Reprinted from "Encyclopaedia of Linguistics, Information and Control"
PERGAMON PRESS
OXFORD · LONDON · EDINBURGH · NEW YORK
TORONTO · SYDNEY · PARIS · BRAUNSCHWEIG

ADAPTIVE CONTROL THEORY (The Structural and Behavioural Properties of Adaptive Controllers).

1. Introduction

The term '*adaptive*' has long been applied in the biological sciences to denote the plasticity of behaviour shown by an organism in its struggle to survive in a novel or changeable environment (Stanier 1953 Sommerhof 1950). More recently control engineers have designed controllers with a similar ability to modify their behavioural strategies in the face of unpredictable changes in the controlled plant or its input, and they have used the term 'adaptive' to qualify both this type of controller and its behaviour. The aptness of a common description for biological and synthetic systems remains to be demonstrated by future studies coupling the disciplines of biology and control-engineering, but research on 'adaptive control' exists in its own right as a body of experimental, practical and theoretical work central to modern automatic control, and it is the foundations of this work which form the subject of this review.

As in all novel fields of research the terminology of adaptive control has varied widely, and at one time the variety of applications of the term 'adaptive' made it potentially meaningless. It is now accepted that 'adaption' is a very rich concept with both structural and behavioural connotations which must themselves be further qualified, and also that it is not an ontogenic property of a controller but rather a triadic relationship between controller, controlled system, and the purpose of the controller. Since the last is 'arbitrarily' assigned by the designer or observer, the application of the term 'adaptive' depends on their point of views, (Mishkin and Braun 1961) and every controller is in some sense 'adaptive'. However, once a point of view has been defined and agreed, the extent and nature of a controller's adaptivity can be determined by well-defined procedures, and it is these which are the concern of adaptive control theory.

The synthesis of adaptive controllers is very well documented, both by specialist papers on the design and performance of specific controllers, and in review papers which attempt an overall survey and classification of adaptive control schemes (Truxal 1963; Aseltine *et al.* 1958; Donaldson and Kishi 1965). The evaluation of adaptive controllers in various environments, and the general problem of comparing the behaviour of different adaptive and learning controllers, have not been studied so intensively, since the variety of adaptive control schemes has been small and their behaviour simple. Now that complex learning controllers are being simulated, and adaptive controllers are being applied to non-linear and noisy environments where their approach to optimality is not uniform or may not necessarily occur, there is a growing interest in the *evaluation* and manipulation *of adaptive behaviour*. The literature on this is brief and dispersed, and this article is intended to summarize and expand it.

The next section introduces the concept of an adaptive controller as a device for the selection and implementation of a control policy from a set of allowable policies. The means for performing this selection give rise to different adaptive control structures, but the structure of a particular controller is arbitrary to the extent that the classification of allowable policies may be varied. In the third section the behaviour of such controllers is analysed by segmenting their interaction with the environment into elements called 'tasks'. The manner in which the controller's satisfactoriness varies whilst performing a sequence of tasks gives rise to various modes of adaptive behaviour, but the mode of adaption shown by a particular segment of behaviour is arbitrary to the extent that the classification of tasks may be varied. In the fourth section a state-description of adaptive behaviour is used to define the 'adaption-automaton' of a controller for a set of tasks, and the previous definitions are stated formally in terms of the structure of this automaton. Finally, in the fifth section, 'training' is introduced as a means of fabricating a controller by varying the initial task sequence given to an adaptive controller. This is equivalent to control of the adaption-automaton, and the strategies used in this control give rise to different types of training.

2. The Structure of Adaptive Controllers

In synthesizing a controller for a novel environment the designer has available a body of knowledge about the behaviour of a class of controllers in various environments. If one of these controllers will perform sufficiently well in the given environment then the design problem is solved, and the designer selects that controller. If none of the controllers is uniformly satisfactory for all possible conditions of the environment, then it may be possible to find a *set* of controllers at least one of which is satisfactory for each condition. If the designer is able to synthesize a device which implements the appropriate 'controller' (or rather 'control policy') according to the environmental conditions then his overall control system is an *adaptive controller*. The adaptive controller thus automates the selection of a control policy, previously performed by the designer, and is in that sense a 'self-organizing system'. The two-level structure of an adaptive controller is illustrated in Fig. 1: the lower level is coupled to the environment by its

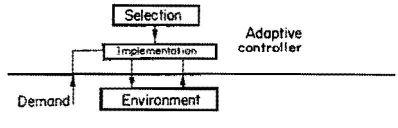

FIG. 1. *Adaptive controller.*

effector/receptor system, and receives information about the condition demanded in the environment; it implements a control policy (mapping between its inputs and outputs) which is specified by the upper level.

The strategy used by the upper level in selecting one control policy from the allowed set of policies may be used to distinguish different adaptive control structures. A major distinction is that between open-loop and closed-loop adaptive controllers (Carruthers and Levenstein 1963): an *open-loop adaptive controller* utilizes information obtained by 'identifying' the environment or demand-signal to select its control policy; a *closed-loop adaptive controller* utilizes information obtained by measuring the performance of its present and past control policies to select its next control policy. Thus the first type of controller implements a mapping from estimates of environment and demand conditions to its control policies, and relies upon the correctness both of these estimates and of the mapping. Whereas the second type of controller actually tests the appropriateness of its control policy and hence obtains feedback about the effects of its adaptive strategy. The two types of adaption are most powerful in combination, as illustrated in Fig. 2, since open-loop adaption can rapidly cope with expected environmental changes but

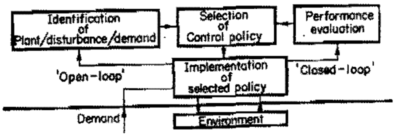

FIG. 2. *Structure of an adaptive controller.*

closed-loop adaption is necessary to check that this is operating correctly and to change it, perhaps by 'trial-and-error', if it is not.

The open-loop adaptive structure can be further classified by considering the canonical forms of the environment and of the system which is generating demand signals. Any physical system may be represented as a state-determined automaton with an output and a partially controllable input (that is a controllable input plus disturbance). A controller which identifies some property of the state of this automaton has been called, 'system-variable adaptive', and if it identifies the transitions of the automaton it has been called, 'system-characteristic adaptive' (Aseltine *et al.* 1958). A controller identifying some property of the demand signal has been called 'input-signal adaptive', although a better name would be 'demand-signal adaptive'. Six such distinctions may be made, although in practice they merge into one another since, for example, the effect of a disturbance may be ascribed to an unobservable distinction between states.

Closed-loop adaptive structures may be further classified into those which change their control policy randomly until one is found which gives a satisfactory performance, and those which change their control policy according to definite rules dependent on past policies and their performances. The former may be said to adapt by an 'evolutionary' process (Ashby 1960; Bremmerman *et al.* 1965), and the latter may be said to be 'hill-climbing systems'.

The structural definition of an adaptive controller as one having the form of Fig. 1 does not define the class of control policies from which the upper level may select, and since one control policy may often be regarded as a class of simpler policies the definition contains an arbitrary element. This arbitrariness is not apparent in the literature because the class of linear controllers (and in particular PID controllers) has been the only one of practical interest. Thus the relay controller has been classified as 'passively adaptive' (Aseltine *et al.* 1958) since its describing function (equivalent linear controller) has a gain varying inversely with the magnitude of the error, and hence the simple relay controller may be said to 'select' a control policy from the class of linear controllers consisting of a pure gain. If, instead of this class, one considered the class of control policies as 'linear-plus-relay and linear controllers', then the relay controller would no longer have an adaptive control structure, but the dual-mode controller would still be 'passively adaptive'.

Even the simple servo becomes an adaptive controller if the class of control policies is taken to be, 'those with a fixed output voltage'. This is not a trivial example, because the transient behaviour of the simple servo forms a good approximation to the behaviour of many adaptive loops, and also it offers an interesting case of the distinction between open- and closed-loop adaption: consider the very different effects of reversing the sign of the input to a simple servo and to a human operator in a tracking situation; the former now has positive feedback and oscillates, showing that it is only open-loop, system-variable adaptive; the latter changes the sign of his output (in less than half a second) showing that the human operator is closed-loop adaptive (Young *et al.* 1964).

The arbitrariness in the structural definition of an adaptive controller is not a defect, but rather central to the importance of adaptive concepts in the design of controllers. If the class of control policies were fixed then the 'adaptive controller' could be fully investigated until all its applications and its limitations were completely known, as has happened to the 'linear controller' at present. However, this is not so, for the 'adaptive controllers' of today will become the 'control policies' of tomorrow, and adaption may be seen as a design tool for extending the range of application of the available class of controllers.

3. *The Behaviour of Adaptive Controllers*

Even if a closed-loop adaptive controller is operating correctly and is always eventually able to select a satisfactory control policy, it may take so long to do this that it is worthless as a controller. Thus the designer is faced with the problem of not only evaluating the control policy of the adaptive controller but also evaluating its adaptivity in varying this policy appropriately. If the upper level of the adaptive controller were linear or permitted a linear approximation then its behaviour could be analysed conventionally on the basis of stability, time constants, and so on. However, the possibility of such an analysis is rare since, even if the lower level implements linear policies, the upper level tends to be designed as a discrete, decision-taking controller which does not permit a linear approximation. This section outlines a theory of controller behaviour which is applicable to these more general forms of adaptive controller as well as to the simpler linear, time-varying, controllers.

3.1. Segmentation of the controller environment interaction into 'Tasks'. The expected behaviour of an adaptive controller when coupled to an environment is that, if its control policy is not satisfactory for the condition of the environment, then it will eventually become satisfactory. Thus it must be possible to segment its interaction with the environment into two phases, in the first of which it is not satisfactory and in the second of which it has become so. This segmentation of the interaction between controller and environment into units, for which the controller is or is not satisfactory, is extended in the definition of a 'task' to form the basis for a taxonomy of adaptive behaviour.

A 'task' is a segment of the interaction between controller and environment for which it is possible to say whether or not the controller has performed satisfactorily. Equivalence relationships between tasks (so that it is possible to say, for instance, that an interaction consists of the same task repeated several times) are arbitrary, but will in practice follow from the natural relationships between different types of controlled system. A 'task' will typically consist of some specification of the plant parameters, initial conditions and period of interaction, together with a tolerable performance level below which a control policy is not satisfactory.

The importance of this segmentation into 'tasks' may be seen by considering the effect of performing a 'task' on the upper level of the adaptive controller. At the beginning of the interaction this level will be in some state which causes it to implement a particular control policy. At the end of the interaction it will be in another state, and it will be possible to say whether the overall control policy which has been implemented is satisfactory or not. If the controller is deterministic and the task is reasonably defined then the final state of the upper level will depend only on its initial state and the 'task' given to it. Thus the 'task' may be considered to be an 'input', in some sense, to the upper level of the adaptive controller. The description of the adaptive behaviour of the controller in terms of its performance of task sequences is then based on a state-description of the upper level of the controller, and this is applicable both to linear and non-linear controllers.

The segmentation of an interaction into tasks may be performed in many ways: the time of interaction between controller and environment may be fixed; a criterion for the termination of an interaction in terms of the behaviour itself may be given; the interaction may be terminated as soon as a decision can be made about the satisfactoriness of the controller. The 'termination' itself may be purely conceptual, a convenient division of a continuous sequence of behaviour into separate sub-sequences, or it may have a physical reality in that the plant is modified at the termination of an interaction. The next sub-section describes one example of the segmentation of an interaction into tasks.

3.2. An example of a set of tasks for a single-input, single-output controller. Consider the stable, noiseless second-order plant shown in Fig. 3, consisting of two

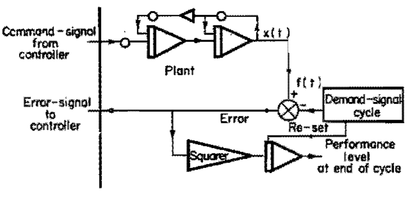

FIG. 3. *Example of a 'task' for testing the adaptivity of a controller.*

integrators in cascade with feedback from the output; its parameters are the gain, undamped natural frequency, and damping ratio. Let a task be defined by ascribing values to these three paramaters and the two initial values of the integrators, together with a time-varying demand signal, $f(t)$ for $0 \leq t < T$, and a decision procedure such that the interaction is satisfactory if and only if:

$$\frac{1}{T}\int_{t_0}^{t_0+T} [f(t_0+t)-x(t_0+t)]^2 \, dt < E^2,$$

where t_0 is the time at which the interaction starts, $x(t)$ is the output of the plant, and $E > 0$ is some tolerance on the r.m.s. error.

To test the adaptivity of a controller to the plant and demand conditions specified by such a task, it is connected to the plant and the demand-signal cycled with period T. After every cycle the task is complete, and the r.m.s. error during that cycle determines whether or not the controller has performed satisfactorily. If the controller is adaptive, the r.m.s. error in each cycle might be expected to decrease, and hence (by suitable choice of E) the controller will be unsatisfactory initially, but after a number of repetitions of the task it will become satisfactory and remain so. Many other forms of adaptive behaviour might arise, however: the controller could be always satisfactory or always unsatisfactory; it could start by being satisfactory and become unsatisfactory; it could vary between being satisfactory and being unsatisfactory, never settling at one or the other. If other tasks with different values of the plant parameters of demanded signal were interpolated, then the adaptive behaviour would become richer still. It is the description of this variety of possible behaviours which concerns the behavioural theory of adaption.

3.3. Varieties of adaptive behaviour. If a task is regarded as an 'input' to the upper level of an adaptive controller, and the satisfactoriness of its performance as an 'output', then, whilst the general theory of controllers considers all possible input sequences, the theory of adaptive controllers places especial emphasis on the effects of the repetition of a single input (for example the cycling of a task described in the example above); an adaptive system is characterized by its changing response to the same situation. This prompts the following definitions:

Acceptability—an interaction between controller and environment consisting of the repetition of a single task is *acceptable* if it is eventually always satisfactory.

Thus, in an acceptable interaction, the initial performance of the controller does not matter and for a number of repetitions of the task it may be satisfactory, unsatisfactory, or waver between the two. However, it must eventually become satisfactory and remain so; an acceptable interaction is one which reaches a *stable* condition of satisfactoriness.

Adapted—an interaction between controller and environment consisting of the repetition of a single task is *immediately acceptable* if it is always satisfactory; an immediately acceptable interaction is obviously acceptable. A controller in such a condition that it would have an immediately acceptable interaction with a task is *adapted* to that task.

The concepts of acceptability and adaptedness concern the controller's interaction with a single task, but the adaptivity of the controller is generally required because the particular task it must perform is incompletely specified or may change. The possible varieties of adaptive behaviour in a control situation involving a number of tasks are many, but there are three of particular interest which will be discussed in the following sub-sections.

3.4. Potential adaption. Very often an adaptive controller is used to perform a single task which will not change, but whose characteristics are unknown in advance. It must be capable of having an acceptable interaction with any of a range of tasks, but need not necessarily be capable of adapting to a sequence of different tasks. This prompts the following definition:

Potential adaption—a controller in such a condition that it will have an acceptable interaction with any one of a set of tasks is *potentially adaptive* to that set of tasks.

A potentially adaptive controller fulfils one aim of adaptive design in compensating for the designer's ignorance about the task to be performed. It will not necessarily fulfil another major aim by performing satisfactorily in a changing situation, since there is no implication that having adapted to one task it remains potentially adaptive to the others. Potential adaption is implied in statements like, 'a shoe adapts itself to the shape of a foot', and is the weakest form of goal-attainment to merit the designation 'adaptive'.

3.5. Compatible adaption. A controller which is designed to perform satisfactorily in a changing situation must not only adapt to its immediate task but must also remain potentially adaptive to other tasks which it may meet. This prompts the following definition:

Compatibly adapted—a controller is *compatibly adapted* to a task with respect to a set of tasks if, in an interaction consisting of a repetition of the task, it is not only always satisfactory but also potentially adaptive to the set of tasks.

A controller which is to be of use in a changing situation must be potentially adaptive to all those tasks which it is required to perform satisfactorily, no matter what tasks it has previously performed. This prompts the following definition:

Compatibly adaptive—a controller is *compatibly adaptive* to a set of tasks if, given any sequence of tasks from that set, it remains potentially adaptive to the set of tasks.

Thus a controller which is compatibly adaptive to a set of tasks will have an acceptable interaction with any one of those tasks, and no matter how it becomes adapted to one of them it will be compatibly adapted with respect to the remainder.

3.6. Joint adaption. A controller which is compatibly adaptive to a set of tasks is not necessarily able to become adapted simultaneously to all of them. It is, however, quite possible for two tasks to be so similar that a controller which is adapted to one may also be adapted to the other. This prompts the following definition:

Jointly adapted—a controller is *jointly adapted* to a set of tasks if, given any sequence of tasks from that set, it remains adapted to every member of the set.

Thus a controller which is jointly adapted to a set of tasks will be always satisfactory given any sequence of those tasks. This is a very strong condition, and an even stronger one is that when a controller adapts to one of a set of tasks it should eventually become jointly adapted to all of them:

Jointly adaptive—a controller is jointly adaptive to a set of tasks if it is both compatibly adaptive to the set and, during an acceptable interaction with any task in the set, it eventually becomes jointly adapted to the whole set.

3.7. Inter-relationships between different types of adaptive behaviour. The preceding definitions of adaptive behaviour have been given in order of increasing strength, for if a controller is jointly adaptive to a set of tasks it is also compatibly adaptive to them, and if a controller is compatibly adaptive to a set of tasks it is also potentially adaptive to them.

Jointly Adaptive $\Rightarrow$ Compatibly Adaptive $\Rightarrow$ Potentially Adaptive. These three modes of adaption are by no means exhaustive, and it will have been obvious that many variations are possible, defining other forms of adaptive behaviour. However, most of these would be regarded as pathological, in that no advantage is gained by designing them into the controller, and the three chosen for discussion are especially important in forming explicata of the common stereotypes of adaptive behaviour.

The definitions of *adapted, compatibly adapted, jointly adapted*, and *potentially, compatibly* and *jointly adaptive*, may be used to define binary relationships on the set of tasks, relative to a controller in a given condition; for example, $task_1$ is related to $task_2$ if the controller is compatibly adapted to $task_1$ with respect to $task_2$. All six relationships are reflexive, and only that induced by 'compatibly adapted' is not symmetric. However, only 'adapted' and 'potentially adaptive' induce relationships which are also transitive (and hence are equivalence relationships). For instance, a controller may be jointly adapted to $task_1$ and $task_2$, and also jointly adapted to $task_2$ and $task_3$, but given a sequence containing $task_1$ and $task_3$ there is no reason why even its potential adaptation to both tasks should not disappear. It is this lack of equivalence relationships which gives adaptive behaviour its extraordinary richness. A controller which shows no 'pathological' behaviour is rare, although the more drastic forms will be designed out if possible; for example, the relationship induced by 'compatibly adaptive' ought to be one of equivalence, for no sequence of normal tasks should be able to destroy a controller's ability to adapt to one of them.

3.8. Arbitrariness and triviality in the definitions of 'adaptive'. Just as the structural definition of an 'adaptive controller' contains an arbitrary element, because the classification of allowable control policies is left undefined, so do the behavioural definitions of 'adapted' and 'adaptive' contain an arbitrary element because the classification of tasks is left undefined. This arbitrariness need cause no confusion provided it is realized that at some stage in the discussion of an adaptive controller and its behaviour both these classifications must be agreed. Much early controversy over the application of the term 'adaptive' arose because the 'obvious', tacit classifications of one engineer were not those of another, or because disagreement over such classifications was wrongly ascribed to the definition of adaption itself.

Even when the arbitrariness in the definitions is accepted there remains the possibility that some types of adaptive structure and adaptive behaviour may be 'trivial'. Open-loop adaptive controllers have no feedback from their performance by which to guide their adaption, and it has been suggested that they be treated as 'unusual' non-linear but 'non-adaptive' controllers (Raible 1963). In the behavioural definitions of 'adaptive', 'jointly adaptive' is a very strong condition which is often trivial in practice—the tasks to which a controller becomes jointly adapted are equivalent and need not be distinguished. 'Potentially adaptive' is a very weak condition which again may often be regarded as trivial, because it is satisfied by systems showing an irreversible descent to equilibrium. 'Compatibly adaptive' adds the requirement of reversibility, and is closest to what is commonly regarded as being 'really adaptive'. However, although a compatibly adaptive controller shows the behaviour which one would expect, it may still do so 'trivially' by being adapted to all its tasks all the time, and hence showing no adaptive dynamics—it is just a very good, but static, controller!

In testing the behaviour of an artifact (or animal) for 'adaption' or 'learning' it may be desirable to eliminate this 'trivial' adaption; it is meaningful, for example, to ask whether an animal performs well in two different situations because it has a policy suited to both, or because it changes its policy according to the situation. To force the controller to show adaptive dynamics, one might say that it is 'really adaptive' to a task if it has an acceptable, but not immediately acceptable, interaction consisting of the repetition of that task. This is bound to occur if a controller is compatibly, but not jointly, adaptive to a pair of tasks, for there will then be a sequence of the tasks which causes it to become adapted to one but not the other. It has been suggested that a crucial test for 'learning' may be constructed by requiring an artifact to be compatibly adapted to two tasks, which are so defined that it is *logically* impossible for it to be jointly adaptive to them (Martens 1959).

4. *State-Descriptions of Adaptive Behaviour and 'Adaption-Automata'*

Although the definitions of various modes of adaptive behaviour have used the term 'condition' of a controller in such a way to imply that this is the 'state' of the upper level of an adaptive controller, there is no need to introduce structural considerations into state-descriptions of adaptive behaviour and the term 'condition' may be taken as what is actually defined. Such a view is taken in this section in order to introduce the concept of an 'adaption-automaton', and the preceding definitions are embedded in the automaton structure. A clearer picture of the inter-relationships between the various modes of adaption is then possible.

The mechanics for the introduction of a state-description into a system defined extensively by its behaviour are well-known (Zadeh 1964) and will be outlined only briefly. The object so far defined is a set of tasks, with relationships between them defined by the effect of sequences of tasks on the satisfactoriness of the controller environment interaction. The state of the controller should then be a description enabling its satisfactoriness to be predicted for each member of any sequence of tasks. Thus a description of the condition of a controller will be called its 'state' at the start of a sequence of tasks if it completely determines the satisfactoriness of the controller for each task in the sequence. If such a description is available of the controller at the termination of any task, it is said to have a complete state-description. If two descriptions of the conditions of a controller always predict identical sequences of satisfactoriness they are said to be equivalent. A complete state-description under the quotient mapping induced by this equivalence is said to be a minimal state-description.

This is merely a definition of 'state' for a fully controllable (not necessarily observable) state-determined automaton with inputs. The existence of a complete state-description implies that there is a minimal state description such that the state of the controller at the beginning of a task and the task itself together determine its state at the end of the task and the satisfactoriness of its performance. Thus there is defined an automaton whose state is a description from the minimal state-description of the controller, whose input is a task and whose output is the binary variable 'satisfactory' or 'not satisfactory'. This is the *adaption-automaton* of the controller for the set of tasks. It is not a finite-state machine, but considerations of acceptability give it many of the properties of one; for example, the output must retain the same value after a finite number of repetitions of a task for which the controller is potentially adaptive, and hence the states from then on are in some way equivalent. In the next-section the previously defined modes of adaption will be re-phrased in terms of the structure of the adaption-automaton.

4.1. Adaption-automata. An adaption automaton has a possibly infinite set of states, probably a finite set of inputs (tasks) and two outputs (satisfactory 0^+, unsatisfactory 0^-). A typical state will be represented by the letter, s (s_1, s_2 etc. if there are several); a typical input by the letter, t (t_1, t_2 etc. if there are several); and the outputs by symbols, 0^+, 0^-. Let an automaton in state s, be given a task t, such that its next state is s' and its output is o. We have the transition mapping:

$$s' = \sigma(s, t)$$

and the output mapping:

$$\text{o} = \theta(s, t).$$

Since we are interested in the effects of sequences of inputs, especially those generated by the repetition of a single task or by any means from a set of tasks, it is important to have a clear notation distinguishing beween tasks, sets of tasks, sequences of tasks and sets of sequences of tasks. A sequence consisting of task t_1 followed by task t_2 will be written t_1t_2 (with the obvious extension to longer sequences); a sequence consisting of the task t repeated n times will be written t^n. A typical set of tasks will be represented by the letter, T(T_1, T_2 etc.); a typical sequence of tasks will be represented by the letter, u(u_1, u_2 etc.). The set of sequences generated by the set of tasks, T, as free generators, will be written $U(T)$; that is, $U(T)$ is the set of all sequences of tasks which may be formed using the members of T. The mapping, σ, has an obvious extension from tasks to task sequences:

$$\text{if} \quad T = tT', \quad \text{then} \quad \sigma(s, T) = \sigma(\sigma(s, t), T').$$

4.2. Adaption sets. Let

$$W(T) \equiv \{s \mid \forall t \in T,\ \theta(s, t) = 0^+\};$$

that is, $W(T)$ is the set of states such that the controller will have a satisfactory interaction given any task from the set, T.

Let $$A(t) \equiv \{s \mid \forall n,\ \sigma(s, t^n) \in W(t)\};$$

that is, $A(t)$ is the set of states in which the controller is adapted to the task, t.

Let $$P(T) \equiv \{s \mid \forall t \in T,\ \exists N\colon \sigma(s, t^N) \in A(t)\};$$

that is, $P(T)$ is the set of states in wich the controller is potentially adaptive to the set of tasks, T.

Let
$$C_A(t, T) \equiv \{s \mid \forall t \in T, n,\ \sigma(s, t^n) \in W(t) \cap P(T)\};$$

that is, $C_A(t, T)$ is the set of states in which the controller is compatibly adapted to the task, t, with respect to the set of tasks, T.

Let $$C(T) \equiv \{s \mid \forall t \in T,\ u \in U(T),\ \sigma(s, u) \in P(T)\};$$

that is, $C(T)$ is the set of states in which the controller is compatibly adaptive to the set of tasks, T.

Let
$$J_A(T) \equiv \{s \mid \forall t \in T,\ u \in U(T),\ \sigma(s, u) \in A(t)\};$$

that is, $J_A(T)$ is the set of states in which the controller is jointly adapted to the set of tasks, T.

Let
$$J(T) \equiv \{s \mid \exists N\colon\ \forall t \in T,\ u \in U(T),\ \sigma(s, ut^N) \in J_A(T)\}$$

that is, $J(T)$ is the set of states in which the controller is jointly adaptive to the set of tasks, T.

This defines all the various modes of adaption previously described—they now appear as constraints on the structure of the adaption-automaton. There are many inclusion relationships between the adaption sets, of which the following are the most important:

$$\begin{aligned} W(T_1 \cup T_2) &= W(T_1) \cap W(T_2); \\ P(T_1 \cup T_2) &= P(T_1) \cap P(T_2); \\ C(T_1 \cup T_2) &\subset C(T_1) \cap C(T_2); \\ J(T_1 \cup T_2) &\subset J(T_1) \cap J(T_2); \\ A(t) &\subset W(t) \cap P(t); \end{aligned}$$

This last relationship is illustrated in Fig. 4: the space of all states is split into overlapping regions, $W(t)$, $P(t)$,

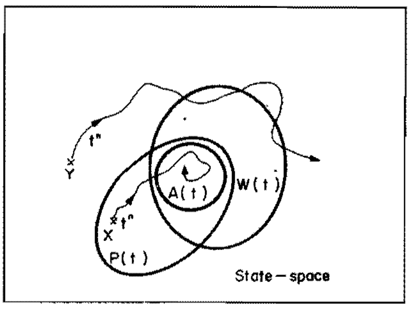

FIG. 4. *Behaviour of adaption–automaton for single task.*

$A(t)$; performance of the task, t, causes a transition from one state in the space to another; it is shown that the trajectory generated by a repetition of the task, t^n, starting at X in $P(t)$, must eventually enter $A(t)$ and, once in, it cannot escape; whereas a similar trajectory starting at Y outside $P(t)$ cannot enter $A(t)$, and if it should enter $W(t)$ must always leave it again. Thus $A(t)$ is the maximum set contained in $W(t)$ which is closed under the operator, t; it is the maximum 'trapping set' of $W(t)$ under t. Similarly $P(t)$ is the maximum trapping set for the whole space under t, which contains no non-transient states outside $W(t)$.

Further important relationships are:

$$C_A(t, T) = A(t) \cap P(T);$$
$$P(t_1 \cup t_2) \supset A(t_1) \cap A(t_2) \supset J_A(t_1 \cup t_2);$$
$$P(T) \supset C(T) \supset J(T).$$

Some of the relationships between potential, compatible and joint adaption for two tasks are illustrated in Fig. 5: the space of all states is split into regions $A(t_1)$,

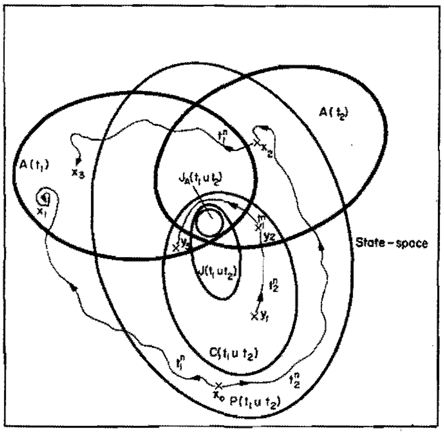

FIG. 5. *Behaviour adaption–automaton for two tasks.*

$A(t_2)$, $P(t_1 \cup t_2)$, $C(t_1 \cup t_2)$, $J(t \cup t_2)$; since the state X_0 is within $P(t_1 \cup t_2)$ but outside $C(t_1 \cup t_2)$, trajectories of the form t_1^n or t_2^n will eventually reach $A(t_1)$ or $A(t_2)$ respectively, but leave $P(t_1 \cup t_2)$ in so doing; this loss of potential adaptivity may not take place immediately (as in the trajectory from X_0 to X_1), as is shown by the trajectory from X_0 to X_2 under t_2^n which enters $A(t_2)$ within $P(t_1 \cup t_2)$ and hence is within $C_A(t_1 \cup t_2)$—on taking advantage of its adaptivity to t_1, however, it reaches X_3 where it has lost the possibility of returning to $A(t_2)$; $C(t_1 \cup t_2)$ is the maximum trapping set of $P(t_1 \cup t_2)$ under t_1 and t_2, and hence a trajectory starting from Y_1 within $C(t_1 \cup t_2)$ cannot escape from $P(t_1 \cup t_2)$ no matter what sequence of tasks it is given. $J_A(t_1 \cup t_2)$ is the maximum trapping set within $C(t_1 \cup t_2)$ which is contained in the intersection of $A(t_1)$ and $A(t_2)$, and $J(t_1 \cup t_2)$ is the maximum trapping set within $C(t_1 \cup t_2)$ which contains no non-transient states outside $J_A(t_1 \cup t_2)$.

The possible modes of adaptive behaviour are many, and only a few have been illustrated. It will have been apparent, however, that certain modes of behaviour are to be preferred—it would be pleasant to build a controller that always settled in $J_A(T)$, or, more realistically, that always remained in $C(T)$. This might not be possible at the design stage, however, and the initial state of a controller could be outside $P(T)$! Such a controller might well be very attractive if it were cheap to fabricate, and there existed a sequence of tasks which would cause it to enter $C(T)$ (where T is the set of tasks it is required to perform). This sequence of tasks would be a 'training' sequence given to the controller for the sole purpose of shaping its response to later tasks, and the possibility of 'training' adaptive controllers will be considered in the next section.

5. *Strategies for Training*

It has been assumed that the tasks given to the controller are those which it is required to perform satisfactorily, and hence the emphasis has been on task sequences consisting of a repetition of the same task; such a sequence is called a *fixed training sequence* for the task. The success of fixed training depends upon the controller being at least potentially adaptive to the task it has to perform, or, more strongly, compatibly adaptive to the set of tasks it has to perform. The selection of tasks by the trainer in fixed training involves no observation of the condition of the adaption-automaton, and the controller is given only those tasks which it is required to perform satisfactorily.

In *open-loop training* the trainer still does not observe the controller, but prepares it for adaption to the main tasks by giving it an initial sequence of auxiliary, or training, tasks for which it is not required to be satisfactory. Let:

$$P(T \mid u) \equiv \{s \mid \sigma(s, u) \in P(T)\};$$

that is, $P(T \mid u)$ is the set of states from which the training sequence, u, takes the controller to a state where it is potentially adaptive to the set of tasks, T. A controller whose adaption-automaton is in one of these states may not adapt to a task, $t \in T$, when given the fixed training sequence, t^n, but will do so if given the open-loop training-sequence ut^n; such a controller is said to be *potentially open-loop trainable* by the task sequence, u, for the set of tasks, T.

Similar conditional-adaption sets may be defined for *compatibly open-loop trainable:*

$$C(T \mid u) \equiv \{s \mid \sigma(s, u) \in C(T)\};$$

and for *jointly open-loop trainable.*

$$J(T \mid u) \equiv \{s \mid \sigma(s, u) \in J(T)\}.$$

There is obviously an even greater variety of modes of adaption with which to contend in conditionally adaptive situations—one may question whether a particular open-loop training sequence is always applicable, whether it can be harmful to the adaptivity toward some tasks, and so on. This variety is so great that it is not worthwhile treating it in a general terminology, and recourse must be made to the specific properties of the adaption-automaton under consideration.

5.1. Training as a control problem. An open-loop training sequence would be chosen to make the conditional-adaption sets as large as possible, so that they include all the possible initial states of the adaption-automaton. Some general restrictions, such as $P(T \mid u) \supset \supset P(T)$, may also be applied to ensure that the training does not destroy adaptivity which is already present. However, it may not be possible to find a single training sequence which has all the properties desired, and hence it may be necessary to apply different training sequences dependent on the initial condition of the controller. The only output from the adaption-automaton is a performance measure, the satisfactoriness of an interaction, and a trainer which utilizes this output in selecting the initial training sequence is said to be a *performance-feedback trainer*. A typical training strategy for this type of trainer would be to give one task until it is performed satisfactorily, and then another, and so on until the required task is being performed satisfactorily.

If other outputs from the adaption-automaton are made available and the trainer uses these in selecting the initial training sequence, then it is said to be a *feedback-trainer*. At this level of complexity training is itself a control problem: there is an 'environment', physically the controller, conceptually its adaption-automaton, whose inputs are tasks and one of whose outputs is the satisfactoriness of the previous interaction; the control problem is to take the automaton from an initial state, where the controller is not adapted to the task, to a final state where the controller is adapted to the task (or potentially, compatibly, or jointly, adapted to a set of tasks); the performance measure for this control problem may be based on the number of tasks given before the controller is adapted, or it may be a more complex cost function based on the cost of giving irrelevant tasks and so on.

The feedback-trainer solves this control problem by utilizing information about the state of the controller's adaption-automaton in selecting the task to be given to it. A stationary feedback-trainer implements a mapping between the state of the adaption-automaton at the end of a task and the next task to be given. This mapping will be a function of the tasks which the controller is required to perform satisfactorily, but, considering only one such task, let the mapping be:

$$t = \tau(s),$$

so that the transition and output equations may be written:

$$s' = \sigma(s, \tau(s)) = \sigma s, \quad \text{say;}$$
$$0 = \theta(s, \tau(s)) = \theta s, \quad \text{say.}$$

If the trainer is successful then ultimately the task should always be the required one, t_0 say, and the output should be always satisfactory—that is, for some N:

$$\forall n > N \quad \tau(\sigma^n s) = t_0$$
$$\theta\sigma^n s = 0^+.$$

If these conditions are not fulfilled then it may still be possible for the task sequence to contain t_0 frequently with a satisfactory interaction, and this mode of behaviour (which is shown in the symbiosis of animals and plants) may be advantageous if continual performance of a single task is not required (Schrodt 1965).

6. Summary and Conclusions

Adaptive controllers have a two-level structure in which the upper level selects a control-policy and the lower level implements it. The class of control-policies from which the upper level selects has generally been that of linear differential operators, but it is central to the importance of adaptive concepts in design that this class expands as our knowledge increases. The upper level of an *open-loop adaptive controller* utilizes information obtained by identifying the environment or demand-signal to select a control-policy. The upper level of a *closed-loop adaptive controller* utilizes information obtained by measuring the performance of its present and past control policies to select its next control policy.

The evaluation of the controller's adaptive strategy depends on the segmentation of its interaction with the environment into 'tasks', so that the behaviour of the controller may be regarded as the performance of a sequence of tasks for each of which it is, or is not, satisfactory. This evaluation reduces to an analysis of the stability, settling time, and so on, of the upper level regarded as a (highly non-linear) controller, which would be quite general were it not for the importance in adaptive control of input sequences consisting of the repetition of a single task. This leads to the definition of modes of behaviour which are peculiarly important to adaptive control theory.

A controller is *adapted* to a task if, given that task an indefinite number of times in sequence, it is always satisfactory. A controller is *potentially adaptive* to a set of tasks if, given any one of the tasks a number of times in sequence, it eventually becomes adapted to it. A controller is *compatibly adaptive* to a set of tasks if, given any sequence of the tasks, it remains potentially adaptive to them all; this is the explicatum of 'adaptive' nearest to its general usage. A special case of this, jointly adaptive, is so strong as to imply some triviality in the definition of the tasks—a controller is *jointly adaptive* to a set of tasks if in adapting to any one of them it eventually becomes adapted to all.

The *adaption-automaton* of a controller's adaptive behaviour is based on a state-description of the controller which enables its satisfactoriness for each of a sequence of tasks to be predicted. The definitions of different modes of adaption may be regarded as descriptions of the adaption-automaton's gross structure, and represent common behaviour in non-pathological controllers.

Training a controller consists of the manipulation of the inputs (tasks) to its adaption-automaton in order to force it into a condition where it is adapted to a particular task (or compatibly or jointly adaptive to a set of tasks). If the controller is already potentially adaptive to the task then *fixed training* may be used, in which it is given only the task for which it is required to be satisfactory. In *open-loop training* an initial sequence of auxiliary tasks, independent of the adaption-automaton's initial state, is given to the controller in order to make it potentially adaptive to the required task. In *performance-feedback training* the satisfactoriness of past interactions is used to select the training sequence given to the controller, and in *feedback training* general information about the state of the controller's adaption automaton is used in this selection.

The early adaptive controllers were too simple for training techniques to be of practical importance, but future '*learning machines*' may become commercially viable only as a general-purpose control-element which is fabricated uniformly and *trained* for a specific application. '*Teaching machines*' for the human adaptive controller (Gaines 1966) illustrate the use of training to synthesize a controller by manipulation of its initial environment rather than its internal structure, and the converse situation has been realized in which the human operator is used to train a learning-machine (Widrow and Smith 1964). Adaptive controllers were originally conceived for their insensitivity to variations in the controlled system, but the opportunities they offer for synthesis through 'training' may well become their main attraction.

See also: Control: basic elements. Learning machines. Automata, finite-state.

Bibliography

ASELTINE, MANCINI and SARTURE (1958) *A survey of adaptive control systems, Trans. I.R.E.*, AC-**6**, Dec., 102.

ASHBY (1960) *Design for a Brain*, London: Chapman and Hall.

ASHBY (1964) *The set theory of mechanism and homeostasis*, Yearbook of Soc. for Gen. Systems Res., Vol. 9.

BREMMERMANN, ROGSON and SALAFF (1965) *Search by Evolution*, in *Biophysics and Cybernetic Systems*, New York: Spartan Press.

CARRUTHERS and LEVENSTEIN (Eds.) (1963) *Adaptive Control Systems*, Oxford: Pergamon Press.

DONALDSON and KISHI (1965) *Review of adaptive control system theories and techniques* in *Modern Control Systems Theory*, New York: McGraw-Hill.

GAINES (1967) *Teaching machines for perceptual-motor skills*, Nat. Prog. Learn. Conf. Loughborough (1966) in *Aspects of the Technology of Teaching)*, London: Methuen.

MARTENS (1959) *Two notes on Machine learning, Inf. and Control*, **2** (4) Dec., 364.

MISHKIN and BRAUN (1961) *Adaptive Control Systems*, New York: McGraw-Hill.

RAIBLE (1963) *Comments on 'An approach to self-adaptive control', Trans. I.E.E.E.*, Ac-**8** (3), July, 270.

SCHRODT (1965) *Controlled cycle operations, Ind. Engng. Chem.*, **4** (1), Feb., 108.

SOMMERHOF (1950) *Analytical Biology*, Oxford: The University Press.

STANIER (1953) *Adaptation in Micro-organisms* (1953) 3rd. Symp. Soc. Gen. Microbiol., Cambridge: The University Press.

TOLMAN (1925) *Purpose and Cognition: the determiners of animal learning, Psychol. Rev.*, July.

TRUXAL (1963) *Adaptive control*, Survey paper, 2nd Congr. Intern. Fed. Automatic Control.

WIDROW and SMITH (1964) *Pattern recognizing control systems*, in *Computer and Information Sciences*, New York: Spartan Press.

YOUNG, GREEN, ELKIND and KELLY (1964) *Adaptive dynamic response characteristics of the human operator in simple manual control, Trans. I.E.E.E.* HFE, **5** (1), Sept., 6.

ZADEH (1963) *On the definition of adaptivity, Proc. I.E.E.E.*, **51** (3), March, 469.

ZADEH (1964) *The concept of state in system theory*, in *Views on General Systems Theory*, New York: Wiley.

B. R. GAINES