Guessing with Lies

Erdal Arikan[1]
Electrical-Electronics Engineering Dept.
Bilkent University, 06533 Ankara, Turkey
e-mail: arikan@ee.bilkent.edu.tr

Serdar Boztaş
Dept. of Mathematics, RMIT University
Melbourne 3001, Australia
e-mail: serdar@rmit.edu.au

I. Introduction

The *noiseless* case ([1, 3]) of the guessing problem is when a sequence of questions of the form *Is* $X = x$*?* are posed until a YES answer determines the correct value of a random variable X with range $\mathcal{X} = \{x_1, x_2, \ldots\}$ and distribution P_X (see also [2, 4] for extensions). Here, we assume there is a nonzero probability that the NO answer received is not the right answer, while the YES answer is noiseless.

Let L denote the number of lies (erroneous NO answers) encountered during the course of the search. L may depend on X but is independent of the algorithm employed to find X. Let $P_{X,L}(x,\ell)$ be the joint distribution of (X, L). A *guessing strategy* for identifying X is any sequence $g_1, g_2, \ldots$ of elements from $\mathcal{X}$–g_i will be the ith probe if all previous probes have yielded NO answers. An *optimal* guessing strategy is one which minimizes the average number of guesses

$$\mathbf{E}[G] = \sum_{\ell=0}^{\infty} \sum_{x \in \mathcal{X}} P_{X,L}(x,\ell) G(x,\ell) \tag{1}$$

where $G(x,\ell)$ is the time index of the $(\ell+1)$th probe of x. Clearly, the *guessing functions* G satisfy the *precedence constraints* $G(x,k+1) > G(x,k)$ for all x and $k \geq 0$. It can be shown that this problem is equivalent to a noiseless guessing problem on the (X, L) space with no false answers but subject to precedence constraints. Such a problem is very difficult to solve explicitly. We obtain (i) a practical algorithm for directly generating an optimal guessing sequence for guessing X under lies L; (ii) information-theoretic bounds on the average number of guesses for optimal strategies.

II. The Optimal Guessing Algorithm

Our algorithm generates an optimal guessing sequence one probe at a time. At any point, each element $x \in \mathcal{X}$ will have been probed k_x times. The *state vector* $(k_x : x \in \mathcal{X})$ indicates that the algorithm has probed the set of points $\{(x,\ell) : x \in \mathcal{X},\ 0 \leq \ell < k_x\}$ and received NO answers. Given the current state $(k_x : x \in \mathcal{X})$, the next probe has to be chosen from the *available set* $\{(x, k_x); x \in \mathcal{X}\}$.

If $P(x,\ell)$ is nonincreasing in ℓ for any fixed x, a simple greedy algorithm that probes the element (x, k_x) in the available set for which $P(x, k_x)$ is largest is optimal. Otherwise, the simple greedy algorithm may fail to be optimal. The optimal algorithm in the general case uses a different metric to prioritize its search. Define for $\ell_2 \geq \ell_1 \geq 0$

$$A_x(\ell_1,\ell_2) = \frac{1}{\ell_2 - \ell_1 + 1} \sum_{\ell=\ell_1}^{\ell_2} P(x,\ell), \quad \text{and}$$

$$A_x(\ell_1) = \max_{\ell \geq \ell_1} A_x(\ell_1,\ell)$$

We call an algorithm a *greedy-A* algorithm if it chooses its next probe from the available set so as to maximize the quantity $A_x(k_x)$.

Theorem 1 *Any guessing sequence generated in accordance with the greedy-A algorithm is optimal, i.e., it attains the minimum possible average number of guesses.*

Greedy-A algorithms typically generate their guesses in batches; i.e., they probe the same element successively a number of times before moving on to another element. This property is used to bound the expectation $\mathbf{E}[G]$:

Theorem 2 *Let G be any optimal guessing function for guessing X in the presence of lies L. Then, the average number of guesses for G is upperbounded by*

$$\mathbf{E}[G] \leq 1 + \exp[H_{1/2}(Q)] \tag{2}$$

and lowerbounded by

$$\mathbf{E}[G] \geq (1 + \ln|\mathcal{X}|)^{-1} \exp[H_{1/2}(Q)] \tag{3}$$

where Q is a distribution derived from the batches of G, and $H_{1/2}(Q) = \ln(\sum_i \sqrt{Q_i})^2$ is the Rényi entropy of order $\frac{1}{2}$.

Remark: Assume $P_{X,L}(x,\ell)$ is nonincreasing in $\ell \geq 0$ for each fixed $x \in \mathcal{X}$; e.g., a geometric distribution with an x–dependent parameter. Then, each batch has size 1, and the bounds of Theorem 2 are valid with $P_{X,L}$ in place of Q. The Rényi entropy of order 1/2 satisfies $H_{1/2}(P_{X,L}) = H_{1/2}(P_{L|X}) + H_{1/2}(P_X)$, where the *conditional* Rényi entropy is defined as $H_{1/2}(P_{L|X}) = \ln \sum_x \left[\sum_\ell \sqrt{P_{X,L}(x,\ell)}\right]^2$. In this case, the guessing effort can be thought of as consisting of two parts, one directed at X, the other at L given X.

We also note that the A-greedy algorithm can be modified to efficiently solve the general noiseless guessing problem with precedence constraints. This problem can, in principle, be solved by Markov decision theory and dynamic programming at the cost of exponential complexity in search domain size.

[1] E. Arikan was visiting the RMIT Mathematics Department, supported by an RMIT Faculty of Applied Science grant, when this work was in part performed.

References

[1] E. Arikan, An Inequality on Guessing and Its Application to Sequential Decoding, *IEEE Trans. Inform. Theory,* Vol. IT-42, pp. 99-105, 1996.

[2] E. Arikan and N. Merhav, Joint Source-channel Coding and Guessing with Application to Sequential Decoding, *IEEE Trans. Inform. Theory,* Vol. IT-44, pp. 1756-1769, 1998.

[3] S. Boztaş, Comments on 'An Inequality on Guessing and Its Application to Sequential Decoding', *IEEE Trans. Inform. Theory,* Vol. 43, No. 6, pp. 2062-2063, November 1997.

[4] N. Merhav and E. Arikan, The Shannon Cipher System with a Guessing Wiretapper, *IEEE Trans. Inform. Theory,* Vol. IT-45, pp. 1860-1866, 1999.

0-7803-7501-7/02/$17.00©2002 IEEE.