modelling procedures including most grammatical inference schemes to date (Fu & Booth, 1975*a*, *b*). The string approximation measures of section 3.2 are also widely applicable whenever behaviour is representable as a linear string (or set of strings—the required extension will be covered in this section). It is not necessary for the class of models to be probabilistic automata—we have used simple push-down automata for some studies and string-pattern based models (based on Andreae & Cleary's studies (1972–5, 1976)) for others. One only needs a set of models that is complete in the sense that it can offer a good enough approximation to any possible behaviour, and that is intrinsically ordered in complexity in some intuitively meaningful way and extrinsically ordered in poorness-of-fit by any behaviour. This lack of constraint upon families of models is important because it emphasizes the arbitrariness of our choice in choosing, for example, probabilistic automata—science tends to make its major advances through changes of viewpoint entailing changes in the family and evaluation of acceptable models rather than through incremental improvements in approximation. Our general-purpose identification scheme searching a single space of models is probably *not* an adequate model of inductive inference on this account, although as Solomonoff (1964*a*, *b*) and Watanabe (1969) have argued such schemes do account for some aspects of inductive reasoning.

4.1. THE ATOM MODELLING AND PREDICTION SYSTEM

In the remainder of this paper, I shall concentrate only on identification algorithms in which the class of models M is that of finite-state probabilistic automata with the relationships, $\leqslant$ and $\leqslant_b$, defined as in section 3.3 by the number of states in the model and the logarithmic poorness-of-fit criteria, LE, respectively. The computational algorithms to determine the admissible models are one of a suite of programs called "ATOM" written in the interpretive, string-handling language BASYS (Gaines & Facey, 1975) on a time-shared PDP11/45 (the algorithms have also been used in FORTRAN on a PDP10 with KA10 processor and run some 50 times faster). ATOM provides facilities for interactively entering observed data and forming on-line predictions from models, and so on. However, for the automata modelling studies it is generally used in restartable background batch mode since computational runs of hundreds of hours are involved.

A behaviour to be modelled is input to ATOM as a string of arbitrary character strings separated by spaces or end-of-line terminators. Thus:

```
MARY HAD A LITTLE LAMB
ITS FLEECE WAS ?
```

is a sequence of behaviour consisting of 9 symbols, and:

```
I=2
P=A(I)
J=P/I+7
```

is a sequence of behaviour consisting of 3 symbols. This acceptance of free format strings is particularly helpful in some examples such as natural language processing and automatic programming.

ATOM assumes that all the symbols are automaton *outputs* unless it is separately informed that a certain set of symbols are *inputs* and/or another set are *delimiters*. All the modelling schemes treat these two classes in a similar fashion: *inputs* are not brought into the string approximation measurement, i.e. one does not evaluate the extent to which input symbols are predicted correctly; *delimiters* are taken to indicate that the string before the delimiter may not be used to predict that after it—in the automaton models a delimiter causes a reset to the initial state of the model. Otherwise both inputs and delimiters are treated as any other symbols in the string of behaviour. Note that the availability of delimiters enables separate samples of behaviour (separate sentences say in a grammatical inference problem) to be freely concatenated together, separated by delimiters, to form a single sequence of "behaviour". Note also that the modelling process does not necessitate inputs and delimiters being specified in this way. If they are then the computation is faster, but if they are not then their nature may be inferred from the results—i.e. inputs are "outputs" that cannot be predicted and delimiters are those which appear as a general reset—examples of such inferences will be given later.

The automaton identification subprogram in ATOM generates, for a given behaviour, the admissible subspace of either Mealy or Moore probabilistic automata, as requested, commencing with 1-state models (Mealy) or k-state models (Moore—where k is the number of different symbols in the behaviour). The actual output of the program is thus the set of best-fit 1-state models, the set of best-fit 2-state models, etc. The search ceases when no more admissible models are found, but in practice this condition rarely arises

since the search space for larger models becomes very large and the program is terminated by lack of time rather than by completion of the search. However, since the simpler admissible models are output first, the modelling is always complete up to models with the number of states at which it was terminated.

The search procedure is essentially simple because *only the space of non-deterministic automata has to be searched,* not that of probabilistic automata, i.e. the transitions are initially regarded as being only present or absent. When a non-deterministic model of the behaviour has been generated the actual transition probabilities are filled in from the relative frequencies of the transitions in the particular model with the given behaviour. This is legitimate because these values are known to minimize the measures LE and SE. LE and SE for the model/behaviour pair are then calculated and used to ascertain the poorness-of-fit relative to previous models generated. If the poorness-of-fit on either criterion is the same, or better, than that of the best models previously formed then the new model is added to the set of potentially admissible models. Any models with the same number of states but a poorer fit on both criteria are discarded. The search then continues. Whenever the models with a given number of states have been searched then the remaining best models with that number of states are filed as being admissible. The values of E (assuming maximum-likelihood estimates), SE and LE are filed with the model.

The generation of models is basically an exhaustive enumeration of all possible observable non-deterministic automata. However, some care is necessary to avoid duplication and to take advantage of any structural features of the sample behaviour (e.g. some symbols never following other symbols). Models are generated using the actual behaviour to fill in state transitions. The initial model is a 1-state automaton and, if N-state models are being searched, N is taken as a bound on the number of states. The initial state has to be the 1 and only state. Each symbol in turn in the behaviour is then examined. If it corresponds to an existent transition no action need be taken. If there is no transition corresponding to it then one is filled in to the first state and a marker placed on a stack. The state is then advanced to its next value and the next symbol checked.

Eventually a model has been formed and may be evaluated for SE and LE. A backtrack is then made by taking off the stack the last transition entered and, if it is to state k, changing it to be to state $k+1$ and continuing as before. However if state k was a new state then it is removed and backtracking performed again, or if k was the last state and not new, and k is less than the allowed maximum number of states, then a new state is added and the transition entered to this. Eventually backtracking is no longer possible and all models with the allowed number of states have been generated without duplication and without considering transitions not necessitated by the behaviour being modelled.

"Delimiter" symbols are taken to cause a reset to the initial state. "Input" symbols are not taken into account in the calculations of the poorness-of-fit measures.

The following sections contain examples of ATOM automaton modelling in action.

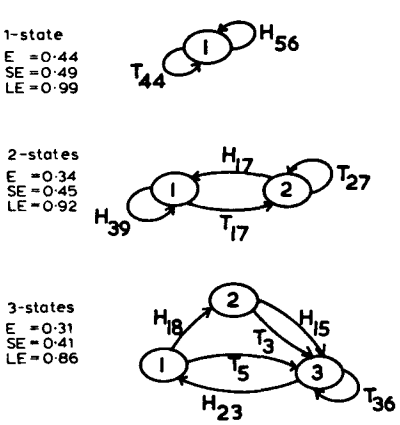
4.2. EXAMPLE—A BIASED GAMING MACHINE

The first example is the identification of a simple stochastic automaton with binary output. It is taken from a delightful example given in the form of an anecdote by Andreae (1972–75, No. 4, p. 122) about a gaming club owner who suspected a penny tossing machine of being rigged. A sample observed behaviour of 100 heads and tails from the

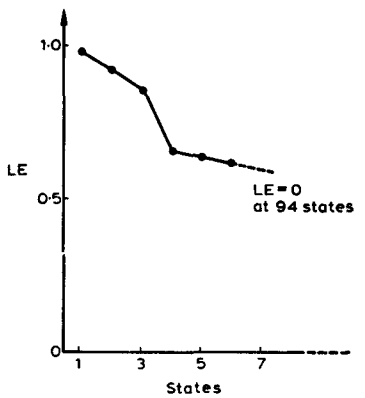
machine is given at the top of Fig. 1 (I suggest for each example that the reader try to evaluate the sequence themselves—most of the results are simple in retrospect but not in prospect!).

H H T H H T H H H T T H H H H T T H T T T H H H H T H H H T T T
H H H T T T H H H H H T T H T T T T T H T T H T T T H H H T T H
H H T H H H H H H H H T T H H H T T T T H H H H H T T T T T T H

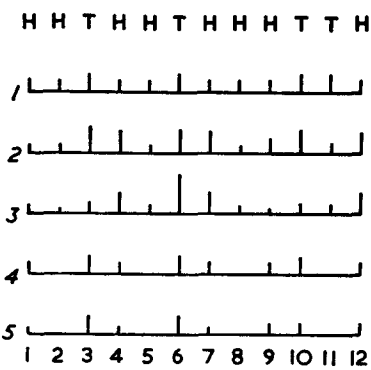
(a) Observed sequence of behaviour—100 events.



(b) Admissible models (1 through 5 states).



(c) Admissible models poorness-of-fit (logarithmic measure, LE).



(d) "Surprise" at each of first twelve observations for 1 through 5 state admissible models.

FIG. 1. Identifying a stochastic source.

Figure 1(b) shows the admissible models generated with 1 through 5 states—the subscripts to the H's and T's on the transitions indicate the number of times that path is followed—the initial state is always state 1. The values of E, SE and LE given are normalized with respect to the length of the sequence and that of E is with respect to maximum-likelihood predictions (i.e. it represents the proportion of predictions that would be wrong if either H or T had to be predicted at each event). The close covariation of SE and LE will be noted—either measure may be used to define the admissible set and we use LE in practice. E does not necessarily always vary monotonically with the others,

i.e. sometimes a lower error score with maximum-likelihood predictions is obtained through a higher entropy model.

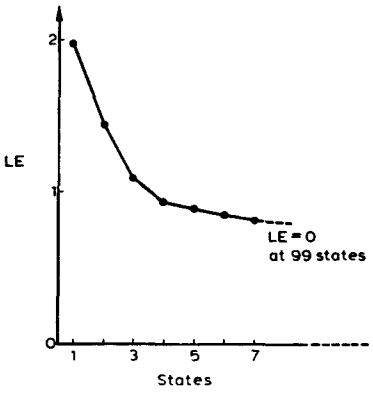
Figure 1(c) is a plot of LE for the admissible set of 1 through 6 state automata. It is asymptotic to zero with a deterministic model of 94 states, i.e. about that expected according to the bounds of section 2.2 for a Bernoulli sequence with generating probability of 1/2. Note the sudden drop at a 4-state model. This indicates that some structure has been found in the observed behaviour, and it can be seen that the 4-state model of Fig. 1(b) has two deterministic branches.

Figure 1(d) allows the source and effect of this structure to be clearly seen. It is a plot of the "surprise" expressed by each of the models shown in Fig. 1(b) at each of the first 12 events in the sequence of observed behaviour (i.e. minus the log of the probability ascribed to the symbol that occurs). As we go from 1 to 4 states a pattern of surprise becomes evident until, at 4 states, no surprise is evinced at each 3rd symbol (events 2,5,8,11), i.e. each 3rd symbol is a perfectly predictable. The 4-state model in fact shows that every 3rd symbol is a repetition of the preceding symbol. The 5-state model contribites no further information—it can be seen that it just splits state 4 of the 4-state model so that H and T go to separate states.

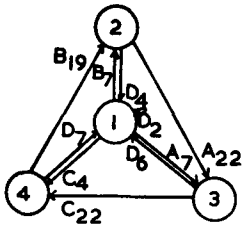
The source of Andreae's anecdote is now clear—here is an apparently random gaming machine at which we should achieve 50% success but can achieve 67% success if we know how!

A D B A D A C B D C B A C B D B A C B A C D C D A C B A D
A C B D B A C B A C B A C D B A D A C B A C B A C B A D A
C B A C B A C B A D C D B A C B A D B A C D B D C B A C D
A C B A C B A C B A C D D A

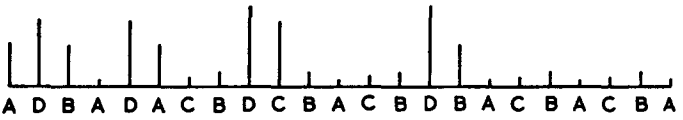
(a) Observed sequence of behaviour—101 events.



(b) Admissible models poorness-of-fit (logarithmic measure, LE).



(c) 4-State admissible model.



(d) "Surprise" at each of first 23 observations for 4-state model.

FIG. 2. A sampled deterministic source.

4.3. EXAMPLE—SAMPLED DETERMINISTIC MACHINE

The observed behaviour for the next example is shown at the head of Fig. 2 and contains 100 symbols drawn from A,B,C and D. Again it is of interest to ascertain how readily the human brain can comprehend the behaviour. In fact it is that of a deterministic machine generating the repetitive sequence CBACBACBA etc., sampled in short segments of arbitrary length with the symbol D inserted as a delimiter at segment boundaries; it consists of 19 samples of behaviour concatenated together between separators.

Figure 2(b) shows the fall of LE with number of states for admissible models and a turnover at 4 states is apparent. The transition diagram of Fig. 2(c) shows the corresponding 4-state model and the dominant CBA cycle is immediately apparent. Superimposed on this are some less frequent "noise" transitions of which the most notable are those involving D since it may be seen that D may occur from any state, always goes to state 1, and from this state any symbol may be generated. This indicates that D is a delimiter. The automaton itself is a "Moore" model in which state 1 corresponds to D, state 2 to B, state 3 to A, and state 4 to C. Figure 2(d) shows the "surprise" evinced by this model at each of the first 23 observations. It can be seen that D and the symbol after it are both "surprising" but the rest of the symbols, within the sampled behaviour, are not.

Note the occurrence of a "Moore" model with outputs associated with states stems from the problem structure, not the modelling process. Similarly the inference that D is a delimiter is inferred from the models not imposed. This affords a useful illustration of the value of information: there are 100 million 4-state Mealy models; this reduces to 100,000 if D is specified as a delimiter; and further reduces to 1 if only Moore models are searched. The value of correct structural hypotheses as to the class of possible models is thus enormous in saving computation. However, as shown in section 2.2, if the hypothesis turns out to be incorrect then the result may be, not just an approximation, but totally meaningless.

Note also that there is no question of "probabilistic inference" from this source. The sampling is irregular but it was not done probabilistically and shows non-stationarity in that the initial sequences are shorter. The poorness-of-fit measures are logical and ensure the correct result without assumptions of an underlying probabilistic process.

4.4. EXAMPLE—INFERENCE OF INPUTS

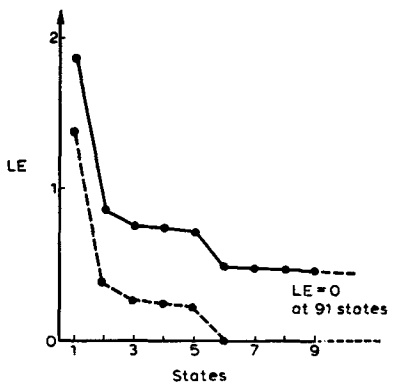
The third example (Fig. 3) is again taken from Andreae (1972–75, No. 2, p. 97) and is in fact the observed behaviour of a deterministic automaton with outputs P and Q and inputs A and B. It has been analysed by ATOM with, and without, the "input" symbols being specified. The upper poorness-of-fit plot in Fig. 3(b) (analysis with no "inputs" specified) shows a marked dip at 2 states where the basic output/input/output/input.... structure is picked up. The next dip is at 6 states where the actual deterministic structure is completely resolved [Fig. 3(c)] and only the (unpredictable) inputs remain indeterminate. The "surprise" pattern for this model is clearly zero for an "output" (P or Q perfectly predictable) and about unity ($-\log_2(\frac{1}{2})$) for an "input" (A or B unpredictable) so that the input/output distinction is readily inferred.

Informing ATOM that A and B are inputs drops them from the poorness-of-fit calculations so that the plot reaches zero at 6 states [lower plot of Fig. 3(b)] and the admissible set is then complete. There is no computational speed-up involved because the space of models searched remains the same but, of course, a clear-cut decision is reached at LE = 0 for a 6-state model and no further models need be investigated.

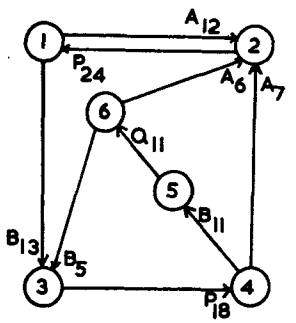
This example illustrates the importance of the model space search being "driven" by the observations. The majority of possible models with n states over four symbols are not generated because they do not fit the ((P|Q)(A|B))* pattern. With exhaustive search it would not have been possible to investigate up to 9-state models in a reasonable time.

P A P B P B Q B P A P B P B Q A P A P B P A P B P B Q B P B
Q A P A P A P A P B P B Q A P B P A P A P A P B P B Q B P B
Q A P B P A P A P A P B P B Q B P B Q A P B P A P B P B Q B
P A P A P A P A P B P B Q A P B P A

(a) Observed sequence of behaviour—108 events.



(b) Admissible models poorness-of-fit (upper plot —"inputs" unspecified; lower (dashed) plot— "inputs" A and B specified).



(c) 6-State admissible model.

FIG. 3. A deterministic source with inputs.

4.5. EXAMPLE—A GRAMMATICAL INFERENCE PROBLEM

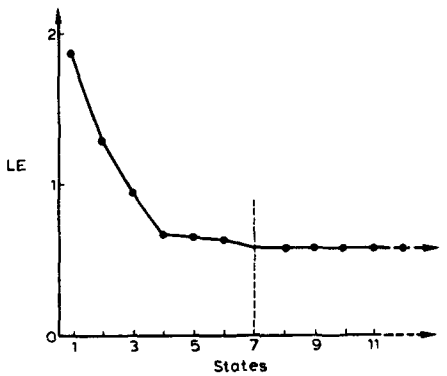
The fourth example is a simple grammatical inference problem analysed by Evans (1971) and ascribed to Feldman, Gips, Horning & Reder (1969). The 7 sample strings are: {CAAAB, BBAAB, CAAB, BBAB, CAB, BBB, CB}, and these are fed to ATOM concatenated together with / as a separator as shown in Fig. 4(a). LE for the admissible models falls with the number of states as in Fig. 4(b) showing a major break at 4 states and a very minor one at 7 where it reaches a final asymptotic value (strictly, there are no admissible models with more than 7 states).

Figure 4(c) shows the corresponding admissible models with 1,2,3,4 and 7 states, and the corresponding regular sets defining the languages they specify. The 4-state machine corresponds precisely to the grammar derived by Feldman *et al.* (1969) and Evans (1971)—in the ATOM results its special status is apparent from the sudden change in the LE plot at 4 states. The further decline at 7 states corresponds to a model and grammar generating a regular set that is precisely that put in. Note that the improvement in entropy of the 7-state model over the 4-state one does not appear significant—the inferred A* in the regular set, which is a truly inductive leap from a finite sample to an infinite conclusion, is seen to be justified in the ATOM results.

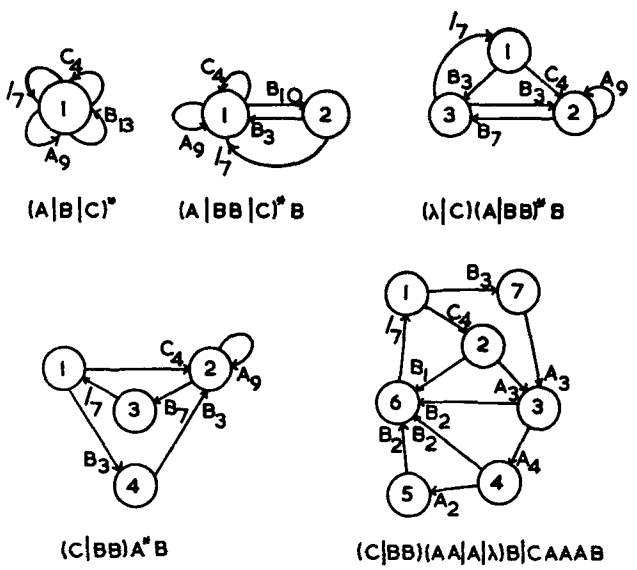
The other models are also of interest: the 1-state model is naturally the free monoid allowing any string of the atomic symbols to be in the language; the 2-state model recognizes the terminal B and the overall odd number of B's; the 3-state model eliminates

/ C A A A B / B B A A B / C A A B / B B A B / C A B / B B B / C B /

(a) Sample strings of language (separated by "/").



(b) Admissible models poorness-of-fit.



(c) 1, 2, 3, 4 and 7 state admissible models and corresponding regular events (note that λ is the null-string symbol).

FIG. 4. Grammatical inference.

the possibility of repeated C's. All of these models are viable hypotheses about the language of which the segments given are a sample and there is no deductive basis for discriminating between them (or at least none that would not arrive at the 4-state model as the "correct" result). However, the results obtained, even on this small sample, show that the approach described in this paper provides a clear-cut methodology for the inductive inference required in determining the grammar of an extensively defined language. Note again that there is no probabilistic hypothesis in this example—the methodology *does* also apply to the inference of probabilistic grammars but it is equally relevant and important in the non-probabilistic case.

4.6. EXAMPLE—SOME HUMAN BEHAVIOUR

As noted in section 1 one of the original motivations for the work described in this paper was to provide a tool for the study of human and animal behaviour—a purely behaviourist methodology with the minimum ontological commitment. The use of ATOM to analyse human and insect behaviour is now being studied. For example, Fig. 4 shows the plots of LE against states for supposedly "random" binary sequences generated by human participants in the first stage of an experimental game. The main stages of the game are concerned with guessing the next symbol(s) in a sequence of events, and, to start with, participants are asked to generate a sequence that the other players will find difficult to guess. Unguessability is a far simpler concept than "randomness", with a clear operational definition, although it clearly forces participants to generate unstructured, and hence pseudo-random, sequences.

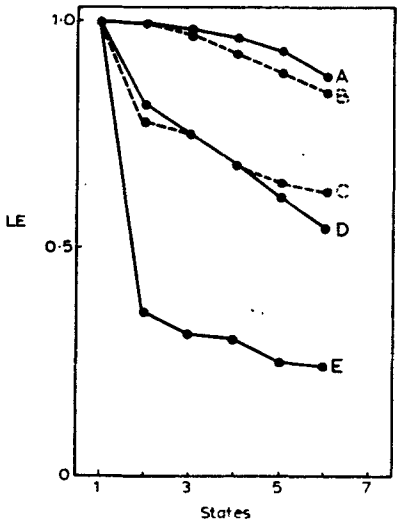


FIG. 5. Some human behaviour—admissible models poorness-of-fit for 5 subjects generating "unguessable" binary sequences.

A variety of patterns of behaviour are apparent from Fig. 4: all plots commence at LE within 0·5% of 1 corresponding to a balance between the two symbols over the whole sequence of 100 symbols—this is remarkable given that none of the subjects kept a count of their past behaviour; plot E was that of a 4-year-old boy and the corresponding 2-state model shows pronounced alternation of the two symbols; plots C and D were produced by adults who did not achieve very "unguessable" sequences—they tend to repeat a symbol too many times; plot B is that of an adult who took great care over the sequence and plot A that of a 6-year-old child who did not but achieves an unguessable result (and does so consistently in further trials).

The analysis of the guessing strategies in later games is also of interest and ATOM has picked out some interesting non-stationarities—sudden changes in behaviour resulting, presumably, from a change in hypothesis. These, and other studies will be reported elsewhere—they illustrate the data analysis potential of the behaviour-structure transformation technique, limited mainly at present by the computation time for larger models, itself possibly associated with the over-generality of general state-determined models for many aspects of human and animal behaviour.

4.7. EXAMPLE—INFERENCE OF PROGRAMS

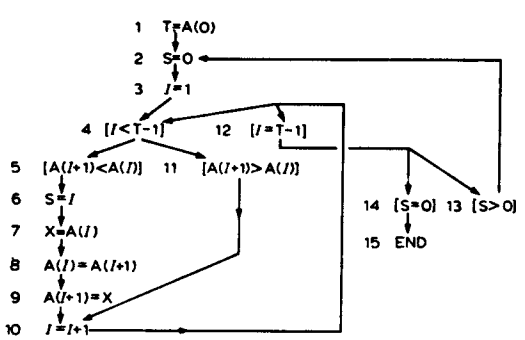
The use of grammatical inference has been suggested for the design of programming languages from natural expressions in the proposed language put forward by the designer (Crespi-Reghizzi, Melanoff & Lichten, 1973), and it may also be used to compute "analogy" relations (Gaines, 1975*b*) between program structures and languages. Biermann has experimented with the use of an automaton identification algorithm (Biermann & Feldman, 1972) to infer programs (Biermann, 1972) from traces generated by people interactively working through specific examples (Biermann, Baum, Krishnaswamy & Petry, 1973).

```
T=A(0) S=0 I=1 [I<T-1] [A(I+1)<A(I)] S=I X=A(I) A(I)=A(I+1) A(I+1)=X
I=I+1 [I<T-1] [A(I+1)>A(I)] I=I+1 [I<T-1] [A(I+1)<A(I)] S=I X=A(I)
A(I)=A(I+1) A(I+1)=X I=I+1 [I<T-1] [A(I+1)<A(I)] S=I X=A(I)
A(I)=A(I+1) A(I+1)=X I=I+1 [I=T-1] [S>0] S=0 I=1 [I<T-1]
[A(I+1)>A(I)] I=I+1 [I<T-1] [A(I+1)>A(I)] I=I+1 [I<T-1] [A(I+1)<A(I)]
S=I X=A(I) A(I)=A(I+1) A(I+1)=X I=I+1 [I<T-1] [A(I+1)>A(I)] I=I+1
[I=T-1] [S>0] S=0 I=1 [I<T-1] [A(I+1)>A(I)] I=I+1 [I<T-1]
[A(I+1)<A(I)] S=I X=A(I) A(I)=A(I+1) A(I+1)=X I=I+1 [I<T-1]
[A(I+1)>A(I)] I=I+1 [I<T-1] [A(I+1)>A(I)] I=I+1 [I=T-1] [S>0] S=0 I=1
[I<T-1] [A(I+1)>A(I)] I=I+1 [I<T-1] [A(I+1)>A(I)] I=I+1 [I<T-1]
[A(I+1)>A(I)] I=I+1 [I<T-1] [A(I+1)>A(I)] I=I+1 [I=T-1] [S=0] END
```

(a) Observed behaviour—trace of a sorting process.

```
    1: T=A(0) : S=0 -> 2(1)
    2: S=0 : I=1 -> 3(4)
    3: I=1 : [I<T-1] -> 4(4)
    4: [I<T-1] : [A(I+1)<A(I)] -> 5(5) [A(I+1)>A(I)] -> 11(11)
    5: [A(I+1)<A(I)] : S=I -> 6(5)
    6: S=I : X=A(I) -> 7(5)
    7: X=A(I) : A(I)=A(I+1) -> 8(5)
    8: A(I)=A(I+1) : A(I+1)=X -> 9(5)
    9: A(I+1)=X : I=I+1 -> 10(5)
   10: I=I+1 : [I<T-1] -> 4(12) [I=T-1] -> 12(4)
   11: [A(I+1)>A(I)] : I=I+1 -> 10(11)
   12: [I=T-1] : [S>0] -> 13(3) [S=0] -> 14(1)
   13: [S>0] : S=0 -> 2(3)
   14: [S=0] : END -> 15(1)
*  15: END :
```

(b) Admissible model output by ATOM.



(c) Admissible model—flow-chart form.

FIG. 6. Autoprogramming from traces.

Figure 6(a) shows the "trace" of a sort applied to a 1-dimensional array A in which A(0) contains the total number of elements in the array (one more than the number to be sorted) and A(1), A(2), etc., contain integers to be sorted within the array to be in order of increasing magnitude. In Biermann's system the trace would be generated by a person interacting with a computer through a graphics system that enables him to specify operations on a sample array (in this case A(0) = 6, A(1) = 3, A(2) = 1, A(3) = 5, A(4) = 4, A(5) = 2, pointers to it (in this case I and $I+1$ are pointers), and auxiliary variables (in this case T, S and X). The symbols enclosed in square brackets are conditional tests (they would appear as "note" statements to Biermann's system) and hence, from an automata-theoretic point of view, are "inputs", i.e. we wish to allow indeterminate branches to a number of tests.

The ATOM analysis of the trace of Fig. 6(a) (put in exactly as shown) is given in Fig. 8(b): it specifies a state, the associated output (Moore model), and the transitions from it (the number in brackets being the number of times they occur). This is the only admissible model produced by items if the symbols in square brackets are specified as "inputs". Figure 6(c) shows the model of Fig. 6(c) in flow-chart form as a program. It is correct and complete except for a branch from state 3 to state 12—this corresponds to an empty array being given for sorting and would have been put in if a further trace had been appended (separated by a delimiter) of a sort of an empty array. A standard requirement of Biermann's methodology is that such exceptional cases are exemplified. However, one notes that this would be a typical error in a humanly generated program! One also notes that the second branch could be inferred in the structure given by a simple closure operation that extends the branch after state 10, half of which coincides in the model with that after state 3, to completely coincide with that after state 3.

This is a simple example showing that ATOM can replicate Biermann's methodology. It may also be used to extend it in that many errors in the trace show up as branches without conditionals the removal of which reduces the poorness-of-fit only a little for a very great increase in the number of states. That these branches are unnecessary may be confirmed by running the data through without them and seeing whether the results coincide with those originally found. For example, we have analysed traces with some conditionals omitted and the results give a clear indication that a test has been omitted —the situation is closely analogous to that in which a human programmer would notice the repetition of code and decide to reduce it by use of a loop or a procedure call.

5. Summary and conclusions

This paper reports the current state of an ongoing research activity and there are many loose ends to be tied and variations on the theme to be played. The work impinges on several distinct areas of activity and the tentative conclusions are best broken down accordingly.

5.1. FORMULATION OF IDENTIFICATION FOR GENERAL SYSTEMS

It was not clear at the start of this work whether it was possible to give a definite meaning to the identification, for example, of probabilistic sources. The problem itself seemed ill-defined and approaches to its solution intuitive and heuristic. The formulation in section 3.1 in terms of a model space ordered firstly *statically by simplicity*, and secondly *dynamically by observation*, leading to the definition of an *admissible subspace* defined

naturally in terms of the joint ordering relation, seems intuitively clear, theoretically sound, and is widely applicable to all variants of the identification problem.

5.2. RELEVANCE TO INDUCTIVE INFERENCE

In particular this formulation of the general system identification problem gives a particularly "clean" analysis of the problem of inductive inference. The class of models ordered by simplicity is theoretically arbitrary but, as the results of section 2.2 show, an inappropriate choice can lead to complete failure to understand the phenomenon observed. The arbitrariness on one hand lends supports to Feyerabend's (1975) "anarchistic" viewpoint, whilst its empirical relevance to the class of phenomena actually observed in our universe leads to the refined constraints that Hesse (1974) would place upon that anarchy. Popper's (1959) "falsifiability" criterion requires models to be distinguishable; the search over the model space corresponds to Carnap's meticulous evaluation of "confirmation" (Swinburne, 1973); whilst the need to change the complete class of models under some circumstances corresponds to Kuhn's (1962) "scientific revolutions".

5.3. STRING APPROXIMATION AND PROBABILITY

The formulation in section 3.2 of string approximation measures for measuring distances between strings when the observed behaviour can be represented in this way is a general one that applies regardless of the class of models considered. It may be used to evaluate other modelling schemes, such as those in Andreae (1972–75) and in the animal behaviour literature. The "probability-forming" behaviour of these measures when the source string is probabilistically generated means that if they are used in the optimization loop a modelling scheme will behave properly (if it is possible for it to do so) when faced with a probabilistic source.

5.4. COMPUTATIONAL COMPLEXITY

The entire modelling scheme described could be regarded as a formulation of probability theory based on computational complexity as envisaged by Kolmogorov (1968). The concept of an "admissible subspace" of models seems to overcome one's intuitive objections to there being a well-defined "computational complexity" for a single sequence. Instead there is a well-defined curve of *entropy against simplicity* that defines the complexity yet still leaves the expected scope for trade-off of simpler explanation against poorer fit. Note that the results of section 3.3 leading to the Shannon entropy are based on a finite set of finite automaton models—this is a practical formulation of computational complexity as a computable function.

5. 5. GRAMMATICAL INFERENCE

A relationship between grammatical inference and computational complexity has already been established in the literature (Feldman *et al.*, 1969; Feldman, 1972). The approach to modelling described here appears to give not only a more rigorous foundation to the problem of inferring probabilistic grammars (Patel, 1972; Maryanski, 1974) but also, as shown by the results of section 4.5, a fresh viewpoint on the problem of grammatical inference itself. Through the admissible-subspace concept a trade-off may be established between grammar complexity and degree of approximation as envisaged by Wharton (1974).

5.6. AUTOPROGRAMMING FROM TRACES

The brief study of ATOM determining program structure from traces shows that Biermann's approach can be extended to situations where the trace given is ambiguous in that conditional tests are omitted or even errors are made. The significance of this is clearly related to the actual relevance of this approach to the problem of auto-programming.

5.7. PRACTICAL APPLICATIONS

The algorithms described have been implemented as a computer program that can be used to analyse arbitrary strings of behaviour up to about 1000 events long giving admissible models, poorness-of-fit curve, and variation of "surprise" with event. It is available as a component of an interactive graphics system enabling the human observer to get the "feel" of the dynamic structure of his data. Tests are currently being made of data from animal research experiments. The main limitation of the modelling process is on the size of models computable in a reasonable time—some 10 states represents a reasonable limit in interpretive BASYS on the PDP11/45. Some speed-up is envisaged through the use of a separate microprocessor for modelling but more will be attained through the use of model spaces better attuned to the problem areas, e.g. automaton models allow for indefinite memory of past events and are probably inappropriate to much animal data—a simpler class of "stimulus-response" grammars has recently been incorporated that seems to give as good models in very much less time for certain types of data.

5.8. RELATED WORK

This paper is based closely on the first detailed technical report (Gaines, 1975*b*) following the original brief description of ATOM (Gaines, 1975*a*). Another paper (Gaines, 1976*b*) analyses the concept of admissible subspace in greater depth, develops a specific model of computational complexity, and links these studies more closely to the philosophical and general system theory literature, particularly to the hierarchical model of the epistemology of system identification proposed by Klir (1975, 1976). A separate report (Gaines, 1976*d*) concentrates on the problem of stochastic grammatical inference and re-analyses Maryanski's (1974) data using ATOM.

In the next stage of development it is intended to widen the range of model sets and orderings available in ATOM, including the techniques of Andreae, Dawkins, Klir and Maryanski, in order to provide a test bed for the computer analysis of different approaches to modelling. The sharing of data bases, containing both artificial and empirical behaviour samples, amongst the various workers in this field is expected to stimulate the further development of techniques. The real breakthrough, when and if it comes, will be when the analysis of some empirical data produces an insight into the underlying structure that had not previously been available. That possibility is the driving force that will continue to stimulate activity in the field of automatic behaviour/structure transformation, that and the hope that with modern computers one may make operational some of the concepts of inductive logic that have been so long debated.

I am grateful to Peter Facey, Ladislav Kohout, Roger Moore, Steve Matheson and Ian Witten of this Department, and to Michael Arbib, Richard Dawkins, Joe Goguen, George Klir, and Judea Pearl, for stimulating discussions related to this work; and, in particular, Judea Pearl's critical comments on the report form of this paper. John Andreae, now at the University of Canterbury, New Zealand, first introduced me to these problems and the reports of his group, and personal correspondence, have been a continual source of stimulation and ideas.

References

ACZEL, J. & PFANZAGL, J. (1966). Remarks on the measurement of subjective probability and information. *Metrika*, **5**, 91–105.

ANDREAE, J. H. & CASHIN, P. M. (1969). A learning machine with monologue. *International Journal of Man–Machine Studies*, **1**, 1–20.

ANDREAE, J. H. (1972–75). *Man–Machine Studies Reports UC-DSE/1 through 7.* University of Canterbury, Christchurch, New Zealand.

ANDREAE, J. H. & CLEARY, J. G. (1976). A new mechanism for the brain. *International Journal of Man–Machine Studies*, **8**, 89–119.

ARBIB, M. A. & ZEIGER, H. P. (1969). On the relevance of abstract algebra to control theory. *Automatica*, **5**, 589–606.

ARBIB, M. A. & MANES, E. G. (1974). Foundations of system theory: decomposable systems. *Automatica*, **10**, 285–302.

BIERMANN, A. W. (1972). On the inference of Turing machines from sample computations. *Artificial Intelligence*, **3**, 181–198.

BIERMANN, A. W., BAUM, R., KRISHNASWAMY, R. & PETRY, F. E. (1973). *Automatic Program Synthesis Reports, OSU-CISRC-TR-73–6.* Computer and Information Sciences Research Center, Ohio State University, U.S.A.

BIERMANN, A. W. & FELDMAN, J. A. (1972). On the synthesis of finite-state machines from samples of their behaviour. *IEEE Transactions on Computers*, **C-21**, 592–597.

CHAITIN, G. (1969). On the length of programs for computing finite binary sequences. *Journal of the Association for Computing Machinery*, **16**, 145–159.

CHAITIN, G. (1975). A theory of program size formally identical to information theory. *Journal of the Association for Computing Machinery*, **22**, 329–340.

CRESPI-REGHIZZI, MELKANOFF, M. A. & LICHTEN, L. (1973). The use of grammatical inference for designing programming languages. *Communications of the Association for Computing Machinery*, **16**, 83–90.

DAWKINS, R. & DAWKINS, M. (1973). Decisions and the uncertainty of uncertainty. *Behaviour*, **45**, 83–103.

DAWKINS, M. & DAWKINS, R. (1974). Some descriptive and explanatory models of decision-making. In MCFARLAND, D. J. (ed.) *Motivational Control Systems Analysis.* London and New York: Academic Press, pp. 119–168.

EVANS, T. G. (1971). Grammatical inference techniques in pattern analysis. In TOU, J. T. (ed.) *Software Engineering, Vol. 2.* New York: Academic Press, pp. 183–202.

EYKHOFF, P. (1974). *System Identification.* Chichester: John Wiley.

FELDBAUM, A. A. (1963). Dual control theory problems. In *Proc. 2nd IFAC Congress*, Basle.

FELDMAN, J. A. (1972). Some decidability results on grammatical inference and complexity. *Information and Control*, **20**, 244–262.

FELDMAN, J. A., GIPS, J., HORNING, J. & REDER, S. (1969). Grammatical complexity and inference. *Artificial Intelligence Memo No. 89.* Computer Science Dept., Stanford University, Stanford, Calif., U.S.A.

FEYERABEND, P. (1975). *Against Method.* London: NLB.

FINETTI, B. DE (1972). *Probability, Induction and Statistics.* Chicester: John Wiley.

FITCH, W. M. & MARGOLIASH, E. (1967). Construction of phylogenetic trees. *Science*, **155**, 279–284.

FREUD, S. (1914). *Psychopathology of Everyday Life.* London: Ernest Benn.

FU, K. S. & BOOTH, T. L. (1975*a*). Grammatical inference: introduction and survey—Part I. *IEEE Transactions on Systems, Man and Cybernetics*, **SMC-5**, 95–111.

FU, K. S. & BOOTH, T. L. (1975*b*). Grammatical inference: introduction and survey—Part II. *IEEE Transactions on Systems, Man and Cybernetics*, **SMC-5**, 409–423.,

GAINES, B. R. (1971*a*). Axioms for adaptive behaviour. *International Journal of Man–Machine Studies*, **4**, 169–199.

GAINES, B. R. (1971*b*). Memory minimization in control with stochastic automata. *Electronics Letters*, **7**, 710–711.

GAINES, B. R. (1974). Training, stability and control. *Instructional Science*, **3**, 151–176.

GAINES, B. R. (1975*a*). Approximate identification of automata. *Electronics Letters*, **11**, 444–445.

GAINES, B. R. (1975*b*). Analogy categories, virtual machines and structured programming. In *Proc. 5th International Congress of Gesellschaft für Informatik*, Dortmund, Germany, October.

GAINES, B. R. (1975*c*). Behaviour/structure transformations under uncertainty. *EES-MMS-AUT-75*. Department of Electrical Engineering Science, University of Essex, U.K.

GAINES, B. R. (1976*a*). On the complexity of causal models. *IEEE Transactions on Systems, Man and Cybernetics*, **SMC-6**, 56–59.

GAINES, B. R. (1976*b*). System identification, approximation and complexity. *International Journal of General Systems*, **3** (to appear).

GAINES, B. R. (1976*c*). The role of randomness in system theory. *EES-MMS-RAN-76*. Department of Electrical Engineering Science, University of Essex, U.K.

GAINES, B. R. (1976*d*). Inference of stochastic grammars—a formulation and solution. *EES-MMS-AUT-76*. Department of Electrical Engineering Science, University of Essex, U.K.

GAINES, B. R. & ANDREAE, J. N. (1966). A learning machine in the context of the general control problem. In *Proc. 3rd IFAC Congress*, London, Institute of Mechanical Engineers.

GAINES, B. R. & FACEY, P. V. (1975). Some experience in interactive system development and application. *Proceedings of the IEEE*, **63**, 894–911.

GOGUEN, J. A. (1973). Realization is universal. *Mathematical Systems Theory*, **6**, 359–374.

GOGUEN, J. A. (1975). Discrete-time machines in closed monoidal categories. *International Journal of Computer and System Sciences*, **10**, 1–43.

GOLD, E. M. (1967). Language identification is the limit. *Information and Control*, **10**, 447–474.

GOLD, E. M. (1971). Universal goal seekers. *Information and Control*, **18**, 395–403.

HESSE, M. (1974). *The Structure of Scientific Inference*. London: Macmillan.

HOPCROFT, J. (1971). An $n \log n$ algorithm for minimizing states in a finite automaton. In KOHAVI, Z. & PAZ, A. (eds) *Theory of Machines and Computations*. New York: Academic Press, pp. 189–196.

HULL, C. L. (1943). *Principles of Behaviour*. New York: Appleton-Century-Crofts.

KLIR, G. (1975). On the representation of activity arrays. *International Journal of General Systems*, **2**, 149–168.

KLIR, G. (1976). Identification of generative structures in empirical data. *International Journal of General Systems*, **3** (to appear).

KOLMOGOROV, A. N. (1968). Logical basis for information theory and probability theory. *IEEE Transactions on Information Theory*, **IT-14**, 662–664.

KUHN, T. S. (1962). *The Structure of Scientific Revolutions*. Chicago: University of Chicago Press.

KWAKERNAAK, H. (1965). Admissible adaptive control. In *Proc. IFAC Symposium on "The Theory of Self-Adaptive Control Systems"*, Society of Instrument Technology, London.

MARYANSKI, F. J. (1974). Inference of probabilistic grammars. *Ph.D. Thesis*. University of Connecticut, Storrs.

MARTIN-LÖF, P. (1966). The definition of random sequences. *Information and Control*, **9**, 602–619.

MICHOTTE, A. (1963). *The Perception of Causality*. London: Methuen.

NAGEL, E. (1961). *The Structure of Science*. London: Routledge and Kegan Paul.

NERODE, A. (1958). Linear automaton transformations. *Proceedings of the American Mathematical Society*, **9**, 541–544.

PATEL, A. R. (1972). Grammatical inference for probabilistic finite-state languages. *Ph.D. Thesis*. University of Connecticut, U.S.A.

PEARL, J. (1975*a*). On the complexity of inexact computations. *UCLA-ENG-0775*. School of Engineering and Applied Science, UCLA, Calif., U.S.A.

PEARL, J. (1975*b*). On the complexity of imprecise causal models. *UCLA-ENG-7560*. School of Engineering and Applied Science, UCLA, Calif., U.S.A.

PEARL, J. (1975*c*). An economic basis for certain methods of evaluating probabilistic forecasts. *UCLA-ENG-7561*. School of Engineering and Applied Science, UCLA, Calif., U.S.A.

PEARL, J. (1975*d*). On the complexity of computing probabilistic assertions. *UCLA-ENG-7562*. School of Engineering and Applied Science, UCLA, Calif., U.S.A.

POPPER, K. R. (1959). *The Logic of Scientific Discovery*. London: Hutchinson.

POPPER, K. R. (1972). *Objective Knowledge*. Oxford: Clarendon Press.

SANKOFF, D. (1972). Matching sequences under deletion insertion constraints. *Proceedings of the National Academy of Sciences, U.S.A.*, **69**, 4–6.

SAVAGE, L. J. (1970). Elicitation of personal probabilities and expectations. *Journal of the American Statistical Association*, **66**, 783–801.

SCHILPP, P. (ed.) (1949). *Albert Einstein, Philosopher-Scientist*. Evanston, Ill., U.S.A.:

SELLERS, P. H. (1974). An algorithm for the distance between two finite sequences. *Journal of Combinatorial Theory (A)*, **16**, 253–258.

SHACKLE, G. L. S. (1955). *Uncertainty in Economics*. Cambridge University Press.

SHACKLE, G. L. S. (1961). *Decision, Order and Time*. Cambridge University Press.

SHANNON, C. E. (1948). The mathematical theory of communication. *Bell Systems Technical Journal*, **27**, 379.

SHUFORD, E. H., ALBERT, A. & MASSENGILL, H. E. (1966). Admissible probability measurement procedures. *Psychometrika*, **31**, 125–145.

SHUFORD, E. H. & BROWN, T. A. (1975). Elicitation of personal probabilities and their assessment. *Instructional Science*, **4**, 137–188.

SOLOMONOFF, R. J. (1964*a*). A formal theory of inductive inference. Part I. *Information and Control*, **7**, 1–22.

SOLOMONOFF, R. J. (1964*b*), A formal theory of inductive inference. Part II. *Information and Control*, **7**, 224–254.

SUPPES, P. (1974). *Probabilistic Metaphysics*. Filosofiska Studier, Uppsala Universitet, Sweden.

SWINBURNE, R. (1973). *An Introduction to Confirmation Theory*. London: Methuen.

TOLMAN, E. C. (1951). A new formula for behaviorism. *Collected Papers in Psychology*. University of California Press, 1–8.

WAGNER, R. A. & FISCHER, M. J. (1974). The string-to-string correction procedure. *Journal of the Association for Computing Machinery*, **21**, 168–173.

WATANAKE, S. (1969). *Knowing and Guessing*. New York: John Wiley.

WEISS, L. (1961). *Statistical Decision Theory*. New York: McGraw-Hill.

WHARTON, R. M. (1974). Approximate language identification. *Information and Control*, **26**, 236–255.

WILLIS, D. (1970). Computational complexity and probability constructions. *Journal of the Association for Computing Machinery*, **17**, 241–259.

WINKLER, R. L. (1974). Probabilistic prediction: some experimental results. *Journal of the American Statistical Association*, **66**, 625–688.

WINKLER, R. L. & MURPHY, A. H. (1968). "Good" probability assessors. *Journal of Applied Meteorology*, **7**, 751–758.

WITTEN, I. H. (1976). The apparent conflict between estimation and control. *Journal of the Franklin Institute*, **301**, 161–189.

ZADEH, L. A. (1962). From circuit theory to system theory. *Proceedings of the IRE*, **50**, 856–865.