Theoretical Bounds on the Complexity of Inexact Computations

JUDEA PEARL, MEMBER, IEEE

Abstract—**This paper considers the reduction in algorithmic complexity that can be achieved by permitting approximate answers to computational problems. It is shown that Shannon's rate-distortion function could, under quite general conditions, provide lower bounds on the mean complexity of inexact computations. As practical examples of this approach, we show that partial sorting of N items, insisting on matching any nonzero fraction of the terms with their correct successors, requires $O(N \log N)$ comparisons. On the other hand, partial sorting in linear time is feasible (and necessary) if one permits any finite fraction of pairs to remain out of order. It is also shown that any error tolerance below 50 percent can neither reduce the state complexity of binary N-sequences from the zero-error value of $O(N)$ nor reduce the combinational complexity of N-variable Boolean functions from the zero-error level of $O(2^N/N)$.**

I. Introduction

ONE OF the main aims of the science of computation complexity is to capture generic rules which govern exchanges among the important parameters of computations [1]. Tradeoffs among memory-space, program length, running time, and hardware cost have been the main focus of complexity studies [2]–[4]. Except in the area of numerical analysis, however, *precision* has hardly been regarded as a computational commodity. Errors have, traditionally, been assumed to have no place in computer systems and to be shunned at all cost.

Recently, there has been an increasing interest in the possibility of saving a substantial amount of computational resources by deliberately allowing some imprecision in computation procedures. The observation that humans, using inexact reasoning, can outperform machines in tasks such as parking a car or translating languages gave rise to the belief that, when problem complexity increases beyond a certain level, inexactness could be a blessing [5]. Systems operating on incomplete data with approximate inference rules have been constructed and successfully tested in the area of medical diagnosis [6]. Rabin [7] conjectured that the disparity between the superexponential complexity of proofs in even the simplest algebras and the apparent simplicity of everyday human planning activity can be explained by man's willingness to tolerate a small amount of error.

This paper uses information-theoretic considerations to establish absolute bounds on the reduction in complexity that can be achieved by tolerating a mean error of a specified type and magnitude. The method is applicable to many typical computation problems. One may wonder, for instance, if the $O(n \log n)$ comparisons necessary for sorting a sequence of n items could be reduced substantially when one settles for only partially sorted outputs. Or, as another example, while it is well known that the exact realization of most Boolean functions of n variables requires at least $O(2^n/n)$ two-input gates, one may be willing to tolerate a certain percentage of erroneous outputs and may inquire whether the expected number of gates could be reduced by a significant factor.

Our model of inexact computation consists of a task environment (functions, transformation, etc.) and a class of computation units (machines, algorithms, sequences of operations, etc.) which could be employed to accomplish the required tasks. When no error is allowed, each task is accomplished by a distinct computation unit. When error is tolerated, one may exploit the option of assigning some of the tasks, especially the most frequent and most complex ones, to less complex processing units which do not exactly accomplish the requested tasks but produce outputs somewhere in the neighborhood of the desired ones. Our problem is to bound from below the mean complexity of the resultant computation over all assignments which preserve a fidelity criterion of a specified level.

II. Problem Statement

The model construed to facilitate the analysis has the following components:

i) a finite set F of tasks;
ii) a countable set Ω of machines (or computation units);
iii) a probability measure P on the subsets of F;
iv) a complexity measure $c: \Omega \rightarrow \{1,2,\cdots\}$;
v) a penalty function $d: F \times \Omega \rightarrow [0,\infty)$, $d(f,\omega)$ being the cost incurred when task "f" is assigned to machine "ω";
vi) a set of assignments $m: F \rightarrow \Omega$, each giving rise to an average cost

$$\sum_{f \in F} P(f) d(f, m(f))$$

and an average complexity

$$\sum_{f \in F} P(f) c(m(f)).$$

Manuscript received July 18, 1975; revised March 8, 1976. this work was supported in part by the National Science Foundation under Grants GJ 42732 and MCS-18734.

The author is with the School of Engineering and Applied Science, University of California, Los Angeles, CA 90024.

The main problem of this paper is the underbounding of the function

$$C(D) = \min \sum_{f \in F} P(f)c(m(f)), \tag{1}$$

the minimum being over assignments m with average cost not exceeding D.

Remarks: The assignment m may be many-to-one allowing for the execution of several tasks by the same machine. We exclude, however, the option of employing nondeterministic assignments, thus $|m(F)| \leq |F|$. Assignments m with average cost not exceeding D will be referred to as D-admissible computations.

In certain cases, one would be more interested in complexity measures other than $C(D)$; for example, the maximum complexity $\max_{\omega \in m(F)} c(\omega)$, or the mean static complexity

$$\frac{1}{|m(F)|} \sum_{\omega \in m(F)} c(\omega).$$

Bounds on these measures are usually easier to obtain and will be treated only briefly in the sequel.

The following three examples illustrate the applicability of the model.

Example 1: Consider the problem of realizing an arbitrary Boolean function of N variables using logic-circuit technology. The task environment F consists of the collection of all N-variables Boolean functions f: $\{0,1\}^N \rightarrow \{0,1\}$ with $p(f) = 2^{-(2^N)}$. Ω is the set of logic circuits available for the realization of F, and c is usually taken as the number of gates employed by each circuit (combinational complexity) or the maximum number of logic levels between the input and the output (time complexity). A reasonable measure of error-induced penalty would be the fraction of arguments over which the circuit output would not agree with the specified function.

Example 2: Consider the problem of generating specified binary sequences of length N using sequential circuits. The task environment consists of functions f: $\phi \rightarrow \{0,1\}^N$ with $P(f) = 2^{-N}$ and Ω being the set of autonomous sequential machines. An accepted measure of complexity is the number of states which the machine must employ, and a reasonable penalty function would be the Hamming distance between the desired sequence and the one generated.

Example 3: Consider the problem of devising a sorting algorithm for partially sorting N items with a minimum number of binary comparisons [3, p. 86]. Each task can be regarded as that of executing one permutation among $N!$ possible ones. Since only binary comparisons are allowed, the execution of each such task would consist of a sequence of binary decisions based on the relative order of the two items inspected. We may choose, therefore, to regard each such decision sequence as the basic computation unit under consideration. Ω would comprise then the set of sequences of binary decisions, and the complexity measure would be the length of each such sequence. If, upon receiving the input list f_1, the algorithm goes through a decision sequence $\omega(f_1)$, then the cost function $d(f_1,\omega(f_1))$ should reflect the damage incurred by receiving the partially sorted list produced by $\omega(f_1)$ instead of the fully sorted one desired. The appropriate cost function depends on the usage finally made of the sorted lists; in some cases the number of items placed in the wrong rank might reflect that damage, in others it might be the number of pairs produced out of order.

III. Lower Bound for $C(D)$

The exact calculation of $C(D)$ may be a very difficult undertaking, because the relation between $c(\omega)$ and $d(f,\omega)$, for given f and ω, are hard to establish (see examples). Lower bounds for $C(D)$ can be obtained using information-theoretic arguments based on Shannon's rate-distortion theory [8]. The main result which we shall borrow from the theory is Shannon's source-coding theorem: given an information source $[K,P]$ over an input alphabet K, an output alphabet L, and a distortion measure $d(k,l)$; it is not possible with any coding scheme to transmit the source information at a rate less than $R(D)$ and with average distortion not exceeding D. $R(D)$ is called the rate-distortion function and is given by

$$R(D) = \min_{P(l|k) \in \boldsymbol{P}_D} \sum_{k,l} P_k P(l|k) \log \frac{P(l|k)}{P_l} \tag{2}$$

where

$$\boldsymbol{P}_D = \left\{ P(l|k) : \sum_{k,l} P_k P(l|k)\, d(k,l) \leq D \right\}. \tag{3}$$

$R(D)$ has been studied extensively in the literature on information theory [9] and has been calculated for a variety of distortion criteria and source statistics.

A direct corollary of the source coding theorem is as follows.

Corollary: Every D-admissible computation yields an entropy not smaller than $R(D)$, that is,

$$H(Q) = -\sum_{\omega \in m(F)} Q(\omega) \log Q(\omega) \geq R(D) \tag{4}$$

where Q is the probability induced on Ω by mapping m

$$Q(\omega) = \sum_{f:m(f)=\omega} P(f). \tag{5}$$

This theorem follows from identifying $[F,P]$ with the information source, Ω with the output alphabet, and m with a source code. A violation of (4) would mean that m exists by which $[F,P]$ could be encoded at rate $H(Q) < R(D)$ and with distortion $\leq D$, so violating Shannon's theorem.

Using (4) one can readily bound $|m(F)|$ by

$$|m(F)| \geq \exp [R(D)] \tag{6}$$

where m is D-admissible; this follows from the fact that the entropy of any N-element ensemble cannot exceed $\log N$ (nats).

Note that the minimum size of $m(F)$ can, in turn, yield bounds on static complexity measures such as the maximal

complexity or the mean static complexity. Both achieve their minimum values when the machines employed under m are sequentially chosen from Ω in order of increasing complexity, and when their number $|m(F)|$ is the smallest allowable by (6). Let $g(x)$ denote the number of machines in Ω having complexity exactly equal to x, $x = 1,2,\cdots$; then inequality (6) gives the bounds

$$\max_{\omega \in m(F)} c(\omega) \geq n_s \tag{7}$$

and

$$\frac{1}{|m(F)|} \sum_{\omega \in m(F)} c(\omega) \geq e^{-R(D)} \sum_{x=1}^{n_s} x g(x) \tag{8}$$

where n_s is the smallest integer satisfying

$$\sum_{x=1}^{n_s} g(x) \geq \exp [R(D)]. \tag{9}$$

Equations (7) and (8) are generalizations of complexity bounds commonly produced by enumeration [2] under the assumption that $D = 0$ and that $P(f)$ is the uniform distribution so that $R(D) = \log |F|$.

Bounds on operational complexity measures, such as $C(D)$, require a further analysis since (6) does not constrain the probability distribution on $m(F)$; hence the mean complexity might be further decreased by assigning higher probabilities to machines in the lower levels of complexity. The essence of (4), on the other hand, is that, in order to achieve a mean distortion not exceeding D, the entropy H could not be too low and so the probability distribution Q could not be too uneven. This, in turn, limits the difference between $C(D)$ and the static mean complexity.

We now produce a lower bound on $C(D)$ by calculating the minimum of $\Sigma Q(\omega)c(\omega)$ over all probability assignments $Q(\omega)$ which satisfy (4):

$$C(D) \geq \min_{Q \in \boldsymbol{P}_{R(D),F}} \sum Q(\omega)c(\omega) \tag{10}$$

where $\boldsymbol{P}_{R,F}$ is the set of all R-admissible probability assignments on at most $|F|$ machines:

$$\boldsymbol{P}_{R,F} \triangleq \{Q(\omega): H(Q) \geq R, \ |\{\omega: Q(\omega) > 0\}| \leq |F|\}. \tag{11}$$

Note that the right side of (10) is much easier to calculate than $C(D)$; the constraint $Q \in \boldsymbol{P}_{R,F}$ does not involve d or P.

It will now be shown that $\Sigma Q(\omega)c(\omega), Q \in \boldsymbol{P}_{R,F}$, is minimized by selecting a set of $|F|$ machines from the lowest complexity levels of Ω and assigning them exponentially decreasing probabilities.

Lemma 1: Let $Q^* \in \boldsymbol{P}_{R,F}$ be a probability assignment such that, for all $Q \in \boldsymbol{P}_{R,F}$,

$$\sum Q^*(\omega)c(\omega) \leq \sum Q(\omega)c(\omega).$$

Then

$$H(Q^*) = R. \tag{12}$$

In other words, Q^* is on the boundary of $\boldsymbol{P}_{R,F}$.

Proof: Consider any arbitrary subset $\Omega' \subseteq \Omega$ containing at most $|F|$ machines. The set of R-admissible probability assignments on Ω' constitutes a convex region, since $H(Q)$ is a strictly convex $\cap$ function of Q. On the other hand, $\Sigma_{\Omega'} Q(\omega)c(\omega)$ is linear in Q and so must assume its unique extremum on the boundary $H(Q) = R$. Applying this argument to all subsets Ω' proves the lemma. Q.E.D.

Lemma 1 permits the reduction of the minimization in (10) to a simple variational problem with the equality constraint $H(Q) = R$. Its solution leads to an exponential probability distribution

$$Q^*(\omega) = \frac{e^{\mu c(\omega)}}{\sum_{\omega} e^{\mu c(\omega)}} \tag{13}$$

for any set of $|F|$ machines, where μ is a Lagrange multiplier satisfying (12). Clearly, $\Sigma Q(\omega)c(\omega)$ is minimized when the set of $|F|$ machines are "packed" into the lowest n levels of complexity, where n is the smallest integer satisfying

$$\sum_{x=1}^{n} g(x) \geq |F|. \tag{14}$$

Treating μ as a variable parameter, (12) and (13) constitute a pair of parametric equations relating R and the average complexity C, viz.

$$C_\mu = \frac{\sum_{x=1}^{n} x g(x) e^{\mu x}}{\sum_{x=1}^{n} g(x) e^{\mu x}}, \qquad -\infty < \mu \leq 0 \tag{15}$$

$$R_\mu = \log \sum_{x=1}^{n} e^{\mu x} g(x) - \mu C_\mu, \qquad -\infty < \mu \leq 0. \tag{16}$$

The solution to (15) and (16), which we call the *complexity rate* function $C_g(R)$, provides the sought for lower bound

$$C(D) \geq C_g[R(D)]. \tag{17}$$

Thus, given a value D for the tolerated mean distortion, $C(D)$ can be bounded by first calculating the rate-distortion function $R(D)$, and then the complexity rate function $C_g(R)$ at $R = R(D)$. The conditions under which upper bounds on $C(D)$ may have a similar relationship to $R(D)$ remain an open question.

IV. Properties of $C_g(R)$, the Complexity Rate Function

Several typical $C_g(R)$ functions are depicted in Fig. 1. Whereas the exact nature of $C_g(R)$ depends on the occupation function $g(x)$, some general characteristics of this function can be deduced from (15) and (16).

A) Successive differentiations of (15) and (16) with respect to μ yield

$$\frac{dC_g(R)}{dR} = -\frac{1}{\mu} \geq 0 \tag{18}$$

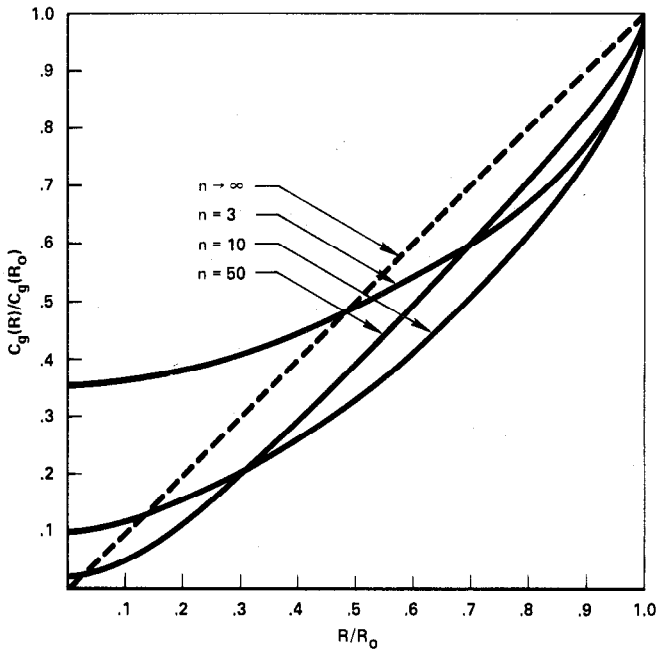

Fig. 1. Complexity Rate Function $C_g(R)$, for $g(x) = x2^x$, $R_0 = \log |F|$.

$$\frac{d^2C_g(R)}{dR^2} = -\mu \frac{\left[\sum_1^n g(x)e^{\mu x}\right]^2}{\sum_1^n g(x)e^{\mu x} \cdot \sum_1^n x^2 g(x)e^{\mu x} - \left[\sum_1^n xg(x)e^{\mu x}\right]^2} \geq 0. \tag{19}$$

Thus $C_g(R)$ is a monotonically increasing convex $\cup$ function of R.

Since $R(D)$ is known to be a monotonically decreasing convex $\cup$ function of D with infinite slope at $D = 0$ [8], we conclude that $C_g[R(D)]$ is also a monotonically decreasing convex $\cup$ function of D with infinite slope at $D = 0$. This supports the notion that greater reductions in complexity can be achieved at low levels of distortion than at high levels.

B) The low-distortion behavior of $C_g(R)$ is governed by the slope

$$\left.\frac{dC_g(R)}{dR}\right|_{D=0} = -\frac{1}{\mu_0}$$

which depends on the input statistics $P(f)$ via (15), (16) and the relation $R_{\mu_0} = H(P)$.

If the input functions are all equiprobable, then $\mu_0 = 0$, $R(O) = \log |F|$, and the low-distortion behavior of $C_g(R)$ is given by the square-root relation

$$\Delta C_g = (2\Delta R)^{1/2} V_g(n) \tag{20}$$

where

$$V_g(n) = \left[\frac{\sum_1^n x^2 g(x)}{\sum_1^n g(x)} - \left(\frac{\sum_1^n xg(x)}{\sum_1^n g(x)}\right)^2\right]^{1/2}. \tag{21}$$

C) Given two distinct occupation functions $g_1(x)$ and $g_2(x)$ such that $g_1(x) \geq g_2(x)$, for all x, then $C_{g_1}(R) \leq C_{g_2}(R)$. This follows from the fact that if $g_1(x) > g_2(x)$, for some $x \leq n$, then any probability assignment optimal under $g_2(x)$ can be further improved under $g_1(x)$ by filling in the "gap" created at x without destroying its R-admissibility. This property implies that the complexity rate function calculated on the basis of an upper bound on $g(x)$ (instead of $g(x)$) still provides a lower bound to $C(D)$. In some cases, to ease the combinatoric labor of computing the exact value of $g(x)$, a reasonable upper bound would suffice.

D) We will now show that if $g(x)$ is a fast growing function with at least an exponential rate of growth, then the complexity rate function $C_g(R)$ approaches a straight line as $n \to \infty$.

Theorem 1: Let $g(x) = Q_k(x) \cdot g_0(x)$, where $Q_k(x) \geq 1$ is a kth degree polynomial in x and $g_0(x)$ satisfies

$$\frac{\log g_0(x_1)}{\log g_0(x_2)} \geq \frac{x_1}{x_2}, \qquad \text{for all } x_1 \geq x_2. \tag{22}$$

Then $C_g(R)$ is bounded by

$$C_g(R) \geq C_g(R_0)\left[\frac{R}{R_0} - (k+1)\,O\left(\frac{\log n}{n}\right)\right] \tag{23}$$

where

$$R_0 \triangleq \log \sum_{x=1}^n g(x) = \log |F|, \tag{24}$$

and, from (15),

$$C_g(R_0) = \left[\sum_1^n g(x)\right]^{-1} \sum_1^n xg(x). \tag{25}$$

The proof is given in the Appendix.

The fact that $C_g(R)$ is a convex function passing through the point $(C_g(R_0), R_0)$ gives the asymptotic behavior

$$\frac{C_g(R)}{C_g(R_0)} \underset{n\to\infty}{\longrightarrow} \frac{R}{R_0} \tag{26}$$

for all values of D such that $R(D)/R_0$ is of order higher than $(\log n)/n$.

For $P(f) = 1/|F|$, we have $R(D = 0) = \log |F| = R_0$ and $C_g(R_0)$ is equal to the classical lower bound under zero distortion [10]. Under these conditions, (26) implies that the relative reduction in the mean complexity induced by tolerating a mean distortion D cannot (in the limit of large $|F|$) exceed the relative change of the rate-distortion function $R(D)$. Note that condition (22) is satisfied in many common cases. For example, the number of autonomous machines with exactly x states is given by $g(x) = x2^x$. The number of combinational circuits employing x two-input gates with p input variables and q output variables is bounded by [10] $g(x) \leq 3^x(x + p + 2)^{(2x+q)}$. The number of paths in a binary tree with depth x is given by $g(x) = 2^x$. We believe, therefore, that $R(D)$ would govern the complexity versus error tradeoffs in a variety of applications such as function realization, language recognition, and theorem proving.

V. Examples—Sorting and Modeling

We shall now apply the result of (26) to the three examples discussed in the introduction.

Example 1: Consider the realization of an arbitrary Boolean function of N variables by combinational circuits containing inverters and two-input AND and OR gates. There are $|F| = 2^{2^N}$ such computational tasks and at most $g(x) = 3^x(x+N+2)^{(2x+1)}$ circuits utilizing x such gates. Taking the number of gates as the complexity measure and the fraction of arguments for which there is disagreement as a distortion criterion and assuming all Boolean functions to be equiprobable, we have [9]

$$R(D) = 2^N[\log 2 - D \log 1/D - (1-D) \log 1/1-D], \tag{27}$$

$$R_0 = 2^N \log 2, \tag{28}$$

and the zero-distortion lower bound is given by [10]

$$C_g(R_0) = O(2^N/N), \tag{29}$$

Therefore, (26) states that

$$C(D) \geq O(2^N/N)[\log 2 - D \log 1/D - (1-D) \log 1/1-D], \tag{30}$$

implying that no amount of error tolerance below $D = 50$ percent could reduce the average combinational complexity of Boolean functions from the zero-distortion value of $O(2^N/N)$.

If, instead of the ensemble of all Boolean functions, we consider a class of partially specified Boolean functions, a lower complexity bound would result. Assume that a fraction p_1 of the 2^N input combinations are assigned an output "1", a fraction p_0 assigned an output "0" and a fraction $1 - p_0 - p_1$ remains unspecified. Taking the probability of error on specified inputs as a distortion criterion, the rate-distortion function becomes [11]

$$R(D) = 2^N(p_0+p_1)\left[H_b\left(\frac{p_1}{p_0+p_1}\right) - H_b(D)\right] \tag{31}$$

where $H_b(x)$ denotes the binary entropy function

$$H_b(x) = -x \log x - (1-x) \log (1-x). \tag{32}$$

With this lower value of $R(D)$, (26) yields

$$C(D) \geq O\left(\frac{2^N}{N}\right)(p_0+p_1)\left[H_b\left(\frac{p_1}{p_0+p_1}\right) - H_b(D)\right]. \tag{33}$$

Pippenger [12] has recently shown that (33) also represents an upper bound for approximate realizations of partially specified Boolean functions. Moreover, the zero-distortion bound in (29) can be tightened to read $C_g(R_0) = 2^N/N[1 + O(1/N)]$. This yields a more precise form of (33), substituting $2^N/N[1 + O(1/N)]$, for $O(2^N/N)$.

Example 2: Consider the construction of a minimal state sequential machine to reproduce an arbitrary binary N-sequence. There are $|F| = 2^N$ such tasks, and $g(x) = x2^x$ machines with state complexity x. Therefore, (from (24) and (25)), $R_0 = N \log 2$ and

$$C_g(R_0) = N\left[1 + O\left(\frac{\log N}{N}\right)\right].$$

Taking as a distortion criterion the mean fraction of erroneous output symbols and assuming the input sequences to be Bernoulli with parameter $q \leq 1/2$, we find that the rate-distortion function is given by [9]

$$R(D) = N[H_b(q) - H_b(D)]. \tag{34}$$

Now, using (26) and (17), we obtain

$$C(D) \geq N\left[1 + O\left(\frac{\log N}{N}\right)\right][H_b(q) - H_b(D)]. \tag{35}$$

Equation (35) implies that, for all $D < q \leq 1/2$, the average number of states needed to construct a sequential machine which reproduces a Bernoulli N-sequence with error frequency not exceeding D remains of the order of N.

Example 3: The task of sorting a list of N items using binary comparisons [3] comprises $|F| = N!$ computational tasks, each involving the execution of one correct permutation. The computation units (Ω) under consideration are sequences of binary comparisons represented by paths along a tree structure. The complexity measure $c(\omega)$ is taken to be the number of comparisons or the depth of the paths. Since there are $g(x) = 2^x$ distinct paths of depth x, the model satisfies condition (22), and so (26) is valid. Moreover, for large N, we have

$$R_0 = \log N! \simeq N \log N \tag{36}$$

and

$$C_g(R_0) \cong \log_2 (N!) \cong \frac{N \log N}{\log 2}. \tag{37}$$

Consequently, (using (26) and (17)) we obtain the asymptotic bound

$$C(D) \geq \frac{R(D)}{\log 2}. \tag{38}$$

Two distortion criteria were analyzed: the fraction of items d_s in the output sequence which are followed by wrong successors, and the fraction of pairs d_p in the output sequence which are out of order. Assuming that the input sequences are equiprobable, one can show [11], [13]

$$R(D_s) \simeq (1-D_s)N \log N, \qquad 0 \leq D_s \leq 1 \tag{39}$$

$$R(D_p) \simeq \begin{cases} N \log N, & D_p = 0 \\ N(\log 2/D_p - 2), & D_p > 0. \end{cases} \tag{40}$$

Equation (39) implies that no sorting scheme exists which matches a fixed fraction $1 - D_s > 0$ of the terms with their correct successors and which requires fewer than $O(N \log N)$ comparisons. On the other hand, (40) allows for the possibility of sorting N-sequences in linear time, if one permits any finite fraction of pairs to be out of order. Indeed, such a sorting scheme exists in the form of QUICKSORT [3] with a predetermined number of iterations. Every iteration involves comparing the items on the list to a fixed subset of randomly chosen items. For every given $D > 0$, one can determine a number $k(D)$ of iterations

necessary to produce, on the average, a fraction at most D of reversed pairs. Therefore, a D-admissible partial sorting can be accomplished by $Nk(D)$ comparisons.

VI. Discussion

We have demonstrated that, under quite general conditions, Shannon's rate-distortion function underbounds the extent to which computation complexity may be reduced by allowing some imprecision. A necessary condition for enabling the conversion of a small degree of imprecision into a drastic reduction of complexity is that the rate-distortion function exhibits a similar drastic drop. Among the examples illustrated above, only the task of sorting with pair-ordering fidelity criterion (d_p) exhibited a significant change in $R(D)/R_0$ behavior between $D_p = 0$ and $D_p > 0$. In the other three examples, the ratio $R(D)/R_0$ approached a constant function of D as $|F| \to \infty$, which forced $C(D)$ to grow at a rate similar to that of $C(0)$.

In all four examples, we chose distortion criteria which measure the fraction of errors produced within some admissible set of tests (or enquiries) on the outputs. The essential difference between sorting with pair-ordering criterion and the other three examples is that the admissible test-set characterizing the former is highly redundant. In sorting with d_s as a distortion measure, for example, one must know the successors of $N - 1$ items before the successor of the remaining item can be deduced with certainty. In pair-ordering sorting, on the other hand, knowing the correct precedence of only a small fraction of pairs (i.e., the $N - 1$ neighboring pairs) is sufficient to deduce (by transitivity) precedence in all the remaining $(N - 1)(N - 2)/2$ pairs. The fact that this redundancy increases with N was necessary (though not sufficient) to allow a sorting complexity linear with N.

A stronger redundancy in the test-set would normally result in a faster drop of $R(D)/R(0)$ for large $|F|$. For example, consider sorting with a distortion measure equal to the fraction of subsets of items whose highest member is not found at the highest relative position in the output list. The test-set in this case contains $2^N - 1$ separate tests, corresponding to the $2^N - 1$ nonempty subsets that can be chosen from the list. The results of all these tests, however, could be deduced from only $N - 1$ tests each involving a pair of neighboring items. It is not hard to show that $R(D)/R(0)$ under such distortion criteria drops at least as fast as $O(1/N)$ for $D > 0$. We could not find, however, a sorting algorithm which would guarantee a fixed D using fewer than N comparisons.

It is interesting to relate the results obtained in this paper to the Chaitin–Kolmogoroff complexity. Chaitin [14] has defined a complexity measure on sequences which formally is almost identical to Shannon's entropy. Chaitin's complexity K is defined as the length of the shortest program (with prefix property) which, when presented to a universal Turing machine, causes the machine to print the desired sequence. It is not surprising to find that this complexity measure behaves like Shannon's entropy since only the program's length, and not its execution, enters into K. Thus the essential role served by the input program is that of a *code* for identifying the desired sequence. Our paper demonstrates that information-theoretic measures also govern executional complexities, i.e., complexity measures which reflect the cost associated with the execution of a set of tasks by a given class of machines or algorithms. Since $R(D)$ can be regarded as Chaitin's program complexity under distortion D, (17) and (26) establish a connection between the latter and the executional complexity of a given computation technology characterized by g.

Appendix

Proof of Theorem 1

$C_g(R)$ is a convex $\cup$ function with $dC_g(R)/dR = -1/\mu$. It must therefore, be the upper envelope of its tangent lines. Hence, for all $\mu \leq 0$, we have

$$C_g(R) \geq C_\mu - \frac{1}{\mu}(R - R_\mu) = -\frac{1}{\mu}(R - \log \textstyle\sum g(x)e^{\mu x}). \quad \text{(A-1)}$$

Equation (A-1) remains valid if we replace $g(x)$ by another function which is everywhere greater. From (22),

$$g_0(x) \leq [g_0(n)]^{x/n} \leq \left[\sum_{y=1}^{n} g_0(y)\right]^{x/n}, \qquad \text{for } x \leq n \quad \text{(A-2)}$$

and so

$$C_g(R) \geq -\frac{1}{\mu}\left[R - \log \sum_{x=1}^{n} Q_k(x) \left[\sum_{y=1}^{n} g_0(y)\right]^{x/n} e^{\mu x}\right]. \quad \text{(A-3)}$$

Since (A-3) is valid for all $\mu \leq 0$, we may choose

$$\mu = -\frac{1}{n}\log \sum_{x=1}^{n} g_0(x) \quad \text{(A-4)}$$

and obtain

$$C_g(R) \geq \frac{n}{\log \sum g_0(x)}\left[R - \log \sum_{x=1}^{n} Q_k(x)\right]. \quad \text{(A-5)}$$

Equation (22) implies that there exists a $\lambda > 0$ (e.g., $\lambda = (1/x_1)\log g_0(x_1)$, for some fixed $1 < x_1 < n$) such that, for all n, $\log g_0(n) \geq \lambda n$. Hence

$$\frac{\log \sum_{x=1}^{n} Q_k(x)}{\log \sum_{x=1}^{n} g_0(x)} \leq \frac{\log \sum_{x=1}^{n} Q_k(x)}{\lambda n} = \frac{k+1}{\lambda} O\left(\frac{\log n}{n}\right). \quad \text{(A-6)}$$

Substitution in (A-5) establishes Theorem 1.

References

[1] M. Minsky, "Forms and content in computer science," *Journal of the Association for Computing Machinery*, vol. 17, pp. 197–215, Apr. 1970.

[2] J. E. Savage, "Computational work and time on finite machines," *Journal of the Association for Computing Machinery*, vol. 19, pp. 660–674, Oct. 1972.

[3] A. V. Aho, J. E. Hopcroft, and J. P. Ullman, *The Design and Analysis of Computer Algorithms*. Reading, MA: Addison-Wesley, 1974.

[4] M. Minsky and S. Papert, *Perceptrons*. Cambridge, MA: M.I.T. Press, ch. 12, 1969.

[5] L. A. Zadeh, "Outline of a new approach to the analysis of complex systems and decision processes," *IEEE Trans. Syst., Man, and Cybern.*, vol. SMC-3, Jan. 1973.

[6] E. H. Shortliffe and B. G. Buchanan, "A model of inexact reasoning in medicine," *Mathematical Biosciences*, vol. 23, pp. 351–374, 1975.

[7] M. O. Rabin, "Theoretical impediments to artificial intelligence,"

Proc. IFIP Congress 1974, Stockholm, Sweden, pp. 615–619, Aug. 5–10, 1974.
[8] C. E. Shannon, "Coding theorems for a discrete source with fidelity criterion," *IRE Nat. Convention Record,* Part 4, pp. 142–163, 1959.
[9] T. Berger, *Rate Distortion Theory.* Englewood Cliffs, NJ: Prentice-Hall, 1971.
[10] J. E. Savage, "The complexity of decoders," *IEEE Trans. Inform. Theory,* vol. IT-15, pp. 684–695, Nov. 1969.
[11] J. Pearl, "On the storage economy of inferential question-answering systems," *IEEE Trans. Syst., Man, and Cybern.,* vol. SMC-5, pp. 595–602, Nov. 1975.
[12] N. Pippenger, "Information theory and the complexity of switching networks," *Proc. 16th Annual Symp. on Foundations of Computer Science,* IEEE 75 CH 1003-34, Berkeley, CA, pp. 113–118, Oct. 13–15, 1975.
[13] J. Pearl, "On coding precedence relations with a pair-ordering fidelity criterion," *IEEE Trans. Inform. Theory,* vol. IT-22, pp. 118–120, Jan. 1976.
[14] A. Chaitin, "A theory of program size formally identical to information theory," IBM, Yorktown Heights, NY, *REP. RC 4805,* Apr. 1974.

Correspondence

A Distribution-Free Performance Bound in Error Estimation

LUC P. DEVROYE AND T. J. WAGNER, MEMBER, IEEE

***Abstract*—It is shown that distribution-free confidence intervals can be placed about the resubstitution estimate of the probability of error of any linear discrimination procedure.**

I. INTRODUCTION

In the discrimination problem the statistician is given an *observation* X, a random vector taking values in $\boldsymbol{R}^d$, and wishes to estimate its *state* $\theta \in \{1,2\}$. The only knowledge that the statistician has of the distribution of X, given $\theta = i$, is that which can be inferred from a sample of size n_i drawn from F_i where

$$P[X \leq x \,|\, \theta = i] = F_i(x), \qquad i = 1,2. \tag{1}$$

The two samples, here called *data,* are denoted $X_1^1, \cdots, X_{n1}^1$ and $X_1^2, \cdots, X_{n2}^2$, respectively, and are assumed to be independent of X regardless of its state.

A discrimination procedure which has been frequently investigated in the past (see, for example, Duda and Hart [1, ch. 5]) is to estimate θ by $\hat{\theta}$ where

$$\hat{\theta} = \begin{cases} 1, & \text{if } w^t X \geq w_0 \\ 2, & \text{if } w^t X < w_0. \end{cases} \tag{2}$$

The vector $w^t = (w_1, \cdots, w_d)$ and the number w_0, called the weight vector and threshold weight, respectively, are chosen from the data. Regardless of what method is used to arrive at a weight vector and threshold weight, the statistician will always be interested in estimating

$$L_i = P[\hat{\theta} \neq i \,|\, X_1^1, \cdots, X^1{}_{n1}, X_1^2, \cdots, X_{n2}^2, \theta = i], \qquad i = 1,2,$$

a random variable whose value is just the frequency of errors when a large number of independent observations, all with state i, have their states estimated using (2).

The resubstitution estimates $\hat{L}_i$ of L_i are defined by

$$\hat{L}_2 = \frac{1}{n_2} \sum_1^{n_2} I_{[w^t X_j^2 \geq w_0]}$$

and

$$\hat{L}_1 = \frac{1}{n_1} \sum_1^{n_1} I_{[w^t X_j^1 < w_0]}.$$

These estimates have the appeal of being very simple to calculate once w and w_0 have been determined and, indeed, some procedures for finding w and w_0 involve the specific calculations above. For example, for a given $0 < \alpha < 1$, one may seek values w and w_0 such that, when $\hat{L}_1 \leq \alpha$, $\hat{L}_2$ is minimized.

The question that we address ourselves to here is: how much confidence can the statistician place in these estimates, that is, for a given $\epsilon > 0$, what is

$$P[|\hat{L}_i - L_i| < \epsilon]. \tag{3}$$

There is, of course, no way of calculating (3) since the distribution functions (1) are unknown. However, if μ_i denotes the measure on the Borel sets corresponding to F_i and $\hat{\mu}_i$ denotes the empirical measure on the Borel sets for $X_1^i, \cdots, X_{n_i}^i$ (e.g., $\hat{\mu}_i(A)$ is the proportion of the X with state i falling in the set A), then

$$P[|L_i - \hat{L}_i| \geq \epsilon] \leq P\left[\sup_{A \in \mathcal{C}_i} |\mu_i(A) - \hat{\mu}_i(A)| \geq \epsilon\right] \tag{4}$$

where $\mathcal{C}_i$ deontes the class of sets of the form $\{x: w^t x \geq w_0\}$, for $i = 2$, and $\{x: w^t x < w_0\}$, for $i = 1$. The random variable on the right in (4) is, in the one-dimensional case, essentially what is dealt with in the Glivenko–Cantelli theorem [2]. Indeed, for $d \geq 1$, Wolfowitz [2] showed that this random variable tends to zero with probability one as $n_i \to \infty$. While this gives the statistician some assurance that, for large n_i, his estimate of L_i will be close to the actual value uniformly in all procedures for determining w and w_0 (see Glick [3] for a thorough discussion of this point), he still falls short of getting a numerical grasp on (3).

Suppose now that $X_1, \cdots, X_n$ is a sample of size n drawn from the distribution function F. If μ denotes the measure corresponding to F and $\hat{\mu}$ denotes the empirical measure for $X_1, \cdots, X_n$, then Vapnik and Chervonenkis [4, theorem 2, p. 269] have shown that

$$P\left\{\sup_{A \in \mathcal{C}} |\mu(A) - \hat{\mu}(A)| \geq \epsilon\right\} \leq 4s(\mathcal{C}, 2n)e^{-n\epsilon^2/8}$$

where $\mathcal{C}$ is a class of Borel sets in $\boldsymbol{R}^d$ and $S(\mathcal{C},n)$ is the maximum over $x_1, \cdots, x_n$ of the number of sets in $\{\{x_1, \cdots, x_n\} \cap A: A \in \mathcal{C}\}$. For the class of "half planes" that we are considering here (e.g., $\mathcal{C}_1$ or $\mathcal{C}_2$),

$$s(\mathcal{C}_1, n) = \sum_0^d \binom{n}{k} \leq n^d + 1, \qquad \text{if } n \geq d.$$

Manuscript received February 2, 1976. This work was supported in part by AFOSR Grant 72-2371.

The authors are with the Department of Electrical Engineering, University of Texas, Austin, TX 78712.