INDUCTIVE INFERENCE OF FUNCTIONS BY PROBABILISTIC STRATEGIES

Karlis PODNIEKS
Institute of Mathematics and Computer Science, University of Latvia
Karlis.Podnieks@mii.lu.lv

Abstract. The following model of inductive inference is considered. Arbitrary numbering tau = {tau_0, tau_1, tau_2, ...} of total functions N->N is fixed. A "black box" outputs the values f(0), f(1), ..., f(m), ... of some function f from the numbering tau. Processing these values by some algorithm (a strategy) F we try to identify a tau-index of f (i.e. a number n such that f = tau_n). Strategy F outputs an infinite sequence of hypotheses h_0, h_1, ..., h_m, ... If lim h_m = n and tau_n = f, we say that F identifies in the limit tau-index of f. The complexity of identification is measured by the number of mind-changes, i.e. by $F^{tau}(f)$ = card{m | h_m <> h_{m+1} }. One can verify easily that for any numbering tau there exists a **deterministic** strategy F such that $F^{tau}(tau_n)$ <= n for all n. This estimate is exact (see [Ba 74], [FBP 91]). In the current paper the corresponding exact estimate ln n + o(log n) is proved for **probabilistic** strategies. This result was published without proof in [Po 75]. Proofs were published in [Po 77-2].

Translation of papers [Po 75] and [Po 77-2] published in Russian.
For the results (without proofs) in English see also [FBP 91].

1. Definitions and Results

Probabilistic strategy is defined by some:

a) Probability space (W, B, P), where W is the set of elementary events, B - a Borel field over subsets of W, P - a probability measure over B;

b) mapping M: N*->Z, where N* is the set of all finite sequences of natural numbers, Z - the set of all random variables over the space (W, B, P) taking their values from N U {oo} (oo means "undefined").

Thus M associates with each elementary event e in W some deterministic strategy F_e. The hypothesis F_e (<f(0),...,f(m)>) takes its values n with fixed probabilities P{F_e(<f(0),...,f(m)>)=n}.

Computable (recursive) probabilistic strategies can be defined by means of probabilistic Turing machines introduced first in [LMS 56]. Let a random Bernoulli generator of the distribution (1/2, 1/2) be fixed. The generator is switched into deterministic "apparatus" of a Turing machine. As a result, the operation of the machine becomes probabilistic, and we can speak of the probability that the operation satisfies certain conditions.

Consider the following Turing machine M operating with a fixed Bernoulli generator. With input sequence

f(0), f(1), ..., f(m), ...,

this machine prints as output an empty, finite or infinite sequence of natural numbers (hypotheses):

$h_0, h_1, ..., h_m, ...,$

where h_m depends only on the values f(0), ..., f(m). To each infinite realization of Bernoulli generator's output (i.e. an infinite sequence of 0's and 1's) there corresponds a completely determined operation of the machine M as a deterministic strategy.

By P{M, tau, f} we denote the probability that a probabilistic strategy M identifies in the limit tau-index of the function f. By P{M, f, <=k} we denote the probability that strategy M makes no more than k mind-changes by the function f.

Let us denote by $f^{[m]}$ the code of <f(0), ..., f(m)>, then the random variable M(<f(0), ..., f(m)>) can be denoted by $M(f^{[m]})$. By $P_m(M, f)$ we denote the probability that M changes its hypothesis at the step m, i.e. $P\{M(f^{[m+1]})<>M(f^{[m]})\}$.

First, let us consider some sufficient condition for P{M, tau, f}=1. We will say that strategy M is tau-**consistent** on the function f if, for all m,

a) $M(f^{[m]})$ is always defined (i.e. defined for all events e in W),

b) if $M(f^{[m]})=n$ for some event e in W, then $tau_n(j)=f(j)$ for all j<=m.

Thus, consistent strategies do not output "explicitly incorrect" hypotheses.

THEOREM 1. For any enumerated class (U, tau) there is a probabilistic strategy M and a constant C>0 such that: a) M always identifies in the limit tau-index of each function f in U, and b) M changes its mind by the function tau_n no more than ln n+C*sqrt(ln n)*lnln n times with probability ->1 as n->oo. For a computable numbering tau, a computable probabilistic strategy M can be constructed..

THEOREM 2. For any countable set FI of probabilistic strategies there exists an enumerated class (U, tau) and a constant C>0 such that: for any strategy M in FI, which identifies in the limit tau-index of each function f in U, there is an increasing sequence $\{n_k\}$ such that M changes its mind by the function tau_{nk} at least ln n - C*sqrt(ln n)*lnln n times with probability ->1 as k->oo. For the class of all computable probabilistic strategies, a computable numbering tau can be constructed.

The upper bound ln n can be proved by means of a strategy from [BF 72]. Essential difficulties arise, however, not in the construction of the strategy but in its analysis.

The main idea is as follows. Let us suppose that the function f in the "black box" which outputs the values f(0), f(1), f(2), ..., is chosen at random from the numbering tau, according to some probability distribution pi={pi_n} (pi_n is the probability that f=tau_n, pi_1+pi_2+pi_3+... =1). Then, having received the values f(0), ..., f(m), let us consider the set E_m of all tau-indices "suitable" for such f, i.e.

$$E_m = \{n \mid (Aj<=m)\ tau_n(j) = f(j)\}.$$

It would be natural to output as a hypothesis any tau-index n in E_m with probability pi_n / s, where s = Sum{pi_n |n in E_m}.

To put it precisely, we define a probabilistic strategy $BF_{tau,pi}$ as follows. Let tau be any numbering of total functions. Take some probability distribution {pi_n}, where pi_n>0 for all n, and pi_1+pi_2+pi_3+... =1.

If the set E_0 = {n | tau_n(0)=f(0)} is empty, then we put $BF_{tau,pi}(f^{[0]})$ undefined with probability 1. If E_0 is non-empty, we put $BF_{tau,pi}(f^{[0]})$=n with probability pi_n/s for every n in E_0, where s = Sum_n{pi_n | n in E }.

Let us assume now that the hypotheses $BF_{tau,pi}(f^{[j]})$ have already been determined for j<m, and $BF_{tau,pi}(f^{[m-1]})$=p. If p is "undefined", then we set $BF_{tau,pi}(f^{[m]})$ undefined with probability 1. Else, if tau_p(m)=f(m) (i.e. the hypothesis p is correct also for the next argument m), we set $BF_{tau,pi}(f^{[m]})$=p with probability 1.

Suppose now that tau_p(m)<>f(m). Let us take the set of all appropriate (for the present) hypotheses, i.e.

$$E_m = \{n \mid (Aj<=m)\ tau_n(j) = f(j)\}.$$

If E_m is empty, then we put $BF_{tau,pi}(f^{[m]})$ undefined with probability 1. If E_m is nonempty, we put $BF_{tau,pi}(f^{[m]})$=n with probability p_n/s for every n in E_m, where s = Sum_n{ pi_n | n in E_m }.

Of course, $BF_{tau,pi}$ always identifies in the limit tau-index of each function f in U.

2. Proofs

LEMMA 1. Let {X_m} be a series of independent random variables taking values from {0,1} in such a way that: a) $Sum_m X_m$ is always finite, b) E = Sum_mP{X_m=1}is finite. Then for any number t>0:

$$P\{ |Sum_m X_m - E|>=t*sqrt(E) \} <= 1/t^2.$$

PROOF. Immediately, by Chebishev inequality.

Lemma 1 allows deriving upper and lower bounds of the number of mindchanges from the corresponding bounds of Sum_m P_m(M, f).

LEMMA 2. For all n,

$$Sum_m\{ P_m(BF_{tau,pi}, tau_n) \} <= ln(1/pi_n).$$

LEMMA 3. Let a function f of the numbering tau be fixed. Then the following events are independent:

$$A_m = \{ BF_{tau,pi}(f^{[m]}) <> BF_{tau,pi}(f^{[m+1]}) \}, m = 0, 1, 2, ...$$

It is curious that the events A_m (i.e. "at the m-th step $BF_{tau,pi}$ changes its mind") do not display any striking indications of independence; nevertheless, they do satisfy the formal independence criterion.

If we take

$$pi'_n = c / (n^* (ln n)^2),$$

with the convention that 1/0=1 and ln 0 = 1, then by Lemma 2 the sum of the probabilities of $BF_{tau,pi}$ changing hypothesis by the function tau_n will be less than ln n + 2lnln n - ln c. Lemma 3 and Lemma 1 (let us take t = lnln n) allow us to derive from this that, as n->oo, with probability ->1, the number of mindchanges of $BF_{tau,pi'}$ by tau_n does not exceed ln n + C*sqrt(ln n)*lnln n, thus proving Theorem 1. For proofs of Lemmas 2, 3 see Section 3.

It is easy to verify that if the numbering tau is computable, then the strategy $BF_{tau,pi'}$ can be modified so that it becomes computable (see Section 3).

The lower bound ln n of Theorem 2 is proved by diagonalization. Let $\{M_i\}$ be some numbering of strategies of the set FI. In the numbering tau to be constructed, for each strategy M_i an infinite sequence of blocks B_{ij} is built in, each block consisting of a finite number of functions tau_n. If M_i identifies (with probability 1) tau-indices of all functions of B_{ij}, then by some function of this block M_i will change its mind sufficiently often.

To construct an individual block B_{ij} the following Lemma 4 will be used.

Let $\{Z_j\}$ be a sequence of independent random variables taking values 0,1 in such a way that

$$P\{Z_j = 1\} = 1/j, P\{Z_j = 0\}=1- 1/j.$$

Thus,

$$P\{Z_2=1\}+ ... +P\{Z_n=1\}= 1/2+...+1/n = ln n+O(1).$$

Let us take t = lnln n in Lemma 1, then for some C>0, as n->oo,

$$P\{ Z_2+...+Z_n >= ln n - Csqrt(ln n)lnln n) \}->1.$$

Now one can understand easily the significance of

LEMMA 4. Let M be a probabilistic strategy, k and n - natural numbers with $k<n$, $e>0$ - a rational number, gamma - a binary string. Then there is a set of n functions s_1, ..., s_n such that: a) each function s_j has gamma as initial segment, and b) if M identifies with probability 1 s-indices of all functions s_j, then by one of these functions M changes its mind >=k times with probability

$$>=(1-e)\ P\{\ Z_2+...+Z_n >=k\ \}.$$

If M is recursive strategy, then the set s_1, ..., s_n can be constructed effectively.

Now, to prove Theorem 2, we derive the block B_{ij} from the set of functions of Lemma 4 for

$$M = M_i,\ n = 2^j,\ k =]j*\ln 2 - sqrt(j)\ln j[,\ e = 2^{-j}\ ,\ gamma = 0...01$$

with $<i, j>$ zeros in gamma, where $<i, j>$ is Cantor's number of the pair (i, j).

Let us introduce a special coding of triples (i, j, s) such that $0<=s<=2^j$ (see [BF 74]). The code of (i, j, s) is defined as the binary number

```
100....0a_t...a_0100....0, ----------(*)
|------ j ----|--|-- i --|--------------
```

where $a_t...a_0$ is the number s. Clearly,

$$2^{i+j+1} + 2^i <= code(i, j, s) <= 2^{i+j+2} - 2^i.$$

Now the numbering tau can be defined as follows. If n is (*) for some (i, j, s), $0<=s<=2^j$, then tau_n is the s-th function of the block B_{ij}. Else tau_n is set equal to zero.

Let a probabilistic strategy M in FI identify with probability 1 tau-indices of all functions of the numbering tau. Then $M=M_i$ for some i, and for each $j>1$ the block B_{ij} contains a function by which M changes its mind at least]j*ln 2 - sqrt(j) ln j[times with probability

$$>= (1-2^j)\ P\{\ Z_2+...+Z_n >=]j*\ln 2 - sqrt(j)\ \ln j[\ \}\ \text{-------------(**)}$$

Let us denote by n_j the (unique) tau-index of this "bad" function. Obviously,

$$2^{i+j+1} < n_j < 2^{i+j+2}.$$

Hence, $\{n_j\}$ is an increasing sequence, and

$$\log_2 n_j -i-2 < j < \log_2 n_j -i-1.$$

By the function tau_{nj} the strategy M changes its mind at least

$$j*\ln 2-sqrt(j)\ \ln j > \ln n_j - Csqrt(\ln n_j)\ln\ln n_j$$

times with probability (**), which ->1 as j->oo, thus proving Theorem 2.

For a proof of Lemma 4 see Section 4 and Section 5.

3. Proofs of Lemmas 2, 3

LEMMA 5. Let L be an empty or finite set of natural numbers, set K(L) = the set of all tuples (i_1, ..., i_s) such that ($i_1<i_2<...<i_s$) & (Ak<=s) i_k in L (the empty tuple included). Then for an arbitrary sequence of reals $\{x_n\}$ we have

$$Sum_i\{ (x_{i1}-1)...(x_{is}-1) \mid i \text{ in } K(L) \}= Prod_j\{ x_j \mid j \text{ in } L \},$$

where Sum_i ranges over all tuples i in K(L).

PROOF. Immediately, by induction.

PROOF OF LEMMA 3. Let us consider the tree of functions of the numbering tau which coincide up to some moment with the function f:

----> b_0...----> b_1...----> b_2..----> b_3......................----> b_m....................................
|..............|..............|.............|.................................|...
|..............|..............|.............|.................................|...
---> a ....(f)
S_0...........S_1...........S_2...........S_3...............................S_m..

The infinite path drawn here corresponds to the function f (which may have more than one tau-index, by the way). The outgoing arrows correspond to functions tau_n declining from f. With each vertice of the tree we associate the probability that a function tau_n chosen according to the distribution pi meets this vertice. Let b_m be the probability that tau_n meets the vertice S_m and immediately after that declines from f. Let

$$B_m = b_m + b_{m+1} +...$$

Then the probability assigned to S_m will be $a+B_m$, where a is the probability assigned to the infinite path f (i.e. a = $Sum_n\{pi_n \mid tau_n=f\}$).

By b_{ij} (i>=0, j>=i) we denote the probability that in the case of changing its mind at the vertice S_i the istrategy $BF_{tau,pi}$ directs its new hypothesis through S_j aside from f. Clearly,

$$b_{ij} = b_j/(a+B_i). \text{ ----------------(*)}$$

The total probability of changing the mind at S_m, m>0, can be expressed by the numbers b_{ij}:

$$P\{A_m\} = Sum\{ b_{0,i1}*b_{i1+1,i2}*...*b_{ik+1,m} \},$$

where Sum ranges over all tuples (i_1, ..., i_k) such that k>=0, $0<=i_1< i_2 < ... < i_k < m$.

The probability of simultaneous mindchanges at S_{m1}, ..., S_{mt} (where $m_1 < m_2 < ... < m_t$) can be expressed similarly:

$$P\{A_{m1} \wedge .. .\wedge A_{mt}\} = Sum\{ b_{0,i1}*b_{i1+1,i2}*...*b_{ik+1,mt} \},$$

where Sum ranges over all tuples (i_1, ..., i_k) such that k>=0, $0<=i_1< i_2 < ... < i_k < m_t$ and {m_1, ..., m_{t-1}}is a subset of {i_1, ..., i_k}.

By (*), the probability $P\{A_{m1} \wedge .. .\wedge A_{mt}\}$ depends only on a, b_0, ..., b_m and B_{m+1}, where $m=m_t$ =max m_i. Let us introduce new variables g_i, 0<=i<=m+1:

$a+B_{m+1} = ag_{m+1}$,

$a+B_m = ag_mg_{m+1}$, $b_m = a(g_m-1)g_{m+1,}$

$a+B_{m-1} = ag_{m-1}g_mg_{m+1}$, $b_{m-1} = a(g_{m-1}-1)g_mg_{m+1}$,

$a+B_j = ag_jg_{j+1} ... g_{m+1,}$ $b_j = a(g_j-1)g_{j+1} ... g_{m+1}$ (j=0,1,...,m).

Then we will have

$$b_{ij} = b_j/(a+B_i) = (g_j-1)/(g_i ... g_j),$$

$$b_{0,i1}*b_{i1+1,i2}*...*b_{ik+1,m} = (g_{i1}-1)...(g_{ik}-1)(g_m-1) / (g_0 ... g_m),$$

$$P\{A_{m1} \wedge .. .\wedge A_{mt}\} = (g_{m1}-1)...(g_{mt}-1) Sum\{ (g_{i1}-1)...(g_{ik}-1) \}/ (g_0 ... g_m),$$

where L = {0, 1, ..., m}-{m_1, ..., m_t}, and Sum ranges over all tuples (i_1, ..., i_k) in K(L), where K(L) is defined in Lemma 5. By Lemma 5, the latter Sum is equal to Prod{ g_i | i in L }, hence,

$$P\{A_{m1} \wedge .. .\wedge A_{mt}\} = (g_{m1}-1)...(g_{mt}-1) Prod\{ g_i | i in L \} / (g_0 ... g_m) = (g_{m1}-1)/g_{m1} * ... *(g_{mt}-1)/g_{mt}.$$

For t=1 we would have:

$$P\{A_m\} = P_m(BF_{tau,pi}, f) = (g_m-1)/g_m = b_m/(a+B_m).$$

Hence,

$$P\{A_{m1} \wedge .. .\wedge A_{mt}\} = P\{A_{m1}\}* ... *P\{A_{mt}\},$$

i.e. events A_i are independent. This proves Lemma 3.

PROOF OF LEMMA 2. As we already know,

$$P_m(BF_{tau,pi}, f) = b_m/(a+B_m) = 1 - (a+B_{m+1})/(a+B_m).$$

Summing up for all m, and using the inequality 1-x <= ln(1/x) we obtain that

$$Sum_m\{ b_m/(a+B_m) \} <= \ln Prod_m\{ a+B_{m+1})/(a+B_m) \}.$$

Since

$$Prod_{m<=s}\{ a+B_{m+1})/(a+B_m) \} = (a+B_0)/(a+B_{s+1}) -> (a+B_0)/a, \text{ as } s->oo,$$

we obtain that

$$P_m(BF_{tau,pi}, f) <= \ln((a+B_0)/a).$$

If $f=tau_n$, then $a>=pi_n$. Clearly, $a+B_0<=1$, hence,

$$\ln((a+B_0)/a) <= \ln (1/pi_n).$$

Q.E.D.

To complete the proof of Theorem 1 we show now, for a computable numbering tau and computable probability distribution pi ($pi_1+pi_2+pi_3+...=1$), how the recursive counterpart $BF'_{tau,pi}$ of the strategy $BF_{tau,pi}$ can be constructed. We will use also a omputable sequence of rationals $\{e_m\}$ such that $Prod_m (1+e_m) < oo$ (for example, $e_m=2^{-m}$).

Let us modify the definition of the hypothesis $BF_{tau,pi}(f^{[m]})$ (see Section 2) as follows. If the numbering tau is computable, the set E_m is recursive, hence, we can compute a binary-rational probability distribution ($lambda_{n1}$, $lambda_{n2}$, ..., $lambda_{nk}$) which e_m-approximates the distribution $\{pi_n/s \mid n \text{ in } E_m\}$, $s = Sum_n\{pi_n \mid n \text{ in } E_m\}$, i.e. $lambda_{n1}+ lambda_{n2}+...+ lambda_{nk} = 1$, and for all i:

$$n_i \text{ in } E_m \ \& \ lambda_{ni} <= (1+e_m) pi_{ni} / s$$

Now define $BF'_{tau,pi}(f^{[m]}) = n_i$ with probability $lambda_{ni}$ for all i = 1, ..., k.

LEMMA 6. Let $BF'_{tau,pi}$ be the modified computable probabilistic strategy. Then for all n and k,

$$P\{BF'_{tau,pi}, tau_n, >=k\} <= P\{BF_{tau,pi}, tau_n, >=k\}* Prod_m(1+e_m).$$

PROOF. Let us return to the proof of Lemma 3. For the probabilities b'_{ij} of $BF'_{tau,pi}$ (corresponding to b_{ij} of $BF_{tau,pi}$ in Section 2) we have:

$$b'_{ij} <= (1+e_i) b_{ij}. \text{-------------(1)}$$

The probability $P\{BF'_{tau,pi}, f, >=k\}$ can be expressed by b'_{ij}:

$$P\{BF'_{tau,pi}, f, >=k\} = Sum\{ b'_{0,i1}*b'_{i1+1,i2}* ... *b'_{ik-1+1,ik} \},$$

where Sum ranges over all tuples (i_1, ..., i_k) such that $k>=0$, $0<= i_1 < i_2 < ... < i_k$. Hence, by (1),

$$P\{BF'_{tau,pi}, f, >=k\} <= Prod_m(1+e_m) * Sum\{ b'_{0,i1}*b'_{i1+1,i2}* ... *b'_{ik-1+1,ik} \}$$
$$<= P\{BF_{tau,pi}, f, >=k\}* Prod_m(1+e_m).$$

By Lemma 6, since the inequality of Theorem 1 holds for the strategy $BF_{tau,pi}$, it holds also for $BF'_{tau.pi}$.

4. Proof of Lemma 4

Let us carry out the (more complicated) "computable case" of the proof. Let M be a computable probabilistic strategy, n, k - natural numbers, k<n, e>0 - a rational number, gamma = $g_0g_1...g_a$ - a binary string. We will construct n functions s_1, s_2, ..., s_n starting with the values from gamma, such that if M identifies with probability 1 s-indices of all functions s_i, then by one of these functions M will change its mind >=k times with probability

$$>= (1-e)\ P\{ Z_2+...+Z_n >=k\},$$

where Z_i are random variables defined in Section2.

Let us consider the idea of the proof for the case n=4, k=2. The generalization is straightforward.

Procedure P_M. We initiate parallel computing of probabilities of the following events:

$$M(b_0)=t_0\ \&\ M(b_0b_1)=t_1\ \&\ ...\ \&\ M(b_0...b_m)=t_m, \text{-----------------}(1)$$

for all binary strings beta=$b_0b_1...b_m$ and all finite sequences t=$t_0t_1..t_m$ of natural numbers. This can be done as follows. For all pairs (alpha, beta) of binary strings alpha, beta the following parallel -computation process is carried out: alpha serves as a finite realization of Bernoulli generator's output (i.e. a finite sequence of 0's and 1's), and beta - as initial segment f(0), ..., f(m) of some function f taking values 0,1. Initially, we associate with each pair (beta, t) an empty set of binary strings alpha. When, during the computation process with alpha and beta, we see the sequence s printed on the output tape of M, then we add alpha to the set associated with (beta, t). If it appears that alpha is too short for some computation, we simply drop this computation. And so on.

End of procedure P_M.

Simultaneously with P_M, we add new values to functions s_1, s_2, s_3, s_4:

$$s_i(0)=g_0, s_i(1)=g_1, ..., s_i(a)=g_a, s_i(a+1)=0, s_i(a+2)=0,...$$

(where gamma = $g_0g_1...g_a$). Only at some particular moments we will interfere into this process, and add a finite number of 1's as values of s_i.

The first of these moments will appear, when the probabilities of (1) will be computed precisely enough to obtain for some number j_1 the following approximate probability distribution of the hypothesis M(gamma + 0^{j1}):

| 1.......2.......3........4 | ....(2)
|q_1.......q_2.......q_3........q_4 |.........................

where $q_1+q_2+q_3+q_4$ =1 and for all i:

$$q_i \le P\{M(\text{gamma} + 0^{j1}) = i\}/(1\text{-delta}),$$

where delta>0 is a constant such that $(1\text{-delta})^3 >= 1\text{-e}$ (here we have n-1=3, see Section 5).

If such a moment does not appear, it would mean, that for each j in N the hypothesis M(gamma+0^j) is undefined or not in {1, 2, 3, 4} with probability greater than delta. Then, with probability >delta, this is the case for infinitely many j's simultaneously, i.e. by the function gamma + 0^{oo} the strategy M outputs infinitely many hypotheses other than 1, 2, 3, 4. But this is exactly the case, when we do not interfere the definition process of the functions s_1, s_2, s_3, s_4, and hence, they will be all equal to gamma + 0^{oo}. For this case, Lemma 4 holds obviously.

Now let us consider the case, when the probability distribution (2) can be obtained. Using (2) and the algorithm described in Section 5, we exclude one of the numbers 1, 2, 3, 4 in the following sense (for example, let it be the number 1):

The function s_1, instead of the current value $s_1(x_1)=0$, obtains the value $s_1(x_1)=1$, and for all $x>x_1$ s_1 is set equal to 0. The remaining three functions s_2, s_3, s_4 obtain for $x=x_1$ the value 0 (i.e. other than $s_1(x_1)$), and then (after our "interference" is concluded) they continue to obtain zero values. I.e., after this moment, s_1 differs from s_2, s_3, s_4, and by these last 3 functions the hypothesis 1 will be wrong. Our algorithm (see Section 5) guarantees that 1 will be removed only if $q_1>0$.

The definition process of s_2, s_3, s_4 will be interfered for the second time, when the probabilities of (1) will be computed precisely enough to obtain for some $j_2 > j_1$ the numbers q_{1i}:

| 2........3........4 | ....(3)
|q_{12}.....q_{13}.....q_{14} |.........................

such that $q_{12}+q_{13}+q_{14}=1$ and for all i:

$$q_{1i} \le P\{ M(\text{gamma} + 0^{j1})=1 \ \& \ M(\text{gamma} + 0^{j2}) =2 \}/ (1\text{-delta})^2.$$

If such a moment would not appear, it would mean, that for some delta'>0 and all j > j_1:

$$P\{ M(\text{gamma} + 0^{j}) \text{ not in } \{2,3,4\} \mid M(\text{gamma} + 0^{j1}) = 1 \} > \text{delta}'.$$

Hence, with probability >delta' this is the case for infinitely many j's simultaneously. Since our second interference does not take place in this case, the functions s_2, s_3, s_4 will be set equal to gamma + 0^{oo}. Lemma 4 holds in this case obviously.

Let us consider the case, when the distribution (3) can be obtained. Then, by the algorithm of Section 5, we exclude another function s_i (for example, let it be s_2). We set $s_2(x_2)=1$, $s_3(x_2)=s_4(x_2)=0$, and for all $x>x_2$: $s_2(x)=0$ (here, of course, $x_2>x_1$, where x_1 is the location of our first interference).

The third interference (and the last one - when n=4) in the definition process of functions s_3, s_4 will take place, when the probabilities of (1) will be computed precisely enough to obtain a number $j_3 > j_2$ and numbers q_{12i}, q_{22i}:

| 3.......... 4 |.....| 3...........4 |......................(4)
| q_{123}...q_{124} |.....| q_{223}...q_{224} |..........................

such that $q_{123}+q_{124}=1$, $q_{223}+q_{224}=1$, and for i=3, 4:

$$q_{12i} <= P\{ M(gamma + 0^{j1})=1 \ \& \ M(gamma + 0^{j2}) =2 \ \& \ M(gamma + 0^{j3}) =i \}/ (1-delta)^3.$$

$$q_{22i} <= P\{ M(gamma + 0^{j1})=2 \ \& \ M(gamma + 0^{j2})=i \}/ (1-delta)^3.$$

In the case, when the numbers j_3, q_{12i}, q_{22i} cannot be obtained, Lemma 4 holds obviously.

If the numbers (4) have been obtained, the algorithm of Section 5 "removes" another function s_i (for example, let it be s_3). We set $s_3(x_3)=1$ (where, of course, $x_3>x_2$), $s_4(x_3)=0$, and for all $x>x_3$: $s_4(x)=0$. Since n=4, now only one function s_4 remains, let $s_4(x)=0$ for all $x>x_3$.

Hereby we conclude the definition of functions s_1, ..., s_n, corresponding by Lemma 4 to the probabilistic strategy M, natural numbers k, n (k<n) and rational number e>0. The algorithm, described in Section 5, will guarantee that by one of the functions s_i the strategy M will change its mind >=k times with a sufficiently large probability.

5. Exclusion Algorithm

We consider the case n=4, k=2. The generalization is straightforward.

If the strategy M identifies with probability 1 s-indices of all functions s_1, s_2, s_3, s_4, then during the construction process of these functions all the three possible interferences must have been performed. For example, in the following sequence 123(4):

<table>
<tr><td>j_1</td><td colspan="4">1</td><td colspan="2">2</td><td>3</td><td>4</td></tr>
<tr><td>j_2</td><td colspan="2">2</td><td>3</td><td>4</td><td colspan="2">2</td><td>3</td><td>4</td></tr>
<tr><td>j_3</td><td>3</td><td>4</td><td>3</td><td>4</td><td>3</td><td>4</td><td>3</td><td>4</td></tr>
<tr><td>j_4</td><td>4</td><td>4</td><td>4</td><td>4</td><td>4</td><td>4</td><td>4</td><td>4</td></tr>
</table>

Segments split the first row according to the probability distribution (q_1, q_2, q_3, q_4) of Section 4. The first 3 segments of the second row split q_1 according to the distribution (q_{12}, q_{13}, q_{14}). The segments of the third row split q_{12} to (q_{123}, q_{1124}), and q_{22} - to (q_{223}, q_{224}). In the last row we have (with probability 1) the hypothesis 4 - the only "right" s-index of s_4.

Let us introduce a more convenient notation:

$$|1|=q_1,\ |2|=|22|=q_{12},$$

$$|12|=q_{12},\ |133|=q_{13},$$

$$|123|=q_{123},\ |224|=q_{224},...$$

Let us assume (for a moment) that these numbers coincide exactly with the probabilistic characteristics of the strategy M (see Section 4), for example:

$$|123| = P\{M(gamma + 0^{j1})=1\ \&\ M(gamma + 0^{j2})=2\ \&\ M(gamma + 0^{j3})=3\},$$

$$|223| = P\{M(gamma + 0^{j1})=2\ \&\ M(gamma + 0^{j3})=3\}.$$

Then, the probability that by the function s_4 the strategy M will change its mind at least two times (k=2), is

$$>= |123|+|124|+|133|+|223| = |12|+|13|+|223|.\ \text{-----------------(a)}$$

When during the last interference the function s_4 would have been excluded (instead of s_3), then instead of (a) we would have had the sum

$$|12|+|14|+|224|.\ \text{--------------(aa)}$$

Hence, when the functions s_1, s_2 have been already excluded, it would be sensible to exclude next s_3 or s_4 depending on the maximum - (a) or (aa). And this can be really decided - all the items of (a) and (aa) are known at the level j_3.

Now let us consider the level j_2. The function s_1 is already excluded. We know the probabilities |1|, |2|, |3|, |4|, |12|, |13|, |14|. What of the functions should be excluded next - s_2, s_3 or s_4? The decision "exclude s_2" can be estimated by means of the sum (a)+(aa), i.e.,

$$|12|+|13|+|223|+|12|+|14|+|224| = |1|+|22|+|12| = |1|+|2|+|12|.\ \text{-----------(b)}$$

Hence, this sum depends only on the probabilities known at the level j_2. The corresponding sums for s_3 and s_4 are

$$|1|+|3|+|13|, \text{------------(bb)}$$

$$|1|+|4|+|14|. \text{------------(bbb)}$$

Hence, the maximum of (b), (bb) and (bbb) should decide, which of the functions s_2, s_3, s_4 should be excluded next, after s_1. And this can be really decided at the level j_2.

Finally, let us consider the level j_1. The decision "exclude s_1" can be estimated by means of the sum (b)+(bb)+(bbb):

$$3*|1|+|2|+|3|+|4|+|12|+|13|+|14| = 1+3*|1|.$$

The analogous estimates for s_2, s_3, s_4 are

$$1+3*|2|, 1+3*|3|, 1+3*|4|.$$

All these estimates can be computed at the level j. Hence, at this level we should exclude the function s_i having the maximum 1+3*|i|.

Hereby, for n=4, k=2 the definition of our exclusion algorithm is concluded. The generalization is straightforward.

In Section 6 we will prove generally, that all the summing-up operations used in the algorithm are leading to the level, at which the required decision should be performed.

Now let us obtain some estimate of the probability that the strategy M will change its mind >=2 times by the function s_4.

At the level j_1 we had 4 estimates:

$$1+3*|1|, 1+3*|2|, 1+3*|3|, 1+3*|4|.$$

The sum of them is 4+3*1=7, i.e. it depends only on n and k. If, according to our algorithm, we exclude s_1, then:

$$1+3*|1| >= 7 / 4.$$

If, after this, s_2 is excluded, then:

$$|1|+|2|+|12| >= 7 / (4*3).$$

Finally, the exclusion of s_3 means that

$$|12|+|23|+|223| >= 7 / (4*3*2), \text{--------------(1)}$$

i.e. the probability of M changing its mind >=2 times is >= 7 / (4*3*2).

However, this conclusion would be absolutely valid only if the characteristic probabilities of M would be exactly |1|, |12|, |223| etc. This is not the case, but one can select the number delta of Section 4 so that instead of the estimate >= 7 / (4*3*2) we obtain >= (1-e)* 7 / (4*3*2), where e is the third parameter of Lemma 4. Indeed,

$$P\{M, s_4, >=2\}>= P\{M(gamma + 0^{j1})=1 \ \& \ M(gamma + 0^{j2})=2\} +$$

$$+ P\{M(gamma + 0^{j1})=1 \ \& \ M(gamma + 0^{j2})=3\}+$$

$$+ P\{M(gamma + 0^{j1})=2 \ \& \ M(gamma + 0^{j3})=3\}>=$$

$$>= (1\text{-}delta)^2 q_{12} + (1\text{-}delta)^2 q_{13} + (1\text{-}delta)^3 q_{223} >=$$

$$>= (1\text{-}delta)^3 (|12|+|13|+|223|) >= (1\text{-}delta)^3 * 7 / (4*3*2).$$

When delta is selected so that $(1\text{-}delta)^3 >= 1\text{-}e$, then

$$P\{M, s_4, >=2\} >= (1\text{-}e)*7/(4*3*2).$$

To conclude the proof of Lemma 4 (for n=4 and k=2), it remains to verify that

$$7/(4*3*2) = P\{ Z_2+Z_3+Z_4 >= 2 \}, \text{----------------}(2)$$

where Z_i are independent random variables such that

$$P\{Z_i=1\} = 1/i, P\{Z_i=0\}=1-1/i.$$

Of course, one could verify (2) directly:

$$(1/2)*(1/3)*(1-1/4) + (1/2)*(1-1/3)*(1/4) + (1-1/2)*(1/3)*(1/4) + (1/2)*(1/3)*(1/4) = 7/(4*3*2).$$

To obtain a general method (for arbitrary n,k, $k<n$), we can use constructions from the proof of Lemma 3 (see Section 4).

First let us note that the lower bound 7/(4*3*2) of (1) is reached if and only if the table at the beginning of this section is "symmetric": the first row is splitted in ratio 1:4, the second one - in 1:3, the third one - in 1:2. Such a table is simulating the work of the strategy $BF_{tau,pi}$ by the function $s_4 = 0^{oo}$, where

$$tau = (s_1, s_2, s_3, s_4),$$

$$s_1 = 010^{oo}, s_2 = 0010^{oo}, s_3 = 00010^{oo}, s_4 = 0^{oo},$$

$$pi = \{1/4, 1/4, 1/4, 1/4\}.$$

Indeed, the hypothesis $BF_{tau,pi}(<0>)$ = 1, 2, 3 or 4 with equal probabilities 1/4. Further, if $BF_{tau,pi}(<0>)$ = 1, then $BF_{tau,pi}(<0,0>)$ = 2, 3 or 4 with probabilities 1/3. If $BF_{tau,pi}(<0>)=1$ & $BF_{tau,pi}(<0,0>)=2$, then $BF_{tau,pi}(<0,0,0>)$ = 3 or 4 with probabilities 1/2. If $BF_{tau,pi}(<0>)$ = 2, then $BF_{tau,pi}(<0,0>)$ = 2 with probability 1, and $BF_{tau,pi}(<0,0,0>)$ = 3 or 4 with probabilities 1/2. Finally, the hypothesis $BF_{tau,pi}(<0,0,0,0>)=4$ with probability 1.

By using the function tree from the proof of Lemma 3:

```
   ----> 010^oo......----> 0010^oo....----> 00010^oo..............................................................
 |..b0..................|..b1...................|..b2..................................................................................
 |........................|.......................|..........................................................................................
      --------------------------------------------------------------------------> a .........(0^oo)
S0.....................S1.....................S2.........................................................................................
```

where $b_0 = b_1 = b_2 = a = 1/4$, we obtain:

$$P\{BF_{tau,pi}(<0>)<>BF_{tau,pi}(<0,0>)\} = b_0 / (b_0+b_1+b_2+a) = 1/4.$$

$$P\{BF_{tau,pi}(<0,0>)<>BF_{tau,pi}(<0,0,0>)\} = b_1 / (b_1+b_2+a) = 1/3.$$

$$P\{BF_{tau,pi}(<0,0,0>)<>BF_{tau,pi}(<0,0,0,0>) = b_2 / (b_2+a) = 1/2.$$

By Lemma 3, the following X_i (i=0, 1, 2) are independent random variables:

$$X_i = 1, \text{ if } BF_{tau,pi}(<0^i>) <> BF_{tau,pi}(<0^{i+1}>),$$

$$X_i = 0, \text{ otherwise.}$$

Hence, the triple (X_0, X_1, X_2) coincides with the triple (Z_4, Z_3, Z_2). Since the sum $X_0+X_1+X_2 = Z_4+Z_3+Z_2$ represents the number of mindchanges of M by the function 0^{oo}, we obtain that

$$7/(4*3*2) = P\{ Z_2+Z_3+Z_4 >= 2 \}.$$

Obviously, this way (2) can be proved for arbitrary n,k (k<n).

6. Proof of "Tail Reduction"

The general setting is as follows. Let n be a natural number. Some probability distribution over the set {1,2,...,n} is labeled by the empty tuple **o**:

$$|1|_{\mathbf{o}}, |2|_{l\mathbf{o}}, ..., |n|_{\mathbf{o}}.$$

When now some number i in {1,2,...,n} is "excluded", the probability $|i|_{\mathbf{o}}$ is splitted in the following way:

$$|i|_{\mathbf{o}} = Sum_y\{ |iy|_{\mathbf{i}} \mid y<>i \},$$

where the new distribution is labeled by the tuple (i). The probabilities $|x|_{\mathbf{o}}$ for x<>i are retained:

$$|x|_{\mathbf{o}} = |xx|_{\mathbf{i}}.$$

Let us call the 2-tuples, obtained in these two ways (i.e. (iy) for $y<>i$, and (xx) for $x<>i$) **(i)-admissible**.

In the next step another number j in {1,2,...,n}-{i} is "excluded", and the probabilities $|ij|_{\mathbf{i}}$, $|jj|_{\mathbf{i}}$ are splitted in the following way (x=i or x=j):

$$|xj|_{\mathbf{i}} = \mathrm{Sum}_y\{|xjy|_{ij} \mid y<>i, j \}$$

All the other probabilities are retained:

$$|xy|_{\mathbf{i}} = |xyy|_{\mathbf{ij}}.$$

Let us call the obtained 3-tuples **(ij)-admissible**.

In the general case, after the numbers $p_1, ..., p_a$ of the a-tuple $\mathbf{p} = (p_1 ... p_a)$ have been "excluded", and the notion of **p**-admissible (a+1)-tuples has been defined (and to every **p**-admissible x corresponds the probability $|xj|_{\mathbf{p}}$), the next number q in {1,2,...,n}- **p** is "excluded". The p -probabilities $|\mathbf{z}q|_{\mathbf{p}}$ are splitted:

$$|\mathbf{z}q|_{\mathbf{p}} = \mathrm{Sum}_y\{|\mathbf{z}qy|_{\mathbf{p}q} \mid y \text{ not in } \mathbf{p} \text{ and } y<>j\}|. \text{ ---------------(a)}$$

The other probabilities are retained:

$|\mathbf{z}y|_{\mathbf{p}} = |\mathbf{z}yy|_{\mathbf{p}q}$,

where $y<>q$ (and, of course, y not in **p**, since **z**y is **p**-admissible). The left hand side tuples of (a) and (b) are called **p**q-admissible.

Obviously, the tuple $(x_1...x_{a+1})$ is **p**-admissible (where $\mathbf{p}=(p_1...p_a)$), if and only if for all $i<=a$:

(1) x_{i+1} not in $\{p_1, ..., p_i\}$,

(2) x_i not in $\{p_1, ..., p_i)$ implies $x_{i+1} = x_i$.

At the very end of the exclusion process we will have some permutation $(p_1...p_n)$ of the set {1, 2, ..., n}. Let us denote $\mathbf{p} = (p_1...p_{n-1})$. To every **p**-admissible n-tuple **x** a probability $|\mathbf{x}|_{\mathbf{p}}$- is assigned. According to our problem, special attention is paid to n-tuples $\mathbf{x} = (x_1...x_n)$ having the property: $x_i <> x_{i+1}$ for at least k values of i (k is a natural number, $k<n$). Let us denote by S_k the set of all such **x**'s. We will use only the symmetricity of S_k, i.e. if pi is any permutation of the numbers {1, 2, ..., n}, then

$$(x_1...x_n) \text{ in } S_k <--> (pi(x_1)...pi(x_n)) \text{ in } S_k.$$

The "quality" of every permutation $(p_1...p_n)$ of the set {1, 2, ..., n} is defined by the probability

$$T(\mathbf{p}) = \mathrm{Sum}_{\mathbf{x}}\{ |\mathbf{x}|_{\mathbf{p}} \},$$

where **x** ranges over all **p**-admissible n-tuples of S_k (recall that $\mathbf{p} = (p_1...p_{n-1})$).

We extend this "quality" definition to any tuple $\mathbf{q} = (q_1...q_a)$ containing no repetitions (a<=n-1).

If a<=n-2 and $\mathbf{q}$ contains all the numbers of the set {1, 2, ..., n} except q_{a+1}, ..., q_n, then we define by recursion:

$$T(\mathbf{q}) = T(\mathbf{q}q_{a+1})+T(\mathbf{q}q_{a+2})+...+T(\mathbf{q}q_n).$$

In particular, for the empty tuple **o**:

$$T(\mathbf{o}) = T(1) + T(2) + ... + T(n).$$

We will prove that for all **q** and all q not in **q** the value of T(**q**q) can be computed having the probabilities $|\mathbf{x}|_\mathbf{q}$ for all **q**-admissible (a+1)-tuples **x**. This will be proved by showing that T(**q**q) can be represented as

$$T(\mathbf{q}q) = Sum_\mathbf{x}\{ c(\mathbf{x})|\mathbf{x}|_\mathbf{q} \},$$

where **x** ranges over all **q**-admissible tuples, and c(**x**)'s are natural numbers that can be computed having **x**, **q** and q.

First we consider the case $\mathbf{q} = (q_1, ..., q_{n-2})$. Let us denote by q, t the only two numbers (1<=q<t<=n), which do not belong to the tuple **q**. Then, for any **q**q-admissible tuple **x** = $(x_1, ..., x_n)$ we have x_n = t, and $|\mathbf{x}|_{\mathbf{q}q} = |x_1...x_{n-1}|_\mathbf{q}$, hence,

$$T(\mathbf{q}q) = Sum_\mathbf{y}\{ c(\mathbf{y})|\mathbf{y}|_\mathbf{q} \},$$

where **y** ranges over all **q**-admissible (n-1)-tuples, and

$$c(\mathbf{y}) = 1, \text{ if } \mathbf{y}t \text{ in } S_k,$$

c(**y**) = 0, otherwise.

Now let us consider the next case $\mathbf{q} = (q_1, ..., q_{n-3})$. By q, s, t we denote the 3 numbers which do not belong to **q**. Then, by definition:

$$T(\mathbf{q}q) = T(\mathbf{q}qs) + T(\mathbf{q}qt).$$

As we already know,

$$T(\mathbf{q}qs) = Sum_\mathbf{x}\{ |x|_{\mathbf{q}q} \},\text{----------------- (1)}$$

where **x** ranges over **q**q-admissible (n-1)-tuples such that **x**t in S_k. The set S_k is symmetric, therefore, replacing s by t in (1), we will obtain T(**q**qt).

The sum expression T(**q**qs)+T(**q**qt) can be simplified considering the following cases, where **x** is an arbitrary tuple of the expression (1):

1) **x** = **y**s, where **y** does not contain s, t. Then, replacing s by t, we obtain **y**t, and

$$|\mathbf{y}s|_{\mathbf{q}q} + |\mathbf{y}t|_{\mathbf{q}q} = |\mathbf{y}|_{\mathbf{q}}.$$

2) **x** = **y**t, similarly.

3) **x** = **y**ss...s, where **y** does not contain s, t. Then, replacing s by t, we obtain **y**tt...t, and

$$|\mathbf{y}ss...s|_{\mathbf{q}q} = |\mathbf{y}ss...|_{\mathbf{q}},$$

$$|\mathbf{y}tt...t|_{\mathbf{q}q} = |\mathbf{y}tt...|_{\mathbf{q}}.$$

4) **x**=**y**tt...t, similarly.

Hence, we obtain the following representation of T(**q**q):

$$T(\mathbf{q}q)= Sum_{\mathbf{x}}\{ c(\mathbf{x})|\mathbf{x}|_{\mathbf{q}} \},\text{--------------------} (2)$$

where **x** ranges oven all **q**-admissible (n-2)-tuples, and the numbers c(**x**) can be computed from **x**, **q**, q. The representation (2) is symmetric to s, t: if **x** contains s (then **x** does not contain t), then replacing (in **x**) s by t, we obtain **y** such that c(**y**)=c(**x**).

Now we can perform the induction step for the general case. Let **q** = ($q_1...q_a$) be a tuple without repetitions, and the numbers q, r_2, ..., r_{n-a} do not belong to **q**. Let us suppose that

$$T(\mathbf{q}qr_2) = Sum_{\mathbf{x}}\{ c(\mathbf{x})|\mathbf{x}|_{\mathbf{q}q} \}, \text{--------------}(3)$$

where **x** ranges over all **q**q-admissible tuples, and the following symmetry condition holds: if **x** is **q**q-admissible, and **s** = (s_3, ..., s_n-a) is any permutation of the numbers (r_3, ..., r_{n-a}), then the action of **s** on **x** yields a tuple **y** such that c(**y**)=c(**x**).

By definition:

$$T(\mathbf{q}q) = T(\mathbf{q}qr_2)+T(\mathbf{q}qr_3)+...+T(\mathbf{q}qr_{n-a}).\text{----------------} (4)$$

To obtain from (3) a representation of T(**q**qr_i) (where i>=3), one can apply to all **q**q-admissible (a+2)-tuples **x** the permutation of the set {1, 2, ..., n} transposing r_2 and r_i. Thus, the expression (4) can be simplified by considering the following cases:

1) **x** = **y**r_2, where **y** does not contain r_2. Applying the permutations mentioned above we obtain the tuples **y**r_3, ..., **y**r_{n-a}, and

$$|\mathbf{y}r_2|_{\mathbf{q}q} + |\mathbf{y}r_3|_{\mathbf{q}q} + ... + |\mathbf{y}r_{n-a}|_{\mathbf{q}q} = |\mathbf{y}|_{\mathbf{q}}.$$

2) **x** = **y**r_i, where i>=3 and **y** does not contain r_i. Then, (3) contains **all** the tuples of this kind: **y**r_3, ..., **y**r_{n-a}, and with the same coefficient c(**x**). Applying the permutations of the set {1, 2, ..., n} mentioned above we obtain from **y**r_i: a) the tuple **y**r_2, b) n-a-3 copies of **y**r_i itself. All this will be included into (4):

$$(n-a-2)(|\mathbf{y}r_2|_{\mathbf{q}q} + ... + |\mathbf{y}r_{n-a}|_{\mathbf{q}q}) = (n-a-2)|\mathbf{y}|_{\mathbf{q}}.$$

3) $\mathbf{x} = \mathbf{y}r_2...r_2$, where $\mathbf{y}$ does not contain r_2. The permutations mentioned above yield all the tuples $\mathbf{y}r_i...r_i$ (i>=3), where

$$|\mathbf{y}r_i...r_i|_{\mathbf{q}q} = |yr_i...|_{\mathbf{q}},\text{-------------(5)}$$

and after this "tail reduction" the expression (4) contains all the r_i (i>=2) symmetrically.

4) $\mathbf{x} = \mathbf{y}r_i...r_i$, where i>=3 and $\mathbf{y}$ does not contain r_i. Then, (4) contains all the similar tuples

$$\mathbf{y}r_3...r_3, ..., \mathbf{y}r_{n-a}...r_{n-a}$$

with the same coefficient c(**x**). Applying the permutations mentioned above we obtain n-a-2 copies of each tuple $\mathbf{y}r_i...r_i$ (i>=2). Here (5) holds as well, and after the "tail reduction" the expression (4) contains all the r_i (i>=2) symmetrically.

Hence, we have obtained the following representation of T(**q**q) (where q is any number not contained in **q**):

$$T(\mathbf{q}q) = Sum_{\mathbf{x}}\{ c'(\mathbf{x})|\mathbf{x}|_{\mathbf{q}} \},$$

where **x** ranges over all **q**-admissible tuples, and the coefficients c'(**x**) can be computed from **x**, **q** and q. This representation contains all the numbers r not in **q**q symmetrically.

Hereby our induction step is concluded.

Thus, we have proved that the estimate T(**q**q) of any tuple **q**q (without repetitions) can be computed from the probabilities $|\mathbf{x}|_{\mathbf{q}}$ of all **q**-admissible tuples **x**. This proves the correctness of the exclusion algorithm described in Section 5.

Concluding remark. For the empty tuple **o** our result means that the estimate T(o) depends only on the number n and a symmetric set S_k of n-tuples from the set {1, 2, ..., n}. These are the only parameters of the Section 5 algorithm. In particular, when

$$S_k = \{ (x_1...x_n) \mid x_i <> x_{i+1} \text{ for } >=k \text{ values of } i \},$$

then, as we have shown in Section 5:

$$T(\mathbf{o}) / n! = P\{ Z_2+Z_3+...+Z_n>=k),$$

where the random variables Z_i are defined in Section 3.

References

[Ba 74-1] Barzdins J. Limiting synthesis of tau-indices. "Theory of Algorithms and Programs", vol. 1, Latvia State University, 1974, pp. 112-116 (in Russian).

[Ba 74-2] Barzdins J. Prediction and limiting synthesis of finite automata. "Theory of Algorithms and Programs", vol. 1, Latvia State University, 1974, pp. 129-144 (in Russian).
[BF 72] Barzdins J., Freivalds R. On the prediction of general recursive functions. "Soviet Math. Dokl.", 13, 1972, pp. 1224-1228.
[BF 74] Barzdins J., Freivalds R. Prediction and limiting synthesis of effectively enumerable classes of functions. "Theory of Algorithms and Programs", vol. 1, Latvia State University, 1974, pp. 101-111 (in Russian).
[FBP 91] Freivalds R., Barzdins J., and Podnieks K. Inductive inference of recursive functions: complexity bounds. Lecture Notes in Computer Science, 502, Springer-Verlag, 1991, pp. 111-155.
[LMS 56] De Leeuw K., Moore E. F. et al. Computability by probabilistic machines. "Automata Studies (Ann. of Math. Studies, No.34)", Princeton Univ. Press, Princeton, N.J., 1956, pp. 183-212.
[Po 75] Podnieks K. Probabilistic synthesis of enumerated classes of functions. "Soviet Math. Dokl.", 16, 1975, pp. 1042-1045.
[Po 77-1] Podnieks K. Computational complexity of prediction strategies. "Theory of Algorithms and Programs", vol. 3, Latvia State University, 1977, pp. 89-102 (in Russian).
[Po 77-2] Podnieks K. Probabilistic program synthesis. "Theory of Algorithms and Programs", vol. 3, Latvia State University, 1977, pp. 57-88 (in Russian).