Automatica. Vol. 13, pp. 197- 202. Pergamon Press, 1977. Printed in Great Britain

Minor re-edits for clarity 3/19/08 EHB

Brief Paper

Pattern Recognition: an Alternative to Parameter Identification in Adaptive Control*

EDGAR H. BRISTOL†

Key Word Index—Identification; pattern recognition; adaptive control; computer control.

Summary—Practical models used in identification of process control processes must be too simplistic to give precise control information. However, these models can be used for adaptation if they are continuously readapted. But the identification then lacks the precision which might justify the analytic elaboration. One alternative has been to use pattern recognition as a means for allowing a computer system to characterize transient response computing readapted parameters which cause the control behavior to approach a desired transient 'shape'. The paper summarizes work using pattern features as a basis for practice and theory.

Introduction

IN PROCESS control, the processes are analytically complex. They are nonlinear, describable by partial differential equations, and designed in irregular and therefore nonanalytical form. Accurate modeling of these processes is impractical with present techniques. But the equations are typically diffusion equations, and the actual response data shows very smooth, usually monotonic, dynamic, and nonlinear response characteristics. However, practical control succeeds because these processes behave in quite predictable fashion. The theory of positive operators [l, 2] gives a little-known, but quite powerful, theoretical explanation of this kind of process.

None of this means that existing dynamic modeling techniques are not used successfully today. But under the best of circumstances only qualitative matching between process and model will occur, with the mismatch being quite visible in any dynamic data. Depending on the control application, the effect of this error or mismatch can he harmless or severe. And in feedback applications the lack of any formal handling of the potential ill effects makes practical application in adaptive systems *without close manual supervision* ill-advised.

This paper elaborates on previous papers by the author on a complementary approach to practical adaptation and control based on pattern recognition. Although these techniques are experimentally developed and not easily formalized in a useful way, they are based on heuristic observations which would be obvious to the experienced observer, and share several underlying characteristics:

1. They may be intuitively understandable by the experienced, but unsophisticated, user not only under normal operating circumstances but under crisis, facilitating manual supervision.
2. This intuition may extend well into process complexities previously discussed and natural disturbance environments.

The paper references early, technically successful work and presents several admittedly incomplete but suggestive experimental and analytical results to indicate the kind of directions possible from the pattern recognition base. It is felt that the pattern concept, as presented, will allow a more flexible base for operational and computer-oriented process control adaptive systems, and a more fundamental set-theoretic view of the information relationships inherent in adaptive control.

The author's first experimental adaptation work [3] was based on identification because of its intellectual appeal. But it was clear that since the process model must be of low order, any adaptive system must be made insensitive to the model approximations. Accordingly, the initial testing emphasized tests where the process structure was clearly different from that of the model. The control design was based on a very fast continuous on-line identification under the control of an on-line sensitivity calculation. For extreme conditions of process model mismatch, the identification and control parameters could vary markedly, but frequently under conditions that did not affect the control quality in any noticeable way.

A more recent analog computer experiment [3] demonstrates both the problem and the resolution of the problem. Figure 1 shows an experimental setup for comparing two different simulation models, each with three parametric degrees of freedom (k, τ_1, τ_2), in loops with identical PI controllers. An output integral-square criterion function circuit for objective evaluation of tuning and model fitting operations was included. All tuning and model fitting operations were done manually, minimizing integral-square error.

The experiment allowed a simple philosophical representation of the identification and adaptive process. It abstracted the real situation where a real, high-order process must be modeled by a much simpler, structurally different, identification model. The switches in the circuit permit three basic operations in this abstract representation:

1. Fitting of the open-loop responses of the two simulation models to each other, which corresponds to open-loop modeling of the process.
2. Fitting of the closed-loop responses of the two models to each other for given control settings, which corresponds to closed-loop modeling of the process.
3. Tuning either controller to optimum settings. When the 'model' loop is tuned and the settings are applied to the 'process' loop, this corresponds to an adaptation.

Figure 2 shows the results of an initial attempt at open-loop identification followed by a closed-loop adaptation based on tuning the 'model' loop. The open-loop responses show a maximum identification error of only 3%. The control is improved if the process is reidentified under the closed-loop conditions [3] and returned. Further repetition gives a convergence to good control settings.

A simpler but less realistic analytical example is the process

$$\frac{1}{S+1} - \frac{\varphi}{S^2+1}$$

This process is highly stable under negative feedback for ($\varphi = 0$ but unstable if ($\varphi > 0$ however small the φ or feedback gain, as is easily seen from the root locus.

Nonlinear structural differences can give more bizarre effects. A step response to an nth order multilag system tends to have a shape like an nth order parabola at the origin, as in Fig. 3. A first-order system followed by a nonlinear function with the same nth order parabola at the origin can have identical behavior. Thus nonlinearity can affect the apparent dynamic complexity seen by an identification system. There is a considerable literature on modeling inaccuracy [4–6], but the author is not familiar with any studies of the practical effects of structural mismatch.

* Received 10 July 1975; revised 12 December 1975: revised 18 August 1976. The original version of this paper was presented at the Sixth IFAC Congress on Control Technology in the Service of Man which was held in Boston/Cambridge, MA U.S.A. during August 1975. The published Proceedings of this IFAC Meeting may be ordered from: Instrument Society of America. 400 Stanwix Street, Pittsburg, PA 15222, U.S.A. or John Wiley. Bathos Lane, Chichester, Sussex, United Kingdom. This paper was recommended for publication in revised form by associate editor I. Landau.

† The Foxboro Company. Foxboro, MA 02035, U.S.A.

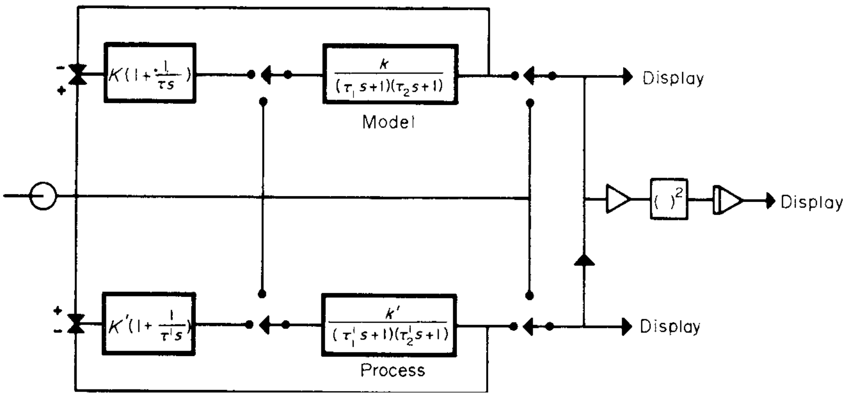

Fig.1.Model-process two loop comparison simuation

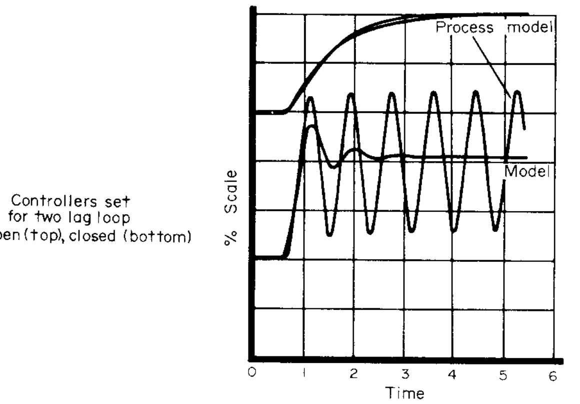

Fig. 2. Open-loop and closed-loop responses; best open-loop fit.

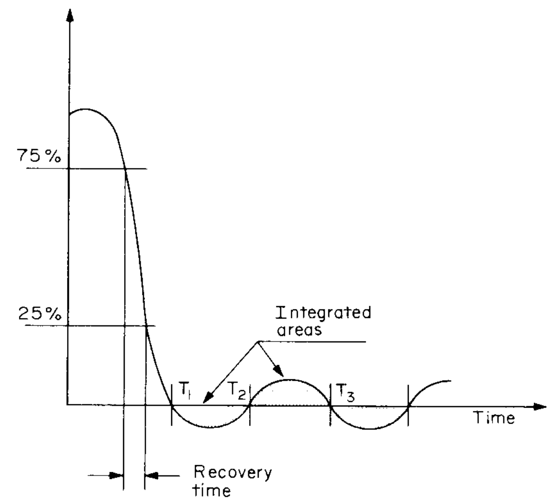

Fig. 4. Pattern recognition adaptive scheme.

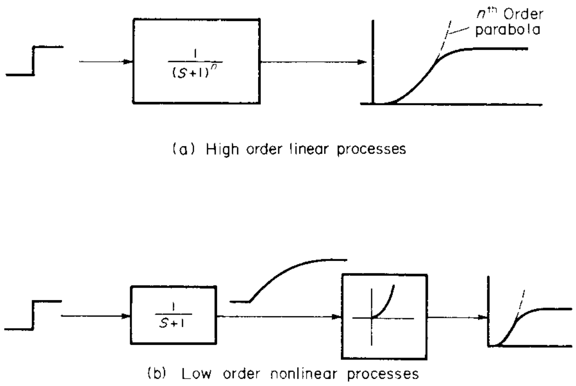

Fig. 4. High order linear and low order nonlinear prcesses with identical.

If identification with practically small models gives inaccurate results and requires a continuous feedback process, a far more simple approach should give nearly as good results. A pattern recognition adaptive scheme was proposed and thoroughly tested [7]. The system used the significant pattern features of a transient recovery as shown in Fig. 4, the recovery time and normalized integrated signal levels in the response, to determine the shape of the response and to adjust controller settings. The effectiveness, high insensitivity to noise, and relative intuitive and operational simplicity are demonstrated in several papers and a videotape [3, 7-9].

A pattern recognition control scheme

If pattern recognition simplifies adaptation, it should also simplify control directly. But any approach to control must address dynamic compensation. In linear feedback systems. dynamic compensation has three functions:

1. Stabilize the loop.
2. Speed the process response.
3. Anticipate effects of disturbances.

In process control, Function 1 exists not because of the process but because of the use of feedback control, and Function 2 is generally not of major economic value.

For many applications, the causes of control error can be broken into three categories:

1. Low level, high frequency, noise-like variations of the variable which cannot be controlled and are best treated as noise, to be filtered and ignored.
2. Low frequency drifts which can be effectively removed by the simplest feedback control.
3. Transient disturbances caused by discrete process events or step changes in the control reference values.

Accordingly, a pattern recognition control system should emphasize pattern features which anticipate the effects of transient disturbances and operation changes. An experimental adaptive control system was programmed using only these three pattern features as the basis for control decisions:

1. The initial inflection point in a disturbance.
2. The magnitude of any steady state error condition.
3. The magnitude of any change in control reference.

Throughout the paper, any reference to inflection points will refer to the point of maximum magnitude of signal derivative in the initial error response to a disturbance. A suitably noise-compensated estimate of the derivative at this point defines the feature. The disturbance was assumed to be of a fixed but unspecified transient form with unknown and variable magnitude. Because disturbance direction is easily distinguished, the positive-going disturbances were adapted separately from the negative-going ones.

In effect. the model of the process for each disturbance was

$$\Delta y_{ss} = a \cdot D_{max} + b \cdot \Delta u_{ss} \quad (1)$$

with y_{ss} being the steady state value of the controlled variable, D_{max} being the magnitude of the derivative at the inflection point in the initial disturbance response, and u_{ss}, being the steady state value of the manipulated variable. Since u is changed infrequently, only steady state behavior is considered. Since the process is treated incrementally, only the constants a and b need be identified. The model assumes that D_{max} does not depend on the change in control variable, but only on the disturbance magnitude. This is a reasonable assumption because the control effects are mostly seen in the recovery portion of the disturbance response.

Control consisted of monitoring changes in reference R, new disturbances, and residual errors after control, using the model to compute the u_{ss} which would reduce the steady state error to zero, as in Fig. 5. The control law which causes zero steady state error is

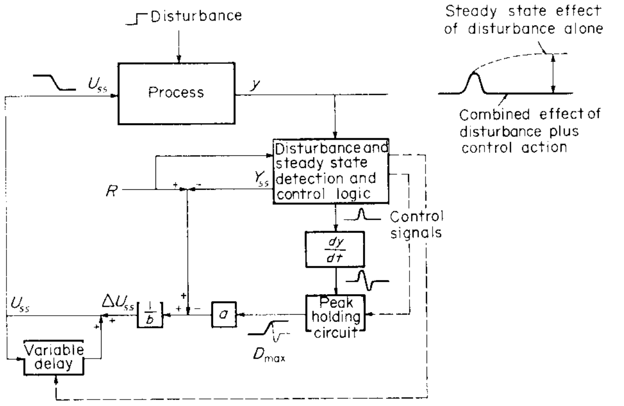

Fig. 5. Block Diagram of a control scheme

obtained by solving equation (2) for Δu_{ss} with Δy_{ss}= R-y_{ss}, R is a reference value, the desired value of y_{ss}.

$$\Delta u_{ss} = \frac{R - y_{ss} - aD_{max}}{b} = -\frac{a}{b}D_{max} + \frac{R - y_{ss}}{b} \quad (2)$$

Control action was taken:

1. Whenever a new disturbance was recognized,
2. Whenever a change in R was entered, or
3. Whenever an offset remained for a sufficient length of time to be classified as a residual steady state error, estimated by monitoring the derivative or taken as a worst-case system parameter. The value is not sensitive.

Figure 6 shows an analog simulated comparison of the kind of control of load disturbances attainable once adaptation is complete. Figure 6a shows the response of a twentieth-order Padé approximation to an equal dead time and lag process with an optimally-tuned PID controller. Figures 6b–d show the same disturbance controlled by the pattern recognition system. Figure 6b shows the controlled variable. 6c shows the derivative of the controlled variable, and 6d shows the result of tracking the derivative response to its peak value, the inflection point, and holding that value as the manipulated variable. The pattern-based control is inherently stable, simple, and of equal control quality to the PID response. Since the control depends on the parameters a and b, it is independent of the details of the dynamic structure of the process, so long as the process is naturally monotonic.

Adaptation consists of using any residual error after the process has settled out as a new value of y_{ss} which, with the corresponding value of D_{max} can be used to reidentify the model parameters a and b in equation (1). Figure 7 shows adaptation under transient process disturbances. All control calculations were digital.

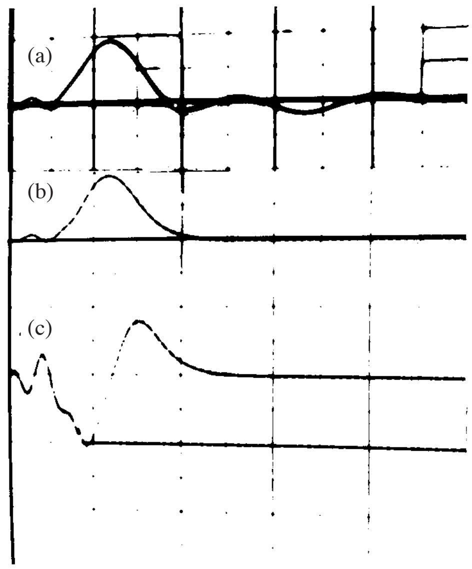

Fig. 6(a, b, c). Pattern recognition control scheme.

Further experiments made more elaborate use of the inflection information to respond quickly to control overshoots. Adaptation was fast because it always involved solving for only one or two parameters from one or two consecutive disturbance records. Each time a new disturbance shape was introduced, the system adjusted after a single trial as in Fig. 7. A more elaborate system might use disturbance peak and valley information to distinguish between different disturbance types and thus avoid continual readaptation to different-shaped disturbances.

A theory of pattern-based control

This section illustrates the formal use of pattern feature variables, the variables which define the pattern features, as a substitute for state variables. The resulting equations exactly model the process with fewer and more directly obtainable variables than with a conventional-state variable model in a process control environment.

Consider a dead time and lag process onto whose delayed input is added a varying disturbance $d(t)$. A state representation of the process might be

$$\frac{dy}{dt} = \frac{1}{\tau}(u(t-T) + d(t) - y) \quad (3)$$

The state information is contained in the value $y(t)$ and the function $u(s)$ for $t\text{-}T \le s < t$ The current time is t and $u(t)$ is the current manipulated variable. If the differential equation is replaced by a difference equation and the u is changed only at the sampling time ($n\Delta t$), then a reduction of state space occurs

$$y_{n+1} = y_n + (1 - e^{-((\Delta t)/\tau)})(u_{n-(T/(\Delta t))} + dt - y_n) \quad (4)$$

The state information is contained in the variables y_{n+1} and

$$u_n, u_{n-1}, \ldots, u_{n-(T/(\Delta t))+1}\,.$$

This model is accurate but there is a trade-off between immediate response to disturbances (small Δt) and the number of state variables required.

Suppose that the system is reexamined in terms of pattern feature variables corresponding to peaks, valleys, inflection points, and steady state sections of the response as in Fig. 8. Suppose that the feature variables for the jth feature are: y_j. which is the magnitude of y; y_j' which is the inflection point derivative of y; and the time of occurrence (T_j) at which the jth peak or other feature occurs. Note, that for the dead time and lag process, every peak is associated with an inflection point right after it, as in Fig. 8. y_j' for the peak will be

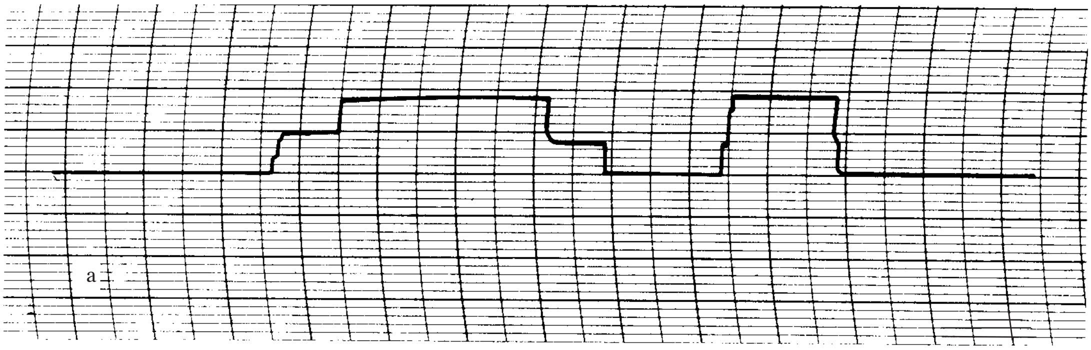

Fig. 7a. The control action (u).

Fig. 7b. The controller response under control (y).

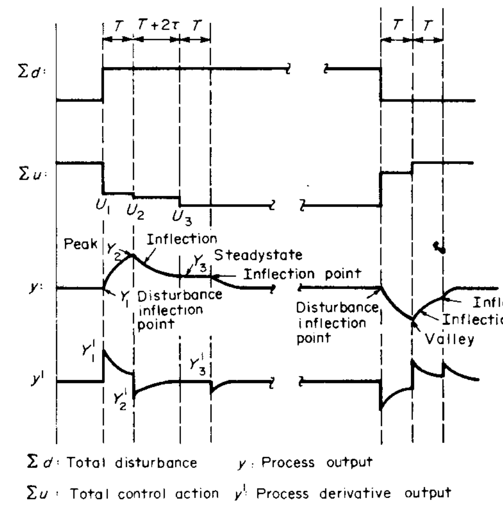

Fig. 8. Pattern analysis of dead time and lag process.

taken from this inflection point. Now suppose that the disturbance consists of step changes (magnitude d_j) on the disturbance input in equation (3), and of a frequency such that the likelihood of more than one disturbance occurring within two dead times of each other is small enough to be ignored. Let d_j occur at time $T(d_j)$.

The control actions (u_j) are steps on the input taken at the times T_j on the basis of the feature variable information. The calculation is simplified if control action is taken only on the first peak or inflection point, the one caused by the disturbance, within a time interval of length T after a disturbance step. Any intervening feature is ignored. That is, if $T_j = T(d_j)$, $T_{j+1} \geq T_j + T$. This also implies that, if $T_j - T_{j-1} < T$, then $T_{j+1} - T_j \geq T$.

For this particular example, the peaks or inflection points caused by control actions occur at time T after the control action. Let steady state be estimated as occurring at a time $T + 2\tau$ after the last peak or inflection point for which $y' > y'_{min}$ for some predetermined y'_{min}. Then the fixed sample-time condition of equation (4) is replaced by a recursive calculation of T_j: (5a)

$T_j = T_{j-1} + T$, if $|y'(Y_{j-1}+T)| > y'_{min}$ and no disturbance change has occurred in the interval $T_{j-1} < t < T_{j-1} + T$.

(5b)

$T_j = T(d_j)$, if the previous condition is not met and a disturbance step occurs at the time $T(d_j)$ with $T_{j-1} < T(d_j) < T_{j-1} + T + 2\tau$.

(5c)

$T_j = T_{j-1} + T + 2\tau$., otherwise (under steady state).

For control calculations as well as to develop the recursive equations for y_j and y'_j, it will be useful to consider the steady state effect (y_j^{ss}) of all process inputs with effects observable at time T_j that is, all control actions occurring at or before $T_j - T$, as well as all disturbances ($d_j = \emptyset$ for nondisturbance values of j).

$$y_j^{ss} = \sum_i^k u_i + \sum_i^j d_i \qquad (6)$$

where

$$\begin{aligned} k &= j-1 \quad \text{if } T_j - T_{j-1} \geq T \\ &= j-2 \quad \text{if } T_j - T_{j-1} < T \end{aligned}$$

Also, intermediate variables v_j and w_j, which take the effect of dead time into account, simplify calculation

$$\begin{aligned} v_j &= u_j + u_{j-1} \quad \text{if } T_j - T_{j-1} < T \\ &= u_j \quad \text{if } T_j - T_{j-1} \geq T \end{aligned} \qquad (7a)$$

$$w_j = [u_j + u_{j-1}e^{-(T_j - T_{j-1})/\tau}]/\tau \quad \text{if } T_j - T_{j-1} < T$$
$$= u_j/\tau \quad \text{if } T_j - T_{j-1} \geq T. \tag{7b}$$

Recall that $T_j - T_{j-1} < T$ implies that $T_{j+1} - T_j \geq T$.

If $T_j - T_{j-1} > T$, v_j and w_j, represent the most recently observed contribution of u to y^{ss}_{j+1} and y'_j, respectively, as will be shown.

Then from equation (3), where y^{ss}_j now replaces $(u(t\text{-}T) + d(t))$

$$y_j = y_j^{ss} + \tau y_j' \tag{8}$$

and from equations (6) and (7)

$$y^{ss}_j = y^{ss}_{j-1} + d_j \quad \text{if } T_j - T_{j-1} < T$$
$$= y^{ss}_{j-1} + v_{j-1} + d_j \quad \text{if } T_j - T_{j-1} \geq T$$

or from equation (8)

$$y_j = y^{ss}_{j-1} + d_j - \tau y'_j \tag{9a}$$
$$= (y_{j-1} + \tau y'_{j-1}) + d_j - \tau y'_j \quad \text{if } T_j - T_{j-1} < T$$

or

$$= (y^{ss}_{j-1} + v_{j-1} + d_j) + \tau y'_j \tag{9b}$$
$$= (y_{j-1} + \tau y'_{j-1} + v_{j-1} + d_j) + \tau y'_j \quad \text{if } T_j - T_{j-1} \geq T$$

Similarly

$$y'_j = y'_{j-1}e^{-(T_j - T_{j-1})/\tau} + \frac{d_j}{\tau} \quad \text{if } T_j - T_{j-1} < T \tag{10}$$

or

$$= \left((y'_{j-1} + w_{j-1}e^{T/\tau})e^{-(T_j - T_{j-1})/\tau} + \frac{d_j}{\tau}\right) \quad \text{if } T_j - T_{j-1} \geq T$$

The equations (5), (9) and (10) constitute a recursion similar to equations (3) or (4), capable of prediction of the effects of control. The state information is carried in the variables T_j, y_j, y'_j, u_{j-1}, and u_{j-2}, with u_j and d_j being independent. From a control point of view, there is the computational advantage of a long sample time and few state or feature variables. The rapid response to disturbances obtainable from a much higher-order model operated at a frequent sampling interval still exists. These equations can be optimized for any appropriate control criterion. In particular, the criterion of zero steady state error amounts to solving the above equations, together with $y^{ss}_{j+1} = R$, for u_j. Note that, from equation (8a), $(v_j - u_j)$ does not depend on u_j.

$$u_j = R - y_j^{ss} - d_j - (v_j - u_j)$$

From an adaptive point of view, all of the pattern feature variables can be measured. Accordingly, the model can be identified by conventional means.

A more elaborate use of pattern features, or a more complex multivariable nonlinear process or disturbance structure, would complicate the feature variable equation and make it less analytical. But it would not alter the basic conclusion that a process model can be based on any fairly broad set of pattern features of the process input and output functions just as well as on conventional state variables. The more nonanalytical character of the resulting equations is not, in itself, a basic problem because the model can be adapted and can have some suitable approximate form.

More importantly, the practical nonlinearities of the process (hysteresis and stick slip) lead to a range of model structure which cannot practically be anticipated with fixed structure models. Current work emphasizes accurate structure-free models suitable to this kind of adaptive control environment. Figure 9 shows the results of some early experiments toward this kind of structure-free, interpolation-based modeling. The data points were fed at random to a computer which attempted to incorporate them into the model. The vertices in the figure represent data points of a function of two independent variables and one dependent one. The lines outline the surface of the function in three-space as it is reconstructed by interpolation. Work is continuing toward an effective system organization for this kind of modeling. The result is a potential bridge of the gap between the analytic complexity and practical simplicity of real industrial processes. The work is related to the off-line work of Barron and Ivakhnenko[11, 12].

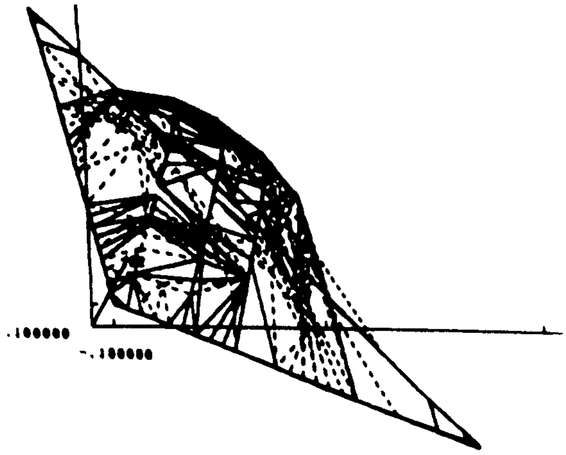

Fig. Structure-free modeling.

An abstract result about learning

The flexibility of the pattern orientation to deal with high-order nonlinear processes is likely to require a more computer-science-oriented flavor than seen heretofore in control. The theoretical counterpart of this would be a more information- or set-theoretic emphasis. The following section illustrates the possibilities.

Definition of learning systems has been a control hobby with many adherents[8, 10]. A rough definition due to Narendra et al.[10] can be modeled with pattern ideas: in Fig. 10, let i be a vector of pattern features which characterize the system input partitioned into subvectors i_0, i_1, and i_2, corresponding to an initial interval of time in which the input is in `steady state', the input contains no information), a middle interval with transient, information-containing, input behavior, and a final interval again with information-free input behavior, respectively. Let s be a vector of pattern features representing the internal state or output behavior, similarly partitioned into s_0, s_1, and s_2. Steady state and transient behavior correspond respectively to the two sets of feature vectors, S and T, or more generally, S_i, S_s, T_i, and T_s. Let the superscripts S and T represent membership in either of these sets: s_0^S, s_1^T, or s_2^S. Thus, s_0^S is shorthand for the expression: s_0^S is some s_0^S, such that $s_0 \varepsilon S$. Let numerical or indexed superscripts further distinguish membership in subsets, so that $s_0^{S_1}$ and s$s_0^{S_2}$ might represent the steady state encoding of two different learned final states. Let F be the function between the i and s: $s = F(i)$; and let G, H, etc. be other functions. This model expresses a situation where an initially stabilized system undergoes a transient 'experience' (i_1^T) and 'learns' something

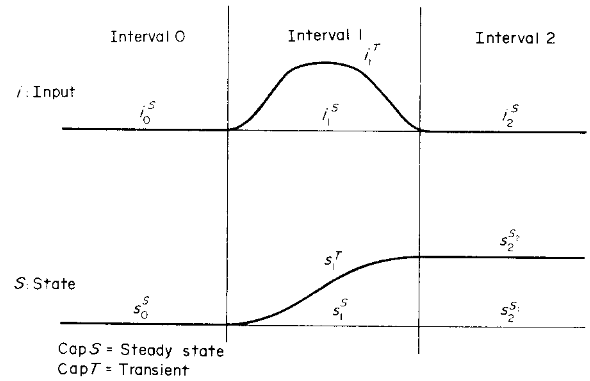

Fig. 10. A model of learning.

(s_2^S). The model and the following definitions and theorem are illustrated in Fig. 10.

Definition 1. A system F as modeled above with nonempty sets S and T is a relative learning system, or simply a learning system, if:

(a) there is a function G such that for any i of form (i_0^S, i_1, i_2^S) and the corresponding s of form (s_0^S, s_1, s_2^S) where (s_0^S, s_1, s_2^S) = $F(i_0^S, i_1, i_2^S)$: $s_2^S = G(s_0^S, i_1)$.

In other words, what the system F learns (s_2^S) *depends only on its past experience* (s_0^S) *and the new information* (i_1).

(b) there is a nonconstant function H such that $s_2^S = H(s_0^S)$ $= G(s_0^S, i_1^S)$ for all i_1^S ($i_1 \varepsilon S$).

In other words, there can be no change in learned state caused by any noninformation-containing input signal. But H does permit 'forgetting' of information.

(c) there is no function K such that $s_2^S = K(s_0^S)$ (that is, $K(s_0^S)$= $G(s_0^S, i_1^T)$ for all s_0^S, i_1^T.

In other words, the learned state must be dependent on the input i_1^T *over at least some part of its range.*

An analog integrator is a learning system if S for the input variable is the set of time segments whose values are constantly zero, and S for the output variable is the set of time segments of signals whose value is constant. Figure 10 illustrates the situation if i_0^S, i_1^S, and i_2^S are the input signal segments which cause the sequence s_0^S, s_1^S, and $s_2^{S_1}$ and (i_0^S, i_1^T, and i_2^S) cause s_0^S, s_1^T, and $s_2^{S_2}$.

Definition 2. A learning system is an absolute learning system if there is a function L such that

$$s_2^S = L(i_1^T) \quad \text{for all } s_0 \,\varepsilon\, S \text{ and all } i_1^T.$$

An absolute learning system must be able to converge to a behavior consistent only with its input information.

A more sophisticated variation of Definition 2 might deal with conventional partial convergence in a more direct way, but it would not change the following theorem. Core memory in a computer is an absolute learning device. Both kinds of learning are useful to adaptive systems. Relative learning will almost always be tied to some kind of feedback mechanism which provides the convergence or even the nonlinearity needed to give the system an absolute learning capability.

Theorem. An absolute learning system must be nonlinear.

Proof. Suppose that the system was linear. Then for any distinct s_0^S, $s_2^{S_1}$, $s_2^{S_2}$, i_1^S, and i_1^T, as in Fig. 10, such that

$$s_2^{S_1} = G(s_0^S, i_1^S)$$

and

$$s_2^{S_2} = G(s_0^S, i_1^T) \quad :$$

$$s_2^{S_2} = G(0, i_1^T - i_1^S) + G(s_0^S, i_1^S) = G(0, i_1^T - i_1^S) + H(s_0)$$

$$s_2^{S_2} = L(i_1^T) \qquad \text{(by Definition 2)}$$

but $G(s_0^S, i_1^T) = H(s_0)$ cannot be a constant over all s_0^S by Definition 1b, so $s_2^{S_2}$ depends on s_0^S, contradicting the hypothesis of linearity. QED.

Since the pattern vectors s, i, etc. could in fact be any mathematical object for which linear operations could be defined, time functions, distributions, etc., this theorem has a quite general impact. This formulation of the problem also recognizes that the learned information can be encoded in many ways other than the static magnitude of a state variable. Thus a system is a learning system in this sense relative to particular sets S and T. A continued theory of learning on this basis becomes a theory of the sets S and T.

Conclusions

Many of the problems arising from the apparently complex, nonlinear, nonanalytic character of industrial processes may be solved by taking an approach which allows easier use of our heuristic or intuitive understanding. This will require a bridge from traditional analytic control to the newer data-base orientation seen in the field of artificial intelligence. The work described in this paper attacks the problem from two directions. The use of a feature variable analysis reduces the process dimension and relates it to simpler, more human, and more general ways of describing process behavior. Current work on the problem of structure-free modeling is emphasizing the practical problems of managing large amounts of model data in the most flexible and general way.

From a theoretical point of view, the concept of treating signals in terms of well-defined parts or features allows a set-based theory of adaptation and learning with powerful implications. When combined with special conditions, such as the positive operators seen in process control, the pattern techniques could be especially powerful.

References

[1] M. A. KRASNOSELSKII: *Positive solutions of operator equations*, P. Noordhoff (1964).

[2] R. BELLMAN and E. F. BECKENBACK: *Inequalities*. SpringerVerlag, New York (1965).

[3] E. H. BRISTOL.: Adaptive control odyssey. *ISA Conference*, Philadelphia, PA, Paper 561-70, October (1970).

[4] T. SODERSTROM, L. LJUNG and I. GUSTAVSSON: On the accuracy problem in identification. *Proc. 6th IFAC World Congress*, Part 2, Paper 18.1, Cambridge, MA.

[5] K. ASTROM and P. EYKOFF: System identification — a survey. *Automatica* 7, 123-162 (1971).

[6] P. EYKOFF: *System Identification*. John Wiley, New York (1974).

[7] E. H. BRISTOL, G. R. Inaloglu and J. F. Steadman: Adaptive process control by pattern recognition. *Instrum. Control Systems* 101-105, March (1970).

[8] E. H. BRISTOL: Adaptation in the process industries. *Proc. the First Annual Advanced Control Conference*, Control Engng and Purdue University, April (1974).

[9] E. H. BRISTOL: Review of process control adaptation: *Decision and Control Conference*, Phoenix, AZ, November (1974).

[10] K. S. NARENDRA and M. A. L. THATHACHER: Learning automat--a survey. *IEEE Trans. Systems, Man Cybernetics* SMC-4 (4) 323-334, July (1974).

[11] R. L. BARRON: Application of learning networks in computer aided prediction and control. SME Technical Paper MS 75-515.

[12] A. G. IVAKHNENKO: Polynomial theory of complex systems. IEEE *Trans. Systems, Man and Cybernetics* SMC1 (4) 364 378, October (1971).