AN AVERAGE-CASE ANALYSIS FOR A CONTINUOUS RANDOM SEARCH ALGORITHM

DIETMAR PFEIFER,* *Technical University Aachen*

Abstract

We give an upper bound for the average complexity (i.e. the expected number of steps until termination) for a continuous random search algorithm using results from renewal theory. It is thus possible to show that for a predefined accuracy ε, the average complexity of the algorithm is $O(-\log \varepsilon)$ for $\varepsilon \to 0$ which is optimal up to a constant factor.

AVERAGE COMPLEXITY; BINARY SEARCH; RENEWAL THEORY

1. Introduction

In recent years, attempts have been made to develop appropriate probabilistic tools and models for a unified treatment of an average-case analysis for certain classes of algorithms such as combinatorial algorithms (Barth (1983)) or searching algorithms covering for instance the famous simplex algorithm of linear programming (Ross (1983), Chapter 4.6) and others. The main tool there is the theory of finite-state Markov chains. This approach, however, is no longer applicable if continuous algorithms are considered, i.e. algorithms involving uncountable state spaces. In the present paper we want to show that a renewal-theoretic approach can be useful in such cases. For this purpose, we consider a random search algorithm which is shown to be asymptotically optimal up to a constant factor, i.e. the average complexity is of the same order as that of the best possible deterministic (i.e. binary) searching algorithm. For the sake of simplicity, the algorithm will be formulated only for the special case that the root of a monotonically increasing continuous function h on the unit interval with $h(0) < 0 < h(1)$ has to be approximated. All other possible applications including the determination of extrema of suitable convex or concave functions or more generally, the approximation of a (random) point in the unit interval with the possibility of exclusion of 'bad' solutions (as is typical in the above examples) can easily be derived from this.

2. The algorithm

We shall recursively construct random intervals $[U_n, V_n] \subseteq [0, 1]$ with $U_n \leq V_n$ and $[U_{n+1}, V_{n+1}] \subseteq [U_n, V_n]$, $n \in \mathbb{N}$ such that $h(U_n) \leq 0 \leq h(V_n)$, and $\bigcap_{n=1}^{\infty} [U_n, V_n] = x^*$ a.s. where x^* denotes the (unique) root of h in $[0, 1]$.

To start with, let $[U_0, V_0] = [0, 1]$, and $\{Y_n; n \in \mathbb{N}\}$ be an i.i.d. sequence of uniformly distributed random variables on $(0, 1)$ which are especially easy to generate by computer methods. $0 < \varepsilon < 1$ will denote the required accuracy of calculation.

Continuous random search algorithm

(a) If $h(U_n) < 0 < h(V_n)$ and $D_n = V_n - U_n > \varepsilon$, let

$$U_{n+1} = \begin{cases} U_n, & \text{if } h(U_n + D_n Y_{n+1}) > 0 \\ U_n + D_n Y_{n+1}, & \text{otherwise} \end{cases}$$

$$V_{n+1} = \begin{cases} V_n, & \text{if } h(U_n + D_n Y_{n+1}) < 0 \\ U_n + D_n Y_{n+1}, & \text{otherwise}; \end{cases}$$

then go to (a) with n being replaced by $n + 1$. Otherwise,
(b) terminate.

As can be seen from the above construction, either U_n or V_n is changed within a single step of the algorithm, depending on whether the conditionally uniformly (over $[U_n, V_n]$) distributed random variable $U_n + D_n Y_{n+1}$, given (U_n, V_n), leads to a non-positive or non-negative value under h. Clearly, $\{(U_n, V_n); n \in \mathbb{N}\}$ represents a two-dimensional Markov chain with uncountable state space, having the following convergence property.

Theorem 1. With probability 1, $h(U_n) < 0 < h(V_n)$ for all $n \in \mathbb{N}$, and

$$(1) \qquad \bigcap_{n=1}^{\infty} [U_n, V_n] = x^*.$$

Proof. The first part is obvious from the continuity of the distributions involved. For the second part, note that by induction,

$$(2) \qquad D_n \leqq \prod_{k=1}^{n} Z_k \quad \text{with} \quad Z_k = \max(Y_k, 1 - Y_k)$$

which is uniformly distributed over $[\frac{1}{2}, 1]$. Alternatively, for $W_k = 2(1 - Z_k)$, which provides an i.i.d. uniformly distributed sequence over $[0, 1]$, we have

$$(3) \qquad D_n \leqq \prod_{k=1}^{n} (1 - \tfrac{1}{2} W_k) \leqq \prod_{k=1}^{n} \exp(-\tfrac{1}{2} W_k) = \exp\left(-\tfrac{1}{2} \sum_{k=1}^{n} W_k\right),$$

hence $D_n \to 0$ a.s. $(n \to \infty)$ which is equivalent to (1).

Note that (2) provides an upper bound for $E(D_n)$ given by $(3/4)^n$ while a direct calculation via the conditional distribution shows that

$$(4) \qquad E(D_{n+1}) = \tfrac{1}{2} E(D_n) + E\left(\frac{(x^* - U_n)(V_n - x^*)}{D_n}\right) \geqq \tfrac{1}{2} E(D_n)$$

for $n \in \mathbb{N}$, giving a lower bound of $(1/2)^n$ for $E(D_n)$. With respect to the average-case behaviour of the algorithm, the random variable of interest will be the stopping time

$$(5) \qquad T = \inf\{n; D_n \leqq \varepsilon\},$$

giving the almost surely necessary number of steps until termination. Here, the right-hand side of (3) suggests that an application of renewal theory as in Russell (1983) or Jensen (1984) could give the desired result. In fact, we have the following theorem.

Theorem 2. With probability 1, $T < \infty$, and

$$(6) \qquad\qquad E(T) < -4 \log \varepsilon + 1.$$

Proof. We have

$$(7) \qquad\qquad T \leq \inf \left\{ n; \sum_{k=1}^{n} W_k \geq -2 \log \varepsilon \right\} = T^*,$$

say, which is almost surely finite with

$$(8) \qquad\qquad E(T^*) < -4 \log \varepsilon + 1$$

and

$$(9) \qquad\qquad E(T^*) = -4 \log \varepsilon + \tfrac{2}{3} + o(1) \quad (\varepsilon \to 0)$$

(see Russell (1983) and Jensen (1984)).

It is possible to show by information-theoretic arguments that, under the above conditions, a lower bound for the average complexity of any deterministic searching algorithm based on comparisons is roughly $-\log \varepsilon/\log 2 \approx -1.44 \log \varepsilon$ (see Knuth (1973), Chapter 6, p. 410), which can be achieved by a binary search algorithm with the unit interval being subdivided into intervals of length roughly ε, or by a modified continuous binary search algorithm with consecutive bisection of the remaining intervals (similar to the above algorithm) until the predefined accuracy ε is obtained. This shows that the algorithm under consideration is asymptotically optimal up to a constant factor with an average complexity of $O(-\log \varepsilon)$.

3. Concluding remarks

From a probabilistic point of view, the above result implies that an appropriate 'guessing' strategy (due to the subdivision of consecutive intervals according to a uniform distribution) can be an (almost) optimal searching technique. We conjecture that a corresponding result also holds for a larger class of searching algorithms, including those mentioned in the introduction.

Acknowledgements

References

BARTH, G. (1983) Analyzing algorithms by Markov chains. In *7th Symposium on Operations Research*, St. Gallen, Switzerland, 19–21 August 1982. *Methods of Operations Research* **45,** 405–418.

JENSEN, U. (1984) Some remarks on the renewal function of the uniform distribution. *Adv. Appl. Prob.* **16,** 214–215.

KNUTH, D. E. (1973) *The Art of Computer Programming*, Vol. 3: *Sorting and Searching.* Addison-Wesley, Reading, Mass.

ROSS, S. M. (1983) *Stochastic Processes.* Wiley, New York.

RUSSELL, K. G. (1983) On the number of uniform random variables that must be added to exceed a given level. *J. Appl. Prob.* **20,** 172–177.