A Variation on the Zero-One Law

Andreas Blass*, Yuri Gurevich†, Vladik Kreinovich‡, and Luc Longpré‡

Abstract

Given a decision problem P and a probability distribution over binary strings, for each n, draw independently an instance x_n of P of length n. What is the probability that there is a polynomial time algorithm that solves all instances x_n of P? The answer is: zero or one.

Keywords: polynomial time solvability, zero-one law

At several meetings, J. Hartmanis asked: If it turns out that NP-hard problems are not solvable in polynomial time, will it mean that there is a "hard" sparse sequence of instances, i.e., a sequence which is hard for every polynomial-time algorithm? A natural next question is: how frequent are such "hard" sequences? If we pick a sequence "at random", what is the chance that this randomly chosen sequence is hard?

In principle, it could happen that almost all sequences are hard; it could happen that almost all sequences are easy; in principle, it may seem that a third alternative is also possible: that, say, half of all sequences (or any other portion different from 0 and 1) are hard, and the rest are easy. In this paper, we show that this third alternative is impossible.

To be more precise, we show that for each decision problem, either almost all *sequences* of instances are easy, or almost all sequences of instances are hard.

Consider a decision problem P on binary strings. For each positive integer n, fix a probability distribution over binary strings of length n. Let $\mathcal{X}$ be the set of infinite sequences $\bar{x} = \langle x_n : n \geq 1\rangle$ of binary strings where the length $|x_n|$ equals n. View $\bar{x}$ as a sequence of independent trials. In other words, consider the probability distribution over $\mathcal{X}$ given by the product measure [1].

Let $\mathcal{A}_P$ be the collection of polynomial time-bounded algorithms A such that, for every binary string x we have:

- if $A(x) =$ Yes, then $x \in P$, and
- if $A(x) =$ No, then $x \notin P$.

It is possible that A outputs neither Yes nor No on x; in this case A *fails* on x. We say that A *solves* the restriction $P_{|x}$ of the problem P to x if $A(x)$ is Yes or No. A *solves* a restriction $P_{|\bar{x}}$ if it solves $P_{|x_n}$ for every component x_n of $\bar{x}$. A restriction $P_{|\bar{x}}$ is *solvable* in polynomial time if there exists an $A \in \mathcal{A}_P$ that solves $P_{|\bar{x}}$. Let $p_n(A)$ be

$$p_n(A) = \text{ Prob}\,(A \text{ fails on the } n\text{th component } x_n \text{ of } \bar{x})$$

where $\bar{x}$ ranges over $\mathcal{X}$.

*Partially supported by NSF grant DMS–9505118. Mathematics Department, University of Michigan, Ann Arbor, MI 48109-1109, ablass@umich.edu

†Partially supported by NSF grant CCR 95-04375 and ONR grant N00014-94-1-1137. EECS Department, University of Michigan, Ann Arbor, MI 48109-2122, gurevich@umich.edu

‡Partially supported by NASA grant NCCW-0089, NSF grants DUE-9750858 and EEC-9322370, and AFOSR grant F49620-95-1-0518. CS Department, University of Texas at El Paso, El Paso, TX 79968, vladik@cs.utep.edu, longpre@cs.utep.edu

Theorem. *For any decision problem P, one of the following two statements holds:*

(1) *For almost all sequences $\bar{x}$, the restriction $P_{|\bar{x}}$ of the problem P to $\bar{x}$ is solvable in polynomial time, that is*

$$\text{Prob}\left(P_{|\bar{x}} \text{ is solvable in polynomial time}\right) = 1$$

(2) *For almost all sequences $\bar{x}$, the restriction $P_{|\bar{x}}$ is not solvable in polynomial time, that is*

$$\text{Prob}\left(P_{|\bar{x}} \text{ is solvable in polynomial time}\right) = 0$$

Proof. We consider two cases.

Case 1: There exists $A \in \mathcal{A}_P$ with $\sum_n p_n(A) < \infty$. Fix such an A and let $\bar{x} \in \mathcal{X}$.

Recall the first Borel-Cantelli Lemma [1, VIII.3, Lemma 1]: Let $E_1, E_2, \ldots$ be an infinite sequence of events each of which depends on a finite number of trials. If $\sum_n \text{Prob}(E_n) < \infty$, then with probability one only finitely many events E_n occur.

Let E_n be the event that A fails at the component x_n of $\bar{x}$. By the first Borel-Cantelli Lemma,

$$\text{Prob}\,(\text{there are infinitely many } n \text{ such that } A \text{ fails on } x_n) = 0$$

Hence, for almost all $\bar{x}$, A solves $P_{|x_n}$ for all but finitely many x_n. Hence, for almost all $\bar{x}$, there exists $A' \in \mathcal{A}_P$ (namely, A augmented with an appropriate finite lookup table) which solves $P_{|\bar{x}}$. Thus (1) is established.

Case 2: For every $A \in \mathcal{A}_P$, we have $\sum_n p_n(A) = \infty$. Let $\bar{x} \in \mathcal{X}$.

Recall the second Borel-Cantelli Lemma [1, VIII.3, Lemma 2]: Let $E_1, E_2, \ldots$ be as above (i.e., an infinite sequence of events each of which depends on a finite number of trials). In addition assume that these events are mutually independent. If $\sum_n \text{Prob}(E_n) = \infty$, then with probability one infinitely many events E_n occur.

Again, let E_n be the event that A fails at the component x_n of $\bar{x}$. By the second Borel-Cantelli Lemma,

$$\text{Prob}\,(\text{there are infinitely many } n \text{ such that } A \text{ fails on } x_n) = 1.$$

Hence, for every A, $\text{Prob}\,(A \text{ solves } P_{|\bar{x}}) = 0$. Since there are only countably many algorithms, $\text{Prob}\,((\exists A)(A \text{ solves } P_{|\bar{x}})) = 0$. Thus, (2) is established. □

The proof gives a little more.

Corollary. *For any decision problem P:*

1. *If conclusion* (1) *of the theorem holds, then there exists a polynomial time-bounded decision algorithm $A \in \mathcal{A}_P$ and there exists a sequence of reals $\varepsilon_n > 0$ with $\sum_n \varepsilon_n < \infty$ such that $p_n(A) \le \varepsilon_n$.*

2. *If conclusion* (2) *of the theorem holds, then, with probability* 1, *for a random sequence $\bar{x}$, the restriction $P_{|\bar{x}}$ of P to $\bar{x}$ is not solvable in polynomial time.*

Actually, the theorem and the corollary are more general. Instead of polynomial time algorithms, one can use any type of partition of binary strings into two parts (fail, solve) provided that (i) the type is closed under finite changes and (ii) there are only countably many partitions of that type.

For example, this proof also holds for an *alternative* definition of a solving algorithm, where $A(x) =$ Yes does not necessarily imply $x \in P$, and $A(x) =$ No may not imply $x \notin P$. For such algorithms, we say that A *solves* $P_{|x}$ if $A(x)$ coincides with the truth value of $x \in P$; if it does not, we say that A *fails* on x. As above, we say that A *solves* the restriction $P_{|\bar{x}}$ of P to the sequence $\bar{x}$ if it solves $P_{|x_n}$ for every component x_n of $\bar{x}$.

Acknowledgments. The authors are thankful to the anonymous referees for comments.

References

[1] William Feller, "An Introduction to Probability Theory and Its Applications", Third Edition, John Wiley and Sons, 1968.