ALBANIAN JOURNAL
OF MATHEMATICS
Volume 2, Number 3, Pages 179–184
ISSN 1930-1235: (2008)

ON IDENTIFICATION FOR SOURCES EXTENDED TO MODEL WITH LIES

Zlatko Varbanov

Department of Mathematics and Informatics
Veliko Tarnovo University, 5000 Veliko Tarnovo
e-mail:vtgold@yahoo.com

1. Introduction

The classical transmission problem deals with the question how many possible messages can we transmit over a noisy channel? Transmission means there is an answer to the question "What is the actual message?"

In the identification problem we deal with the question how many possible messages the receiver of a noisy channel can identify? Identification means there is an answer to the question "Is the actual message u?". Here u can be any member of the set of possible messages.

Allowing randomized encoding the optimal code size grows double exponentially in the block length and somewhat surprisingly the second order capacity equals Shannon's first order transmission capacity (see [5]).

Thus, Shannon's Channel Coding Theorem for Transmission is paralleled by a Channel Coding Theorem for Identification. It seems natural to look for such a parallel for sources, in particular for noiseless coding. This was suggested by Ahlswede in [1].

Let $(\mathcal{U}, P)$ be a source, where $\mathcal{U} = \{1, 2, \ldots, N\}, P = \{P_1, P_2, \ldots, P_N\}$, and let $\mathcal{C} = \{c_1, c_2, \ldots, c_N\}$ be a binary prefix code (PC) for this source with $||c_u||$ as length of c_u. Introduce the random variable U with $\text{Prob}(U = u) = p_u$ for $u = 1, 2, \ldots, N$ and the random variable C with $C = c_u = (c_1, c_2, \ldots, c_{u||c_u||})$ if $U = u$.

We use the PC for noiseless identification, that is user u wants to know whether the source output equals u, that is, whether C equals c_u or not. The user iteratively checks whether C coincides with c_u in the first, second, etc. letter and stops when the first different letter occurs or when $C = c_u$. The problem is: **What is the expected number $L_{\mathcal{C}}(P, u)$ of checkings?**

In order to calculate this quantity we introduce for the binary tree $T_{\mathcal{C}}$, whose leaves are the codewords $c_1, c_2, \ldots, c_N$, the sets of leaves $\mathcal{C}_{ik} (1 \leq i \leq N; 1 \leq k)$, where $\mathcal{C}_{ik} = \{c \in \mathcal{C} : c$ coincides with c_i exactly until the k'th letter of $c_i\}$. If C takes a value in $\mathcal{C}_{uk}, 0 \leq k \leq ||c_u|| - 1$, the answers are k times "Yes" and 1 time "No". For $C = c_u$ we have

$$L_{\mathcal{C}}(P, u) = \sum_{k=0}^{||c_u||-1} P(C \in \mathcal{C}_{uk})(k + 1) + ||c_u|| P_u.$$

Partially supported by RD491-09/2008 project, Veliko Tarnovo University.

©2008 Aulona Press *(Albanian J. Math.)*

For a code $\mathcal{C}$, the number $L_{\mathcal{C}}(P) = \max_{1\geq u\geq N} L_{\mathcal{C}}(P,u)$ is the expected number of checkings in the worst case and $L(P) = \min_{\mathcal{C}} L_{\mathcal{C}}(P)$ is this number for the best code.

2. Uniformly distributed sources

2.1. **Construction of a prefix code.** Let $P^N = \{\frac{1}{N},\dots,\frac{1}{N}\}$. We construct a prefix code $\mathcal{C}$ in the following way. In each node (starting at the root) we split the number of remaining codewords in proportion as close as possible to $(\frac{1}{2},\frac{1}{2})$. It is known [3] that for such code $\mathcal{C}$

$$\lim_{N\to\infty} L_{\mathcal{C}}(P^N) = 2 \tag{1}$$

Example 1. *Let* $N = 9,\ \mathcal{U} = \{1,2,\dots,9\},\ P_1 = \cdots = P_9 = \frac{1}{9}$. *Then,*

$$\mathcal{C} = \{000, 001, 010, 011, 100, 101, 110, 1110, 1111\}$$

$$L_{\mathcal{C}}(P) = L_{\mathcal{C}}(P,c_8) = \frac{4}{9}.1 + \frac{2}{9}.2 + \frac{1}{9}.3 + \frac{1}{9}.4 + \frac{1}{9}.4 = \frac{19}{9} \approx 2,111$$

$$L_{\mathcal{C}}(P,c_9) = L_{\mathcal{C}}(P,c_8); L_{\mathcal{C}}(P,c_7) = \frac{17}{9}; L_{\mathcal{C}}(P,c_5) = L_{\mathcal{C}}(P,c_6) = \frac{16}{9};$$

$$L_{\mathcal{C}}(P,c_1) = L_{\mathcal{C}}(P,c_2) = L_{\mathcal{C}}(P,c_3) = L_{\mathcal{C}}(P,c_4) = \frac{15}{9}$$

In [2] was stated the problem to estimate an universal constant $A = \sup L(P)$ for general distribution $P = (P_1,\dots,P_N)$. Here, we compute such constant for uniform distribution and this code $\mathcal{C}$.

Using decomposition formula for subtrees, we obtain the following recursion

$$L_{\mathcal{C}_N}(P^N) = \frac{\lceil \frac{N}{2} \rceil}{N} L_{\mathcal{C}_{\lceil \frac{N}{2} \rceil}}(P^{\lceil \frac{N}{2} \rceil}) + 1\ ,\ L_{\mathcal{C}_2}(P^2) = 1 \tag{2}$$

where $\mathcal{C}_t$ is the corresponding code with t codewords.

From (2) follows that the worst case for $L_{\mathcal{C}}(P^N)$ is when $N = 2^k + 1$, for any integer k. We compute the exact value for $L_{\mathcal{C}}(P^N)$ in this case.

Theorem 1. $\sup_N L_{\mathcal{C}}(P^N) = 2 + \frac{log_2(N-1)-2}{N}$

Proof. If $N = 2^k + 1$ then 2^k codewords are in level k (the root is level 0) in the binary tree $T_{\mathcal{C}}$ and one codeword is in level $k+1$ (if this codeword is w then $L_{\mathcal{C}}(P^N, w) = L_{\mathcal{C}}(P^N)$). For every node in level i $(0 \leq i \leq k-1)$ we split 2^{k-i-1} codewords in the left side and $2^{k-i-1}+1$ codewords in the right side. Therefore, $P(C \in \mathcal{C}_{wi}) = \frac{2^{k-i-1}}{2^k+1},\ i = 0,\dots,k-1$. Then, for $L_{\mathcal{C}}(P^N)$ we obtain

$$\begin{aligned} L_{\mathcal{C}}(P^N) = L_{\mathcal{C}}(P^N, w) &= \sum_{i=0}^{k} P(C \in \mathcal{C}_{wi})(i+1) + ||c_w||P_w \\ &= \sum_{i=0}^{k-1} P(C \in \mathcal{C}_{wi})(i+1) + P(C \in \mathcal{C}_{wk})(k+1) + ||c_w||P_w \\ &= \sum_{i=0}^{k-1} \frac{2^{k-i-1}}{2^k+1}(i+1) + \frac{1}{2^k+1}(k+1) + \frac{k+1}{2^k+1} \\ &= \frac{2^k}{2^k+1} \sum_{i=0}^{k-1} \frac{i+1}{2^{i+1}} + \frac{2(k+1)}{2^k+1} \end{aligned}$$

$$= \frac{2^k}{2^k+1}.\frac{2^{k+1}-k-2}{2^k} + \frac{2k+2}{2^k+1} = \frac{2^{k+1}-k-2}{2^k+1} + \frac{2k+2}{2^k+1}$$
$$= \frac{2^{k+1}+2+k-2}{2^k+1} = 2 + \frac{k-2}{2^k+1}$$

But $N = 2^k+1$ and $k = log_2(N-1)$. Then we obtain $L_{\mathcal{C}}(P^N) = 2+\frac{log_2(N-1)-2}{N}$. □

2.2. **Average identification length.** Also, in our work we consider the case where not only the source outputs but the users occur at random. In addition to the source $(\mathcal{U}, P)$ and random variable U, we are given $(\mathcal{V}, Q), \mathcal{V} \equiv \mathcal{U}$ with random variable V independent of U and defined by $\text{Prob}(V = v) = Q_v$ for $v \in V$. The source encoder knows the value u of U but not that of V, which chooses the user v with probability Q_v. Again let $\mathcal{C} = \{c_1, \ldots, c_N\}$ be a binary prefix code and let $L_{\mathcal{C}}(P, u)$ be the expected number of checkings on code $\mathcal{C}$ for user u.

Instead of $L_{\mathcal{C}}(P) = \max_{u\in\mathcal{U}} L_{\mathcal{C}}(P, u)$ we can consider the average number of expected checkings (also called *average identification length*):

$$L_{\mathcal{C}}(P, Q) = \sum_{v\in\mathcal{V}} Q_v L_{\mathcal{C}}(P, v);\ L(P, Q) = \min_{\mathcal{C}} L_{\mathcal{C}}(P, Q)$$

A special case is $Q = P$, where

$$L_{\mathcal{C}}(P, P) = \sum_{u\in\mathcal{U}} P_u L_{\mathcal{C}}(P, u); \quad L(P, P) = \min_{\mathcal{C}} L_{\mathcal{C}}(P, P)$$

and for uniform distribution we have $L_{\mathcal{C}}(P^N, P^N) = \frac{1}{N}\sum_{u\in\mathcal{U}} L_{\mathcal{C}}(P^N, u)$.

2.3. **Results.** We calculate exact values of $L_{\mathcal{C}}(P^N)$ and $L_{\mathcal{C}}(P^N, P^N)$ for some N and summarize them in Table 1. We know [3] that for $N = 2^k, L_{\mathcal{C}}(P^N) = L_{\mathcal{C}}(P^N, P^N) = 2 - \frac{2}{N}$.

TABLE 1 - some exact values for uniform distribution, $2^k < N < 2^{k+1}, k \geq 3$

N	$L_{\mathcal{C}}(P^N)$	$L_{\mathcal{C}}(P^N, P^N)$
$2^k + 1$	$2 + \frac{log_2(N-1)-2}{N}$	$2 + \frac{log_2(N-1)-2}{N^2}$
$2^k + 2^{k-1} - 1$	2	$2 - \frac{5(N+1)-3log_2(\frac{2N+2}{3})}{3N^2}$
$2^k + 2^{k-1}$	$2 - \frac{1}{N}$	$2 - \frac{5}{3N}$
$2^k + 2^{k-1} + 1$	$2 + \frac{log_2(\frac{N-1}{12})}{N}$	$2 - \frac{(5N-2)-3log_2(\frac{N-1}{12})}{3N^2}$
$2^{k+1} - 1$	$2 - \frac{1}{N}$	$2 - \frac{2N-log_2(N+1)+1}{N^2}$

3. Extension to liar models

3.1. **Identification and lies.** Suppose that when user u iteratively checks whether C coincides with c_u in the first, second, etc. letter, for some reasons he obtains wrong information in any position. Then, there is a lie(error) in this position of the codeword. In this model with lies (we follow the idea in [4] but here no different costs of the lies), the user knows only that the general number of lies is at most e and no information for the positions of lies.

Let $L_{\mathcal{C}}(P,u) = L_{\mathcal{C}}(P)$ for any $u \in \mathcal{U}$. In this case, we denote by $L_{\mathcal{C}}(P;e)$ the expected number of checkings if there are at most e lies. Now, main question is: **What is the expected number of checkings if there are at most e lies?**

We can see that the user needs of $e+1$ the same answers ("Yes" or "No") to be sure for the correct answer in any position. If the user has done $2e+1$ questions for any position he gets exact information for the value in this position. Therefore, there exists trivial upper bound

$$L_{\mathcal{C}}(P;e) \leq (2e+1)L_{\mathcal{C}}(P) \tag{3}$$

Clearly, this bound (3) can be improved by decreasing the number of remaining lies. The algorithm described below can be used.

3.2. **An Algorithm.** To decrease the number of remaining lies the following algorithm can be used for any $u \in \mathcal{U}$:

Step 0: BEGIN $i := 1, Checkings := 0$, actual message $:= v$;

Step 1: If $i > ||c_v||$ then Step 3. Otherwise, check codeword position i until $e+1$ the same answers. Let t be the number of obtained answers "Yes" and f be the number of obtained answers "No";

Step 2: $Checkings := Checkings + (t+f)$. If $t > f$, then $e := e - f$, $i := i+1$, Step 1. Otherwise, the actual message $v \neq u$;

Step 3: END.

By this algorithm, we obtain the following result.

Lemma 2. *Let v be the current checked codeword and let i be the first position in which c_u and c_v differ (if $c_u = c_v$ then $i = ||c_u||$). Then, the number of checkings in the worst case is $e(i+1)+i$.*

Proof. We can see that the worst case with respect by e is when all lies(errors) occur in position i. In this case

$Checkings = (e+1)(i-1) + (2e+1).1 = e(i+1) + i$.

If there is even one lie in any position m $(1 \leq m \leq i-1)$, for every position j $(m+1 \leq j \leq i)$ the user needs of e the same answers. Then

$Checkings = (m-1)(e+1) + (e+2) + (i-m-1)e + (2e-1) = e(i+1) + m < e(i+1) + i$.

Therefore, this number $e(i+1)+i$ is the maximal number of checkings if this algorithm is used. □

Example 2. *Let $N = 9$, $\mathcal{U} = \{1, 2, \ldots, 9\}$, $P_1 = \cdots = P_9 = \frac{1}{9}$, and $e = 3$*

Then

$$\mathcal{C} = \{000, 001, 010, 011, 100, 101, 110, 1110, 1111\},$$

and

$$\begin{aligned} L_{\mathcal{C}}(P, c_8) &= L_{\mathcal{C}}(P, c_9) = L_{\mathcal{C}}(P) \\ L_{\mathcal{C}}(P; 3) &\le \frac{4}{9}.7 + \frac{2}{9}.(4+7) + \frac{1}{9}.(4+4+7) \\ &\quad + \frac{1}{9}.(4+4+4+7) + \frac{1}{9}.(4+4+4+7) = \frac{103}{9} \end{aligned}$$

3.3. **Results for liar models.** Using Lemma 2, we prove our main result.

Theorem 3. $L_{\mathcal{C}}(P; e) \le (e+1)L_{\mathcal{C}}(P) + e$

Proof. Let $k = ||c_u||$ and $P_{ui} = P(C \in \mathcal{C}_{ui})$. Then, in the worst case we obtain the following

$$\begin{aligned} L_{\mathcal{C}}(P; e) &\le \sum_{i=0}^{k-1} P_{ui}(e(i+2) + i + 1) + (e(k+1) + k)P_u \\ &= e\sum_{i=0}^{k-1} P_{ui}(i+2) + e(k+1)P_u + \sum_{i=0}^{k-1} P_{ui}(i+1) + kP_u \\ &= e\sum_{i=0}^{k-1}(P_{ui}(i+1) + P_{ui}) + e(k+1)P_u + L_{\mathcal{C}}(P) \\ &= e(\sum_{i=0}^{k-1} P_{ui}(i+1) + kP_u) + e(\sum_{i=0}^{k-1} P_{ui} + P_u) + L_{\mathcal{C}}(P) \\ &= eL_{\mathcal{C}}(P) + e.1 + L_{\mathcal{C}}(P) = \underline{(e+1)L_{\mathcal{C}}(P) + e}. \end{aligned}$$

□

Let $M_{\mathcal{C}}(P; e) = (e+1)L_{\mathcal{C}}(P) + e$. Then we have;

Corollary 4. *For uniform distribution* P^N

$$\lim_{N\to\infty} M_{\mathcal{C}}(P^N; e) = 3e + 2$$

Proof. Follows from (1) and Theorem 3. □

Let consider other distribution P when all individual probabilities are powers of $\frac{1}{2}$, $P_u = \frac{1}{2^{\ell_u}},\ u \in \mathcal{U} = \{1, 2, \ldots, N\}$. Since

$$\sum_{u\in\mathcal{U}} \frac{1}{2^{\ell_u}} = 1$$

by Kraft's theorem there is a prefix code $\mathcal{C}$ with codeword lengths $||c_u|| = \ell_u$.

For such code $\mathcal{C}$ we know [2] that $L_{\mathcal{C}}(P, u) = 2(1-P_u)$. Therefore, $\lim_{N\to\infty} L_{\mathcal{C}}(P) = 2$ and by Theorem 3 we obtain the same result for this distribution P.

Corollary 5. $\lim_{N\to\infty} M_{\mathcal{C}}(P; e) = 3e + 2$

Also, for general distribution $P = (P_1, P_2, \ldots, P_N)$ we know that $L(P) \le 3$ ([3], Theorem 3). Therefore, for $L(P; e)$ (the expected number of checkings for the best code $\mathcal{C}$ and at most e lies) we have

Corollary 6. $L(P; e) \le 4e + 3$

References

[1] R.Ahlswede, General theory of information transfer: updated (Original version: General theory of information transfer, Preprint 97-118, SFB 343 "Diskrete Strukturen in der Mathematik", Universität Bielefeld), General Theory of Information Transfer and Combinatorics, a Special issue of Discrete Applied Mathematics.

[2] R. Ahlswede, "Identification entropy", General Theory of Information Transfer and Combinatorics, Lecture Notes in Computer Science, Vol. 4123, Springer Verlag, 595–613, 2006.

[3] R. Ahlswede, B. Balkenhol, and C. Kleinewächter, "Identification for sources", General Theory of Information Transfer and Combinatorics, Lecture Notes in Computer Science, Vol. 4123, Springer Verlag, 51–61, 2006.

[4] R. Ahlswede, F. Cicalese, and C. Deppe, Searching with lies under error transition cost constraints, General Theory of Information Transfer and Combinatorics, Special Issue of Discrete Applied Mathematics, to appear.

[5] R.Ahlswede, G.Dueck, "Identification via channels", IEEE Trans. Inf. Theory, Vol.35, No.1, 15–29, 1989.