INFORMATION AND CONTROL **34**, 286–295 (1977)

An Adaptive Optimal Controller for Discrete-Time Markov Environments

IAN H. WITTEN

Department of Electrical Engineering Science, University of Essex,
Colchester, Essex, England

This paper describes an adaptive controller for discrete-time stochastic environments. The controller receives the environment's current state and a reward signal which indicates the desirability of that state. In response, it selects an appropriate control action and notes its effect. The cycle repeats indefinitely. The control environments to be tackled include the well-known n-armed bandit problem, and the adaptive controller comprises an ensemble of n-armed bandit controllers, suitably interconnected. The design of these constituent elements is not discussed. It is shown that, under certain conditions, the controller's actions eventually become optimal for the particular control task with which it is faced, in the sense that they maximize the expected reward obtained in the future.

INTRODUCTION

Two broadly different, and complementary, approaches to the design of adaptive controllers exist: On the one hand, are highly task-specific special-purpose controllers which—although adaptive—are designed with a particular type of plant in mind, while on the other, one finds general-purpose controllers embodying a minimum of assumptions about the task environment. The general-purpose controllers which have been proposed divide roughly into two types: those which employ a "teacher" to provide intelligent reinforcement signals, and those which rely solely on an (often noisy) indication of whether the environment is currently in a desirable state.

The first type of general-purpose controller can be subsumed under the heading of "pattern recognition." The controller has to associate actions condoned by the teacher with particular states of the environment. It need not attempt to evaluate the influence of its actions on the future behavior of the environment, since the reinforcement signals allow for these effects. By far the vast majority of the literature on adaptive or "learning" control systems deals with this type of controller.

In contrast, the problem faced by the second type of controller, though more

ISSN 0019–9958

general, has received little attention. (The only comprehensive published account of a controller of this type is that by Widrow *et al.*, 1973.) Here, the controller *cannot* just optimize its control strategy directly on the basis of the reinforcement received, for in doing so it would ignore the future repercussions of its actions. Instead, it must monitor and evaluate the effects of its actions and adjust its strategy indirectly, on the basis of this evaluation.

A novel general-purpose controller of the second type is described here. Although it has been used in simulated practical control situations, its major significance is theoretical rather than practical, and lies in the way in which it evaluates the worth of its actions and uses this evaluation to direct future adaptation. It is modular in structure and comprises an ensemble of controlling elements which interact both with each other and with the control task.

The controller operates in a task environment which provides instantaneous reward information and an indication of the state of the environment. The controller's task is to maximize the total expected reward over a long period of time. To accomplish this, it constructs a separate control policy for each state of the environment, amounting to a local optimization of the control action for that state. Because the aim is *global* optimization over a sequence of state transitions, interactions must be introduced between the local optimization problems to encourage the controller to traverse a valley in the reward space if this leads to a sufficiently high peak, and so it is not clear whether optimality of the local control is sufficient to guarantee global optimality.

In Section 3 it is shown that once the controller reaches a state of global optimality, it will never leave it (provided the variance of certain probability estimators is kept sufficiently low). Hence such a state will inevitably be reached, since one can easily show that no other nontrivial trapping states exist. Theorem 2, however, guarantees a more positive sort of "learning": each improvement in the performance of a local optimizer constitutes an improvement in the expected overall performance of the controller. So if the local optimizers behave well, the expected overall performance will climb monotonically toward its limit.

Included in the controller are probability estimators which provide interaction between the local optimization tasks. Because the probabilities being estimated depend on the state of the controller, and this is constantly fluctuating (at least in the initial stages of adaptation), the estimators must have nonzero terminal variance to accommodate these changing conditions. The effects of nonzero variance in the controller, and in particular the likelihood of an optimal control policy being corrupted, is examined qualitatively in Section 5.

1. The Environment

The environment is modeled as a stochastic automaton with n states, numbered 1, 2,..., n. At discrete instants of time, it emits (i) a reward $g \in [0, 1]$,

(ii) a state-output j, which indicates which state the environment is in; and accepts one of m control actions numbered 1, 2,..., m, causing its state to change. The reward output is a random variable whose mean value indicates how well the controller is doing. We assume for convenience that this mean value is a function only of the current state of the environment: Rewards which depend on the last state as well are easily accommodated. Thus the environment is characterized by the $n \times n \times m$ matrix p, where

$$p_{ijk} = \text{Pr[environment goes from state } i \text{ to state } j \text{ under control action } k]$$

and the reward expectation vector g, where

$$g_i = \text{E[reward value when the environment is in state } i].$$

Successive reward outputs are assumed to be statistically independent.

A *control policy* for the environment is a $n \times m$ stochastic matrix π, where

$$\pi_{ik} = \text{Pr[when the environment is in state } i, \text{ control action } k \text{ is taken]}.$$

Note that the choice of control action need not depend on anything other than the current state, since the environment obeys the Markov property. It is assumed that the controller knows the environment's state at all times.

The controller's aim is to find a control policy which maximizes the total expected reward over a long sequence of transitions. A *discount factor* $\gamma \in (0, 1)$ is introduced in order to weight the immediate future more heavily than the distant future. If a sequence of rewards h_0, h_1, h_2,..., h_r is obtained, the *discounted reward* is defined to be

$$(1 - \gamma) \sum_{s=0}^{r} \gamma^s h_s .$$

By manipulating γ, we can make the controller's goal short-term or long-term optimization.

Let $d_i(\pi, r)$ denote the expected discounted reward after r transitions, starting at state i of the environment, under policy π. A control policy π is said to be *optimal* if, for any other control policy π',

$$\operatorname*{Lim}_{r\to\infty} d_i(\pi, r) \geqslant \operatorname*{Lim}_{r\to\infty} d_i(\pi', r) \qquad \text{for all} \quad i \in \{1, 2,..., n\}.$$

That these limits exist is shown in Section 3.

Clearly, a policy can only be optimal if π_{ik} is either 0 or 1 for each i and k—unless there is a state i and two actions k and k', each of which leads to exactly the same expected discounted reward.

2. The Controller

The controller to be discussed is synthesized out of n "elementary controllers," which I call learning automata (LAs), one for each state of the environment. To provide motivation for the introduction of LAs, consider the special m-state, m-control-action environment with transition matrix

$$p_{ijk} = \delta_{jk} \qquad (\delta \text{ is the Kronecker delta}).$$

Here, action k always takes the environment to state k, regardless of the previous state. The optimal control policy consists of exclusively selecting actions k^* for which

$$g_{k^*} = \operatorname*{Max}_k \{g_k\}.$$

This is the m-armed bandit problem, an obvious generalization of the familier two-armed bandit problem which has been discussed extensively in the literature (Cover and Hellman, 1970; Shapiro and Narendra, 1969; Witten, 1973, 1974). Rather than tackling the design of a suitable "m-armed bandit controller," or *learning automaton*, let us assume that we are given the design, and investigate the possibility of connecting such LAs together to make an optimal controller for a general Markov environment. This should at least ensure that our controller performs satisfactorily on the restricted type of environment given by (1).

Specifically, the controller will comprise, for each state i of the environment:

(i) an LA, denoted LA_i, which will be called upon to select the controller's action when the environment's state-output is i, and which will be rewarded after this selection according to its success;

(ii) an estimate e_i of the expected discounted reward obtainable when the environment is in state i.

Suppose the environment is in state i, and LA_i selects control action k, which causes the environment to change its state to j and produce reward output g. Then e_i will be updated by

$$e_i \leftarrow (1 - \beta)e_i + \beta g' \qquad (\beta \in (0, 1) \text{ is a constant}),$$

where g' is a weighted average of the environment's reward g and the controller's new estimate of future reward:

$$g' = (1 - \gamma)g + \gamma e_j \qquad (\gamma \in (0, 1) \text{ is the discount factor}). \tag{3}$$

g' is also used as a "computed reward" to reinforce LA_i for its choice of control action k. When the controller has completed these updating operations, it will call upon LA_j to select the new control action, and the cycle will begin again.

Each of the n LAs is characterized, at any one time, by the m-vector of probabilities with which it will choose the control actions. When these vectors are put together as rows of an $n \times m$ matrix, they form the instantaneous control policy π for the controller. Thus, the state of the controller comprises its policy matrix π together with the vector e of current estimator values. We denote this state by $\langle \pi, e \rangle$.

3. Equilibrium States

Suppose the controller's policy is π. This induces a transition matrix τ on the environment:

$$\begin{aligned} \tau_{ij} &= \Pr[\text{environment goes from state } i \text{ to state } j] \\ &= \sum_k p_{ijk}\pi_{ik} \,. \end{aligned} \tag{4}$$

Now we can compute the mean value $\hat{e}_i$ of the estimator e_i, given π.

$$\hat{e}_i = \mathop{\mathrm{E}}_j \,[(1-\gamma)g_j + \gamma e_j], \tag{5}$$

where the expectation is taken over all states of the environment which can follow state i under policy π.

$$\begin{aligned} \hat{e}_i &= \sum \tau_{ij}[(1-\gamma)g_j + \gamma\, \mathrm{E}[e_j]], \\ &= \sum \tau_{ij}[(1-\gamma)g_j + \gamma \hat{e}_j]. \end{aligned} \tag{6}$$

Therefore,

$$\hat{e}(\pi) = (1-\gamma)(1-\gamma\tau)^{-1}\,\tau g. \tag{7}$$

Note that $(1-\gamma\tau)^{-1}$ exists, since τ is a stochastic matrix and so all eigenvalues of $\gamma\tau$ must be strictly less than 1 in absolute value.

Equation (6) shows that $\hat{e}_i$ is exactly the $\mathrm{Lim}_{r\to\infty}\, d_i(\pi, r)$ mentioned earlier, where $d_i(\pi, r)$ is the expected discounted reward obtained after r transitions, starting at state i, under policy π. Hence the limit exists, and the d_i's can be calculated from the control policy and the parameters of the environment, using Eqs. (4) and (7).

The ith LA is in an *equilibrium state* if its expected reward cannot be improved immediately by a change in its policy alone. This definition is motivated by the similar concept of an equilibrium state of a game (Nash, 1951). Indeed, it seems at first sight that some of the results concerning equilibria of games are immediately applicable to our controller, since the ensemble of LAs can be considered to be an n-person game with outcomes determined by the control environment. Unfortunately, this is not the case: A fundamental assumption in game theory is that the outcomes—our e's—are linear functions of the mixed strategies of the players, but (4) and (7) show clearly that e is not a linear function of π.

The state of the controller, $\langle \pi, e \rangle$, can alter in two ways. First, its control policy π can change. This corresponds to a change in the state of one or more of the constituent LAs. Second, the estimates e_i can change. However, if π remains constant the expected values of these estimates will approach $\hat{e}(\pi)$ exponentially at a speed which depends on β. Thus we call states $\langle \pi, e \rangle$ with

$$e = \hat{e}(\pi)$$

the *mean states* of the controller. If the system is started in a mean state, it will fluctuate about that state in a random manner, but the expected values of the estimates will not change.

The remainder of this section, and the next, treat the behavior of the controller in mean states. Effects of fluctuations about the mean states are examined in Section 5.

THEOREM 1. *Suppose the controller is in a mean state $\langle \pi, \hat{e}(\pi) \rangle$, and each of its constituent LAs is in equilibrium. Then if $\langle \pi', \hat{e}(\pi') \rangle$ is any other mean state,*

$$\hat{e}_i(\pi') \leqslant \hat{e}_i(\pi) \qquad \textit{for all } i.$$

Proof. Let

$$\begin{aligned} \Delta_i(\pi', \pi) &= \hat{e}_i(\pi') - \hat{e}_i(\pi) \\ &= \sum_j \tau'_{ij}[(1-\gamma)g_j + \gamma \hat{e}_j(\pi')] \\ &\quad - \sum_j \tau_{ij}[(1-\gamma)g_j + \gamma \hat{e}_j(\pi)] \\ &= \Gamma_i(\pi', \pi) + \sum_j \gamma \tau'_{ij}[\hat{e}_j(\pi') - \hat{e}_j(\pi)], \end{aligned}$$

where

$$\begin{aligned} \Gamma_i(\pi', \pi) = &\sum_j \tau'_{ij}[(1-\gamma)g_j + \gamma \hat{e}_j(\pi)] \\ &- \sum_j \tau_{ij}[(1-\gamma)g_j + \gamma \hat{e}_j(\pi)], \end{aligned} \tag{8}$$

$$\tau_{ij} = \sum_k p_{ijk}\pi_{ik},$$

and

$$\tau'_{ij} = \sum_{k} p_{ijk}\pi'_{ik} .$$

Then

$$\Delta = \Gamma + \gamma\tau'\Delta,$$

so

$$\Delta = (1 - \gamma\tau')^{-1}\Gamma. \tag{9}$$

Now $\Gamma_i(\pi', \pi)$ is not affected by any but the ith row of π', and so would be unchanged if π' were replaced by a new matrix, π'', equal to π except in the ith row where it is equal to π'. Hence $\Gamma_i(\pi', \pi)$ cannot be positive, since, as (8) shows, it represents the immediate improvement in the expected reward of LA_i if policy π'' were adopted, and π is an equilibrium state for LA_i.

$$\Gamma_i \leqslant 0 \qquad \text{for each } i.$$

Also,

$$(1 - \gamma\tau')^{-1} = \sum_{n=0}^{\infty} (\gamma\tau')^n,$$

a convergent series since all eigenvalues of $\gamma\tau'$ are strictly less than 1 in absolute value. Hence every component of $(1 - \gamma\tau')^{-1}$ is positive, and so

$$\Delta_i \leqslant 0 \qquad \text{for each } i.$$

4. Goal-Directedness

Let us consider the changes in policy as each of the LAs adapts to its local environment, under the assumption that this adaptation takes place much more slowly than that of the estimates e_i.

Theorem 2. *Suppose the controller is in a mean state $\langle\pi, \hat{e}(\pi)\rangle$. If one or more of the LAs changes its policy in the direction of increasing immediate computed reward, giving a new policy matrix π', then*

$$\hat{e}_i(\pi') \geqslant \hat{e}_i(\pi) \qquad \textit{for all } i.$$

Proof. From the assumptions of the theorem,

$$\Gamma_i(\pi', \pi) \geqslant 0 \qquad \text{for each } i,$$

where Γ is given by (8). So, from (9), and since all components of $(1-\gamma\tau')^{-1}$ are positive,

$$\Delta_i(\pi', \pi) = \hat{e}_i(\pi') - \hat{e}_i(\pi) \geqslant 0 \qquad \text{for all } i.$$

5. Effects of Statistical Variance

So far we have assumed that the controller is in a mean state, and have neglected fluctuations about mean states. We now examine the effects of variance in the controller and, in particular, whether it is likely that the inevitable random perturbations of the estimates will cause corruption of an optimal control policy.

Denote by σ_i^2 the variance of e_i when the controller has reached an optimal policy. Here, as well as distinguishing between the expected value $\hat{e}_i$ of the ith estimator and the random variable e_i representing the estimator's value at any given time, we make a similar distinction between g_i, the reward received on a particular occasion when the environment is in state i, and the mean reward $\hat{g}_i$ from that state. The updating rule for e_i is

$$e_i \leftarrow (1-\beta)e_i + \beta g', \tag{10}$$

and so, when the distribution of e_i has reached a stationary hyperstate,

$$\sigma_i^2 = (1-\beta)^2\,\sigma_i^2 + \beta^2\sigma^2\,[g'];$$

$$\sigma_i^2 = \frac{\beta}{2-\beta}\,\sigma^2\,[g'].$$

The random variable g' is selected from a set

$$\{g_1', \ldots, g_j', \ldots, g_n'\}$$

with probability

$$\{\tau_{i1}, \ldots, \tau_{ij}, \ldots, \tau_{in}\},$$

where

$$g_j' = (1-\gamma)g_j + \gamma e_j\,.$$

If the reward signal g_j is confined to the range [0, 1], then g_j' and hence g' must also fall in this range, and so

$$\sigma^2[g'] \leqslant \tfrac{1}{4}\,.$$

Therefore,

$$\sigma_i^2 \leqslant \beta/(4(2-\beta)). \tag{11}$$

When updating rules like (10) are used for averaging, $1/\beta$ acts as a time-constant for the estimator (see, e.g., Minsky and Papert, 1969). Let t be an approximate time-constant for the particular design of LA under consideration. Then the condition for the "goal-directedness" property to hold is that

$$t >> 1/\beta.$$

Since it is clearly advantageous to keep the σ_i's small, (11) requires that $1/\beta$ be large, and hence t must be very large.

Let us turn attention to the role of the discount factor γ. Denote by G_{ik} the computed reward obtained on a single choice of action k in state i, so that G_{ik} is selected from the set $\{g_1',..., g_j',..., g_n'\}$ with probability $\{p_{i1k},..., p_{ijk},..., p_{ink}\}$. e_i estimates G_{ik}'s instantaneous mean, averaged over the possible control actions k. Now

$$\hat{G}_{ik} = (1-\gamma)\sum_j p_{ijk}g_j + \gamma \sum_j p_{ijk}e_j \,.$$

This mean itself fluctuates with a variance

$$\gamma^2\sigma^2\left[\sum_j p_{ijk}e_j\right], \tag{12}$$

and, assuming that the variation of the e_j's is uncorrelated, this can be rewritten as

$$\gamma^2 \sum_j p_{ijk}^2\sigma^2\,[e_j] \leqslant \frac{\gamma^2\beta}{4(2-\beta)} \,. \tag{13}$$

Actually, some correlation between the e_j's will certainly exist, since if one were to change, the others would follow suit via the averaging procedure (2) and (3). However, since these changes take place slowly, the bulk of the variance in e_j will stem directly from the environment's rewards, and hence the correlations will be low. Expression (13) shows that the values which are being estimated fluctuate with a standard deviation proportional to the discount factor γ.

6. Conclusions

While the adaptive controller has been shown to achieve an optimal control policy when the variance of the estimators is neglected and the estimators are assumed to converge rapidly compared to the LAs, in practice the effects of nonzero variance and slow convergence may cause the control policy to deteriorate. The variance of the estimators can be kept low if: (i) γ is small (short-term optimization); (ii) β is small (slow estimating). The latter condition conflicts with the requirement that the estimators converge rapidly compared with the LAs, and so for reliable operation the time-constant of the LAs must be extremely large. Thus, the controller takes a long time to adapt to a new environment.

The performance of a controller constructed along the lines of the one described here has been studied experimentally (Witten and Corbin, 1973). Near-optimal control of a noisy third-order analog plant was achieved consistently, but, as predicted here, the time taken by the controller to discover a near-optimal policy was rather long. (In fact, it was typically 10,000 sampling periods, corresponding to about 30 minutes of real time.) However, using a general adaptive controller such as the system of LAs described here naturally incurs a penalty in convergence time. In general, control performance for particular tasks can (and usually should) be increased by building special-purpose constraints into the controller.

RECEIVED: May 31, 1974; REVISED: December 20, 1974

REFERENCES

COVER, T., AND HELLMAN, M. (1970), The two-armed bandit problem with time-invariant finite memory, *IEEE Trans. Inf. Theory* **IT-16**, 185.

MINSKY, M., AND PAPERT, S. (1969), "Perceptrons: An introduction to computational geometry," MIT Press, Cambridge, Mass.

NASH, J. F. (1951), Non-co-operative games, *Ann. Math.* **54**, 286.

SHAPIRO, I. J., AND NARENDRA, K. S. (1969), Use of stochastic automata for parameter self-optimization with multi-modal performance criteria, *IEEE Trans. Systems Sci. Cybernetics* **SSC-5**, 352.

WIDROW, B., GUPTA, N. K., AND MAITRA, S. (1973), Punish/reward: learning with a critic in adaptive threshold systems, *IEEE Trans. Systems, Man & Cybernetics* **SMC-3**, 455.

WITTEN, I. H. (1973), Finite-time performance of some two-armed bandit controllers, *IEEE Trans. Systems, Man & Cybernetics* **SMC-3**, 194.

WITTEN, I. H. (1974), On the asymptotic performances of finite-time two-armed bandit controllers, *IEEE Trans. Systems, Man & Cybernetics* **SMC-4**, 465.

WITTEN, I. H., AND CORBIN, M. J. (1973), Human operators and automatic adaptive controllers, *Int. J. Man–Machine Studies* **5**, 75.