Proceedings of the 1992 Winter Simulation Conference
ed. J. J. Swain, D. Goldsman, R. C. Crain, and J. R. Wilson

AUTOMATIC LEARNING: THEOREMS FOR CONCURRENT SIMULATION AND OPTIMIZATION

Sidney Yakowitz

Systems and Industrial Engineering Department
University of Arizona
Tucson, Arizona, 85721, U.S.A.

ABSTRACT

Many problems in machine learning and adaptive control can be rephrased as the task of finding the minimum of a function $f(x)$ on the basis of noisy observations,

$$Y(x) = f(x) + e(x). \qquad (1)$$

Building on a tradition of works by H. Robbins, L. Devroye, and others, we presume that $f()$ may be multi-modal and even discontinuous, and x may be multivariate, and that the decision maker may sequentially choose design points x_n on the basis of the history $\{(x_j, y_j), 1 \leq j < n\}$ of action/observation pairs. The noise depends on this history only through the choice of x_n.

Theory will be reviewed which assures that in various senses on-line convergence of $f(x_n)$ to the global minimum of $f()$ takes place. In application of the methodology to hard optimization problems in blackjack strategy, intervention for spatial epidemics, and elsewhere, we have found solutions through simultaneous simulation and optimization that might be very challenging to achieve by alternative techniques. These developments are the rudiments of a theory for "black-box" optimization, i.e., optimization which does not require detailed analysis of the underlying noisy environment. Some previous theory is surveyed, and a new result especially suited to a simulation environment is offered.

1 BACKGROUND AND A LEARNING ALGORITHM

1.1 A Motivational Problem

A common AI approach to certain deterministic games and puzzles involves representing the game as a graph, and at each "move",

i. Expanding a partial subgraph from the current vertex, as time allows,

ii. Heuristically assigning numerical values to the leaves of the subgraph according to intuitive criteria (such as, in chess, the extent to which ones pieces are defended, the center of the board controlled, etc.) , and

iii. Backing up these values by dynamic programming to find the best action from the current vertex.

Several years ago, your author sought to bring optimization ideas to bear on this setting. Suppose a sequence of statistically similar puzzles are to be solved. Then the opportunity exists to adjust the heuristic function in (ii) according to observed relative performance. Chess being so complicated, the simpler challenge of sorting 8-puzzle problems was selected for experimentation. The 8 puzzle (see Figure 1) is a simplified version of a common children's 15 puzzle. One can only slide a numbered tile to fill the blank, at each move. The object is to rearrange the tiles, initially placed at random uniformly among solvable puzzles, so that the numbers read in ascending order.

The vertex for the 8 puzzle graph is any representation showing ordering of the tiles (including the blank). Dr. E. Lugosi, choose as heuristic value function,

$$h(v; x) = d_M(v) + x \cdot d_C(v), \qquad (2)$$

where d_M and d_C are two distinct functions measuring the distance of the current vertex v from the fully-ordered vertex. The parameter $x > 0$ is the relative weight assigned to these two criteria, and the idea was to find the weight x^* that minimizes the average (i.e., expected) number of vertices expanded in reordering a randomly-chosen arrangement.

The author first attempted to do this by using Kiefer-Wolfowitz stochastic approximation on adjustment of x. He had expected that as more puzzles were solved, one would observe improved performance. But such improvement did not materialize. After endless fruitless verification of the program, it

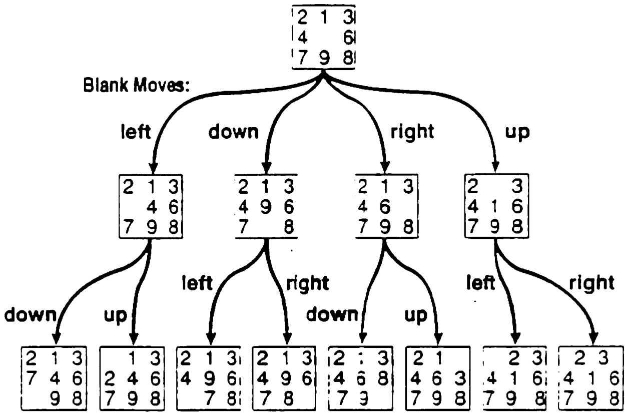

Figure 1: A Sub-graph for the 8 Puzzle

finally occurred to your slow-witted author that the performance function must be a step function, as a function of x, because there can be only a finite number of essentially different strategies, even though the weight x is defined on a continuum. The gist of the situation is that stochastic approximation is not applicable here! This saga motivated a more careful inquiry into automatic learning algorithms and this led to a satisfying attack on the 8 puzzle (Yakowitz and Lugosi (1990)). The present study constitutes a survey of our findings.

1.2 Automatic Learning for The Stochastic Minimization Problem

Definition 1 *Let $f(x)$ denote a bounded real-valued measurable function on an action set $\mathcal{X} \subset R^d$.* **The stochastic minimization problem** *is the task of sequentially choosing actions x_n, on the basis of a history of noisy observations*

$$y_j = y(x_j) = f(x_j) + e(x_j), j < n, \qquad (3)$$

of the objective function $f(x)$ in such a manner that in some specified sense, as $n \uparrow \infty$,

$$f(x_n) \to f_{min}, \qquad (4)$$

where f_{min} is the minimum (or Lebesgue-essential infimum) of $f(x)$.

In terms of the 8 puzzle, $f(x)$ is the expected number of node expansions required to rearrange a configuration randomly chosen from the set of solvable initial orderings, when the heuristic function weight in (2) is x. The noise $e(x)$ embodies the randomness of the initially-chosen board.

It is readily apparent that pointwise convergence in (4) is not generally attainable under randomness, even if the domain of actions is finite. We settle for convergence in probability, but attempt to establish rates. The only on-line learning strategies your author can conceive of are ones which partition the decision times, i.e., the entire set of positive integers, into two subsets, designated *sample times* and *resample times*, respectively. During resample times, the actions are chosen to be those yielding the lowest average observations, and at sample times, exploratory actions are chosen, in hopes of happening upon better values. The sample times become sparse, as the learning process evolves, so that improvement of average performance to the optimal is possible. Below we offer a generic learning algorithm, and the subsequent discussions will be aimed at stating properties of this algorithm under various postulates.

1.3 Some Algorithm Components

i. p(x) is any probability density function having as support all of a decision space $\mathcal{X}$.

ii. {NP(n)} and {N(n)} are nondecreasing unbounded sequences of integers such that

$$N(1) = NP(1) = 1, NP(0) = 0, P(n)N(n)/n \downarrow 0. \qquad (5)$$

Set n=1 and proceed to the iterative step.

1.4 The Iterative Step

New Sample Times (Steps 1 and 2)

If $NP(n-1) < NP(n)$, select a new point from $\mathcal{X}$.

1. Choose a point, designated by t(NP(n)), from $\mathcal{X}$ at random according to the pdf $p(x)$. Set $x_n = t(NP(n))$ and observe $y_n = y(x_n)$.

2. Start a running average $m_{NP(n)}$ for observations at t(NP(n)) by declaring

$$m_{NP(n)} = y_n,$$

and start a counter at t(NP(n)) by setting NS(NP(n))= 1. (Thus you see that "NP(n)" tells the number of New [sample] Points that have been gathered by time n, and "NS(j)" gives the Number of Samples that have been taken at a given sample point $t(j), j \leq NP(n)$.) Sometimes the argument n will be omitted.

Resample Times (Steps 3 and 4)

Else if $NP(n-1) = NP(n)$, resample at the apparently best point.

3. Let I* be any index i, $1 \leq i \leq NP(n)$, such that

$$m_i \leq m_j, 1 \leq j \leq NP(n). \qquad (6)$$

Set $x_n = t(I^*)$, and observe y_n.

4. Update the sample mean and sample counter.

Set

$$m_{I^*} = [m_{I^*} \cdot NS(I^*) + y_n]/(NS(I^*) + 1), \qquad (7)$$

$$NS(I^*) = NS(I^*) + 1. \qquad (8)$$

Assure at least N(n) observations at all points.

5. Skip this step if $NP(n+1) > NP(n)$. If, for some $j \leq NP(n), NS(j) < N(n)$, set n=n+1, set $x_n = t(j)$, observe y_n, and update m_j according to

$$m_j = [m_j \cdot NS(j) + y_n]/(NS(j) + 1) \qquad (9)$$

and set NS(j)=NS(j)+1. Repeat this step as necessary to assure that $NS(j) \geq N(n), 1 \leq j \leq NP(n)$.

6. Set n=n+1. Repeat the iterative step. (There is no stopping condition.)

This algorithm serves as a foundation for a host of methods, each appropriate for certain specific assumptions or possessing some specific property of interest. The remainder of this study gives an overview of such developments and extensions.

2 THEOREMS FOR ON-LINE LEARNING

2.1 The Discontinuous Objective Function Case

The primary intention of this section is to survey some theoretical results regarding the stochastic minimization algorithm with respect to the objective (3).

Assumption 1 *Assume there is a function $P(d,n)$ such that for $\hat{m}(x;n)$ the sample average of n independent observations of $Y(x)$, uniformly for $x \in \mathcal{X}$,*

$$P[|\hat{m}(x;n') - m(x)| > d, \text{ any } n' \geq n] \leq P(d,n). \qquad (10)$$

Regarding the objective function, assume that $f()$ is measurable on the borel set $\mathcal{X} \subset R^d$, and that f_{min} is the essential infimum, with respect to the search density $p(x)$ of the algorithm.

Theorem 1 *Take Assumption 1 to be in force. Furthermore, presume that all observations $\{Y(x(v_j))\}_j$ with $x(v_j)$'s identical, but v_j's distinct times, are mutually i.i.d.. Then if $N(n)=o(n)$ and $NP(n)$ is as in the learning algorithm, for a control sequence chosen by the algorithm, and for any $d > 0$, at resample times n, for some positive number $r < 1$,*

$$P[|f(x_n) - f_{min}| > d] \leq O(r^{NP(n)} + NP(n)P(d, N(n))) \qquad (11)$$

Corollary 1 *If $m \geq 2$ and*

$$\sup_x E[|e(x)|^m] < \infty \qquad (12)$$

or for some positive constants C_1 and C_2, and all $d > 0$,

$$P(d,n) \leq C_1 \exp(-C_2 d^2), \qquad (13)$$

then under (12), sequences NP and N can be chosen so that

$$\sum_{1 \leq v \leq n} P[|f(x_v) - f_{min}| > d] = O(\log(n)\, n^{1/m}), \qquad (14)$$

and under (13), so that

$$\sum_{1 \leq v \leq n} P[|f(x_v) - f_{min}| > d] = O(\log(n)^2). \qquad (15)$$

The theoremv and corollary are essentially those found in §4 of Yakowitz and Lowe (1991). However, here our condition (10) is stronger. It thereby allows observations taken during resample times to be used for updating the estimators m_i.

From Wagner (1969), for example, it is known that if (12) is satisfied, then $P(d,n)$ in (10) may be taken as $O(1/n^{m-1})$. Uniformly bounded random variables satisfy (13), as do 0-expectation Gaussian variables.

The noise variables called "generalized Gaussian" variables are those $e(x)$ which satisfy,

$$E\exp(u\,e(x)) \leq \exp(u^2 a), \tag{16}$$

for some constant a and all u. In §3, we show that for variables satisfying (16) uniformly in x, one can conclude that condition (13) is satisfied.

The rates of the corollary are attained with $NP(n) = O(log(n))$ and $N(n) = O(n^{1/m})$ for condition (12) and $NP(n) = N(n) = O(\log(n))$ for condition (13). Since in both cases

$$O(NP(n)N(n)/n) \to 0,$$

once may conclude that by randomizing the resample times, one can achieve,

$$P[|f(x_n) - f_{min}| > d] = g(n),$$

where $g(n)$ is the rate of (14) or (15), according to the appropriate condition.

The preceding developments and related material are from Yakowitz and Lowe (1991). In the case that there are but finitely many actions in $\mathcal{X}$, a strategy giving the provably optimal rate with respect to the criterion, $\sum_{1 \leq v \leq n} P[|f(x_v) - f_{min}| > d]$ is offered. From examination of the derivations of the other rates in the preceding reference, the reader will be able to show that (14) and (15) are within a factor of at most $\log(n)$ of being optimal.

2.2 Results for Smooth Objective Functions

The preceding developments did not require any smoothness assertions regarding the objective function $f()$. If, however, $f()$ is smooth, in some sense, then one ought to anticipate faster convergence. In fact, one can combine learning with stochastic approximation to achieve global convergence for multimodal functions without sacrifice of the fast rate enjoyed by stochastic approximation.

2.2.1 A Learning Rule with Kiefer-Wolfowitz Steps

The algorithm has the same basic structure as the one given in §1.2, and here we indicate only modifications to that rule needed for the Kiefer-Wolfowitz (KW) step inclusion. The intention is that the operation described in a step with a primed number replaces the indicated step in the earlier algorithm.

Additional Algorithm Components

iii. $\{a(i)\}$ and $\{c(i)\}$ are sequences such that for some positive constants A and C, $a(i) = A/i$ and $c(i) = C/i^{1/3}$.

Learning Alterations for KW Steps

2'. Additionally to the other parts of Step 2, define $H(NP)$ to be the hypercube centered at $t(NP)$ having sides of length $1/NP)^{1/4}$.

3'. Set $Y1 := Y(x_{I^*} + c)$, $Y2 = Y(x_{I^*} - c)$, and define

$$DY = 1/2c\,(Y1 + Y2).$$

Here $c = c(NS(I^*))$, and I^* is as in step 3. Then set

$$\tau = t(I^*) - a(NS(I^*))\,DY.$$

Then redefine

$$t(I^*) = \tau$$

provided the $\tau \in H(I^*)$. Otherwise, take $T(I^*)$ to be the point in $H(I^*)$ closest to τ.

4'. Use both values $Y1$ and $Y2$ when updating m_{I^*}.

2.2.2 Results for the KW Learning Rule

Assumption 2 *Assume that the objective function $f()$ is thrice continuously differentiable on $\mathcal{X}$, which is now presumed open. Also, the set of global minimizers is finite, and for f_{LOC} the values of the objective function on the set of points which are local but not global minima of f, we have*

$$f_{LOC} > f_{min}.$$

Finally, $f()$ is presumed locally strictly convex at its global minima, each such point being interior to $\mathcal{X}$.

Theorem 2 *Assume that Assumptions 1 and 2 hold, and that the noise variables satisfy (12) for some m > 2, and depend on the past only through the choice of x. Then at resample times n, and for some global minimizer x_{min}, a.s.,*

$$x_n \longrightarrow x_{min}. \tag{17}$$

Furthermore,

$$n^{1/3}(x_n - x_{min}) \tag{18}$$

is asymptotically normal, with mean vector 0.

These results are derived in Yakowitz (1992). Note that the proportion of epochs which are sampling times converges to 0, so that by randomizing, one can drop the restriction that n be a resample time.

2.3 A Brief Listing of Related Developments

Yakowitz *et al.* (1992a) have supplied methodology for the case that the noise is Markov dependent, a situation relevant to queueing and network problems. Pinelis and Yakowitz (1992) have studied the distribution of $f(x_n) - f_{min}$. In §3, we give some citations to related works by others.

On the applications side, Yakowitz (1989) has applied these techniques to develop a self-improving Go-Moku code, and Yakowitz and Kollier (1992) have recovered Thorp's (1966) tens-count standing-number table for blackjack by assigning a learning optimizer to each entry and simulating an enormous number of blackjack hands. Yakowitz *et al.* (1992b) apply learning to an idealized epidemiology question which was transcribed into a tough stochastic minimization problem.

The author's opinion is that these applications are as satisfying as any similar experiments on machine learning or adaptive control reported in the literature, with, of course, the exception of linear-dynamics/quadratic-objective problems. The reader should not accept such statements on faith, but duplicate the experiments in these citations and compare with any alternative strategies that come to mind.

3 A THEOREM FOR OFF-LINE LEARNING

In this section, we venture briefly into new territory, as far was we know. In notation of earlier sections, let

$$q_{min}(n) = \min_{t \leq n} f(x_t). \tag{19}$$

In *off-line learning* we seek a strategy such that in some sense,

$$q_{min}(n) \rightarrow f_{min}. \tag{20}$$

The reader will quickly confirm that (20) is a far weaker criterion than (3). In particular, if (3) holds, then (20) is likewise satisfied. The condition (20), which we here refer to as the *off-line* criterion, is not suited to on-line learning because there is no assurance that x_n, the action chosen at decision time n, is t_I^* by which the minimum $q_{min}(n)$ is achieved. In practical terms, one has no reason to think that under the off-line criterion, the average performance improves with increasing n. Many results regarding the stochastic minimization problem (e.g., Devroye (1976, 1978), Gurin (1965), Matyas (1965), Yakowitz and Fisher (1973)) are intended only for off-line minimization. On the other hand, the criterion (20) is appropriate for simulation analysis. The plan would be to allot a decision horizon M to testing, perhaps through simulation, and then selecting the action t_{I^*} which seems best at the end of this learning period for static on-line control. This plan brings us into at least tangency with some traditional problems in statistics and decision theory, as well as the literature for design of clinical trials, *etc.* Authors who have studied off-line learning in nonparametric settings have not, to our knowledge, concerned themselves explicitly with rates of convergence. However, under wide circumstances, conssistency has been established (e.g., the Devroye references above), and if one pays sharp attention to the proofs, the results reported here are fairly evident.

The contribution of the present study which apparently is new is that we take up the quest of prescribing a sequential design which is nearly optimal, in a certain sense, and we make an explicit statement regarding a rate. This initial foray is limited in scope; in particular, we assume a static strategy which does not alter the sampling plan as data accumulates. We continue to use notation from §1 associated with the stochastic minimization problem. The algorithm we prescribe here is as follows:

3.1 Off-Line Learning for The Stochastic Minimization Problem

The off-line algorithm is presumed to coincide with the "Automatic Learning" algorithm of §1, except for the specific details below, where we intend that the operation indicated by a primed number should replace the corresponding step in the on-line rule:

3'. If NP(n-1)=NP(n), then resample at $t(NP-1)$.

4'. Update the sample mean according to (7) and (8), but with NP=NP(n) replacing I^*.

5'. Omit step 5.

6'. Let M be the designated number of observations during the off-line learning phase. If n=M, then stop.

Assumption 3 *The value $f_{min} = 0$ is the p-infimum of the objective function f(x) over $\mathcal{X}$.*

Assumption 4 *The error variable e(x) in (1) satisfies either condition (13) or (16) of §2.*

We give here an off-line learning result which is in the spirit of Theorem 2.

Theorem 3 *Presume that the noise $e(x)$ depends on history only through the choice of x. Under Assumptions 1, 3 and 4 and using the off-line learning algorithm with $NP(i) = Int(i/\sqrt{M})$, for*

$$I^* := Argmin\{m_i, i \le \sqrt{M}\},$$

we have that for some positive number γ,

$$P[f(t(I^*)) > d] = O(\sqrt{M}\exp(-\gamma\sqrt{M})). \qquad (21)$$

Proof: Take $e = d/2$. Presume $q_{min} = q_{min}(M)$ *etc.*. Then

$$\begin{aligned} P[f(t(I^*)) > d] &\le P[q_{min} > e] \\ &+ \ P[\max_{1 \le i \le NP} |m_i - f(t(i))| > e] \end{aligned}$$

Let P1 and P2 denote the two probabilities at the right of (3.1). Then let $G(e) = P[f(t_1) \le e]$. By the assumption that f_{min}, the p()-essential minimum of $f()$, is 0, we have $G(e) > 0$ and

$$\begin{aligned} P1 &\le (1 - G(e))^{NP}, \\ P2 &\le NP \cdot P[|m_1 - f(t(1))| > e] = NP \cdot P3 \end{aligned}$$

Assume that S(n) is the sum of n independent observations of a generalized Gaussian RV with parameter $\alpha \le 1$. Then necessarily S(n) has expectation 0 and one may write that for $u > 0$,

$$\begin{aligned} P[|S(n)| \ge ne] &= P[u(|S(n)| - ne) \ge 0] \\ &\le \exp(-neu)\, E[\exp(u|S(n)|) \\ &\le \exp(-enu) \cdot 8\exp(n\,u^2) \end{aligned}$$

where the term at the end is a consequence of Azuma's inequality (e.g., Stout (1974), p. 238) with $a_i's = 1$. Now take $u = e/2$, to get that

$$P[|m(n)| > e] < 8\exp(-ne^2/4). \qquad (22)$$

Recognize that by scaling the noise and the tolerance e by $1/\sqrt{a}$, with a as in (16), one can always assure that if the i.i.d. summands are generalized Gaussian at all, one can satisfy that $\alpha \le 1$.

In view of (3.1) and (22) conclude that for some constants c_1 and ρ in the open unit interval,

$$P1 + P2 = c_1^{NP} + 8\,NP \cdot \rho^N.$$

If one sets

$$NP = N = \sqrt{M}$$

then evidently

$$P[f(T(I^*)) > d] = O(\sqrt{M}\rho^{\sqrt{M}}).$$

End of Proof

If one tries to improve this rate by increasing NP, for example, the value $NS \sim M/NP$ necessarily decreases, and this results in increase of $P2$, which is a tight inequality. Similarly, decreasing $P2$ can only come at the expense of increasing $P1$, The *log*() term does allow some leeway, but within this factor, the convergence rate cannot be improved.

This rate is quite an improvement over the probability of error, after M observations, in on-line learning under (16), which from (15), must be $O_p(\log(M)/M)$. Your author had intuitively mistakenly thought that the on-line learning rate of Theorem 1 was optimal, even in the off-line case.

ACKNOWLEDGMENTS

Whereas your author thinks that he is responsible for convergence rates described here, readers familiar with works by Robbins, Devroye, and others will see that the analytic machinery was already in place. Theorem 2 was attained while the author was at McGill University as a guest of Professor Luc Devroye, and this result and many others reflect conversations and incisive suggestions by Luc.

The research reported here was helped along by support from NSF grant ECS 8913642 and NIH grant RO 1 AI29462.

REFERENCES

Devroye, L. P. 1976. On the convergence of statistical search. *IEEE Trans. on Systems, Man, and Cybernetics* SMC-6: 46-56.

Devroye, L. P. 1978. Progressive global search of continuous functions. *Math. Programming.* 15:330-342.

Gurin, L. S. 1966. Random search in the presence of noise. *Engrg. Cybernetics* 4:252–260.

Matyas, J. 1965. Random optimization. *Automat. Remote Contr.* 26:244-251.

Pinelis, I and S. Yakowitz. 1992. The time until the final zero crossing of random sums, with application to nonparametric bandit theory. Submitted.

Stout, W. F. 1974. *Almost Sure Convergence* Academic Press. New York.

Wagner, T. J. 1969. On the rate of convergence for the law of large numbers. *Annals of Math. Statist.* 40:2195-2197.

Yakowitz, S and L. Fisher. 1973. On sequential search for the maximum of an unknown function. *J. Math. Analy. and Applic.* 41:234-259.

Yakowitz, S. 1989. A statistical foundation for machine learning, with application to Go-Moku.

Computers and Mathematics, with Applications. 71:1085-1102.

Yakowitz, S. and E. Lugosi. 1990. Random search in the presence of noise, with application to machine learning. *SIAM J. on Scientif. and Statist. Comput.* 11:702-712.

Yakowitz, S. and W. Lowe. 1991. Nonparametric bandit methods. *Annals of Operations Research* 28: 297-312.

Yakowitz, S. 1992. A Globally-convergent stochastic approximation. *SIAM J. Control and Optimization.* To appear in 1992.

Yakowitz, S. and M. Kollier. 1992. Machine learning with application to counting strategies for blackjack. *J. Statistical Plann. and Inference*, to appear in 1992.

Yakowitz, S., Jayawardena, T., and S. Li. 1992a. Theory for automatic learning under partially-observed Markov-dependent Noise, *IEEE Trans. Auto. Control.* to appear in December.

Yakowitz, S., Hayes, R., and J. Gani. 1992b. Automatic learning for dynamic Markov fields, with application to epidemiology. *Operations Research.* To appear, Aug./Sept.

AUTHOR BIOGRAPHY

SID YAKOWITZ received a PhD in EE from Arizona State University in 1967. Since then, he has held positions at the University of Arizona, where he is now Professor. He is author or co-author of books on adaptive control, simulation, and numerical methods, and about 80 refereed publications. In addition to the topics in this study, his research interests include epidemic modelling and control, and nonparametric statistical methods for time series and random fields.