Strategy for Data Transmission over Binary Channels with Noiseless Feedback and Upper Bound on the Number of Questions in Searching with Lies

Vladimir B. Balakirsky
Electr. Eng. Dept., TUE
P.O. 513, 5600 MB Eindhoven
The Netherlands

Abstract — **A transmission strategy that allows the sender to deliver any of M messages to the receiver over a binary channel when at most e errors can occur is presented. The total number of bits required by the strategy differs from the known lower bound by $3e$. This statement simultaneously gives a new upper bound on the number of questions in the process of searching with lies known as the "Ulam's game".**

I. Introduction and formulation of the result

Let $[M] = \{1, \ldots, M\}$ denote the set of messages. One of them should be transmitted over a binary channel where at most e errors $0 \to 1$ and $1 \to 0$ can occur. For all $\tau = 1, 2, \ldots$, the sender noiselessly observes the received bit y_τ and sends the next bit $x_{\tau+1}$ based on the message $m \in [M]$ and the bits $x_1, \ldots, x_\tau, y_1, \ldots, y_\tau$ which were transmitted and received at the previous time instants. Any transmission strategy is the algorithm for computing the bits $x_1, \ldots, x_n$ by the sender and for decoding the message m by the receiver, where n is the total number of transmitted bits. The equivalent problem can be formulated as searching for an integer $m \in [M]$ by asking questions when at most e lies are allowed in the answers : the question $\mathcal{Q}$ is a subset of the set $[M]$; the answer is either 0 or 1, and the answers 0 if $m \in \mathcal{Q}$ and 1 if $m \notin \mathcal{Q}$ are considered as lies.

Theorem : *There exist searching strategies such that any of M integers can be discovered with $n^* = n + 3e$ questions when at most e lies are allowed in the answers, where n is the minimal integer satisfying the inequality $MV_e^{(n)} \le 2^n$ and $V_e^{(n)} = \sum_{i=0}^{e} \binom{n}{i}$.*

II. Basic ideas of the proof

Suppose that $\mathcal{M}_0, \ldots, \mathcal{M}_e$ are pairwise disjoint subsets of the set $[M]$ constructed by the questioner in such a way that if $m \in \mathcal{M}_j$, then j lies are allowed in all further answers, $j = 0, \ldots, e$. Then the vector $\mathbf{c} = (|\mathcal{M}_0|, \ldots, |\mathcal{M}_e|)$ can be interpreted as the state of the search. Let $\mathcal{D}(c)$ denote the set consisting of integers δ such that $|\delta| \le c$ and $\delta \equiv c \pmod 2$. For all $\boldsymbol{\delta} = (\delta_0 \in \mathcal{D}(c_0), \ldots, \delta_e \in \mathcal{D}(c_e))$, let $\mathbf{a}(\mathbf{c}|\boldsymbol{\delta}) = (a_0, \ldots, a_e)$, $\mathbf{b}(\mathbf{c}|\boldsymbol{\delta}) = (b_0, \ldots, b_e)$, where $a_j = (c_j + c_{j+1} + \delta_j - \delta_{j+1})/2$, $b_j = (c_j + c_{j+1} - \delta_j + \delta_{j+1})/2$ and $j = 0, \ldots, e$ (we assume that $c_{e+1} = \delta_{e+1} = 0$). Let the vector $\boldsymbol{\delta}$ specify the question $\bigcup_{j=0}^{e} \mathcal{Q}_j$, where $\mathcal{Q}_j$ is the subset consisting of $(c_j + \delta_j)/2$ smallest elements of the subset $\mathcal{M}_j$ for all $j = 0, \ldots, e$. We will call this procedure "partitioning of the subsets $\mathcal{M}_0, \ldots, \mathcal{M}_e$ in accordance with the vector $\boldsymbol{\delta}$". Let $\mathcal{Q}_{e+1} = \mathcal{M}_{e+1} = \emptyset$ and $\overline{\mathcal{Q}}_j = \mathcal{M}_j \backslash \mathcal{Q}_j$ for all $j = 0, \ldots, e+1$. If the answer is 1, the new vector of sets has components $\mathcal{M}'_j = \mathcal{Q}_j \bigcap \overline{\mathcal{Q}}_{j+1}$ and $\mathbf{a}(\mathbf{c}|\boldsymbol{\delta})$ is the new state of the search. If the answer is 0, the new vector of sets has components $\mathcal{M}''_j = \overline{\mathcal{Q}}_j \bigcap \mathcal{Q}_{j+1}$ and $\mathbf{b}(\mathbf{c}|\boldsymbol{\delta})$ is the new state of the search.

A key point of our considerations is the introduction of a special class of rooted regular binary trees whose nodes "contain" possible states of the search. This construction relates binary trees to coverings of the Hamming spaces by sets having the cardinalities coincident with the sizes of the Hamming balls and leads to the following statement, which is essentially used in the searching strategy. *Let $\mathcal{M}_0, \ldots, \mathcal{M}_e$ be current pairwise disjoint subsets constructed by the questioner and let t be the minimal integer satisfying the inequality $\sum_{j=0}^{e} |\mathcal{M}_j| \cdot V_j^{(t)} \le 2^t$. Then the searching problem under consideration cannot be solved using less than t questions, and a solution with t questions exists only if this problem can be solved using t questions when some set $\hat{\mathcal{M}}_0$ assigned in such a way that $|\hat{\mathcal{M}}_0| = 2^t - \sum_{j=1}^{e} |\mathcal{M}_j| \cdot V_j^{(t)}$; $\hat{\mathcal{M}}_0 \supseteq \mathcal{M}_0$; $\hat{\mathcal{M}}_0 \bigcap \mathcal{M}_1 = \ldots = \hat{\mathcal{M}}_0 \bigcap \mathcal{M}_e = \emptyset$ is substituted for $\mathcal{M}_0$.*

To prove the theorem we present a specific algorithm for assigning the vector $\boldsymbol{\delta}$ for any vector $\mathbf{c}$ that can be obtained by the questioner and denote this vector by $\boldsymbol{\delta}^*(\mathbf{c})$. Let η denote the index of the last positive component of the vector $\mathbf{c}$. If $c_\eta = 1$, then $\delta_1^* := -c_1, \ldots, \delta_{\eta-1}^* := -c_{\eta-1}$; $\delta_\eta^* := 1$; $\delta_{\eta+1}^* := \ldots := \delta_e^* := 0$. If $c_\eta > 1$, then

$$\delta_j^* := \arg \min_{\delta \in \mathcal{D}(c_j)} \left| \binom{t-1}{j} \delta + \sum_{i=j+1}^{e} \binom{t-1}{i} \delta_i^* \right|$$

for $j = e, \ldots, 1$. In both cases, $\delta_0^* := -\sum_{j=1}^{e} \binom{t-1}{j} \delta_j^*$.

The searching strategy is given below, where the vector of length $e+1$ containing 1 in the j–th position and 0's in all other positions is denoted by $\mathbf{1}_j$, $j = 0, \ldots, e$.

1. $c_0 := 2^n - MV_e^{(n)}$; $\mathcal{M}_0 := \ldots := \mathcal{M}_{e-1} := \emptyset$; $\mathcal{M}_e := [M]$.
2. $\mathbf{c} := (c_0, |\mathcal{M}_1|, \ldots, |\mathcal{M}_e|)$. If $\mathbf{c} \in \{\mathbf{1}_0, \ldots, \mathbf{1}_e\}$, then go to 5.
3. Construct the vector $\boldsymbol{\delta}^*(\mathbf{c}) = (\delta_0^*, \ldots, \delta_e^*)$ using the algorithm described above with the value of t determined by the equation $\sum_{j=0}^{e} c_j V_j^{(t)} = 2^t$. If $|\delta_0^*| > c_0$, then $c_0 := c_0 + \sum_{j=0}^{e} c_j \binom{t}{j}$ and go to 2.
4. Partition the subsets $\mathcal{M}_0, \ldots, \mathcal{M}_e$ in accordance with the vector $\boldsymbol{\delta}^*(\mathbf{c})$. If the answer is 1, then $c_0 := (c_0 + c_1 + \delta_0^* - \delta_1^*)/2$ and $\mathcal{M}_j := \mathcal{M}'_j$, $j = 0, \ldots, e$. If the answer is 0, then $c_0 := (c_0 + c_1 - \delta_0^* + \delta_1^*)/2$ and $\mathcal{M}_j := \mathcal{M}''_j$, $j = 0, \ldots, e$. Go to 2.
5. End : the singleton $\mathcal{M}_j$, where j is such that $c_j = 1$, contains the defined integer.