On the distribution of the number of computations in any finite number of subtrees for the stack algorithm

Johannesson, Rolf; Zigangirov, Kamil

Published in:
IEEE Transactions on Information Theory

Published: 1985-01-01

Citation for published version (APA):

General rights
Copyright and moral rights for the publications made accessible in the public portal are retained by the authors and/or other copyright owners and it is a condition of accessing publications that users recognise and abide by the legal requirements associated with these rights.

• Users may download and print one copy of any publication from the public portal for the purpose of private study or research.
• You may not further distribute the material or use it for any profit-making activity or commercial gain
• You may freely distribute the URL identifying the publication in the public portal?
Correspondence

On the Distribution of the Number of Computations in Any Finite Number of Subtrees for the Stack Algorithm
ROLF JOHANNesson AND KAMIL SH. ZIGANGIROV

Abstract—Multitype branching processes have been employed to determine the stack algorithm computational distribution for one subtree. These results are extended here to the distribution of the number of computations in any finite number of subtrees. Starting from the computational distribution for $K-1$ subsequent subtrees, a recurrent equation for the distribution for $K$ subsequent subtrees is determined.

I. INTRODUCTION

It is well known [1]–[3] that the curse of sequential decoding is that its behavior is limited by a computational distribution (conditioned on correct decoding), which is asymptotically Pareto. Until now little attention has been paid to the problem of determining the computational distribution for a small number of computations. In our previously published papers [4], [5] we employed multitype branching processes to determine the stack algorithm computational distribution for the first incorrect subtree. The results in [5] were obtained for random tree codes, while in [4] we considered the class of binary, rate $1/2$, complementary + random tree codes, a fictitious entity that is a reasonable model for constant convolutional codes that have column distance $d_c = 2$.

In a complementary + random tree code the channel input symbols on the transmitted path are all zeros, and the channel input symbols on the incorrect branches stemming from nodes on the correct path are all ones. For all other branches, each channel symbol is chosen independently and according to a specified probability distribution. In this work we extend the results in [4] and determine the probability distribution for the number of computations in any finite number of subsequent subtrees.

II. PRELIMINARIES

Let $C_n$, $n = 1, 2, 3, \cdots$, denote the number of computations made by the sequential decoder in order to decode the $n$th correct node. In Fig. 1 we show a partially explored code tree with $C_1 = 4$. For a tree of unbounded length the random variables $C_1, C_2, C_3, \cdots$ have the same distribution, but they are certainly not independent.

Let $M_n$ be the cumulative metric for the first $n$ branches of the correct path; i.e.,

$$M_n = \sum_{k=1}^{n} z_k,$$

where $z_k$ is the branch metric for the $k$th branch. Let $D_n$, $n = 0, 1, 2, \cdots$, be the difference between the cumulative metric and the smallest succeeding value; i.e.,

$$D_n = M_n - M_{\text{min}, n},$$

where

$$M_{\text{min}, n} = \min \{ M_n, M_{n+1}, \cdots \}.$$

The nonnegative random variables $D_n$ all have the same well-known distribution [4], [6] since the metrics for the correct path stemming from the $n$th node have the same statistical character for all $n$.

Finally, let $C^N_n$ be the total number of computations in the $n$th, $(n+1)$th, $\cdots$, $N$th subtree

$$C^N_n = \sum_{k=n}^{N} C_k.$$  

III. RECURRENT EQUATION FOR DETERMINING THE DISTRIBUTION OF $C^N_n$

Let $F_n^N(r,i,k)$ be the conditional distribution for the random variable $C^N_n$ on the condition that $D_n = i$ and $z_n = k$; i.e.,

$$F_n^N(r,i,k) = P[C^N_n = r|D_n = i, z_n = k].$$

By dividing the number of computations in the $n$, $n+1$, $\cdots$, $N$th subtrees into two parts, $C_n$ and $C^N_{n+1}$, we have

$$F_n^N(r,i,k) = \sum_{i=1}^{r-1} P[C_n = s|D_n = i, z_n = k] \cdot P[C^N_{n+1} = r-s|D_n = i, z_n = k].$$

The conditional probability distribution for $C^N_{n+1}$ can be expanded as

$$P[C^N_{n+1} = r-j|D_n = i, z_n = k] = \sum_{j=0}^{\infty} \sum_{l=0}^{\min(i,j)} P[C_n = r-s|D_n = i, z_n = k] \cdot P[D_{n+1} = j, z_{n+1} = l|D_n = i, z_n = k].$$
where the variable $l$ is summed over all branch metric values.

The random variable $C_{n+1}^N$ conditioned on $D_{n+1}$ and $z_{n+1}$ is independent of $D_n$ and $z_n$. Furthermore, the random variables $D_{n+1}$ and $z_{n+1}$ conditioned on $D_n$ are independent of $z_n$. Hence we obtain the recurrent equation for $F_n^N(r, i, k)$

$$F_n^N(r, i, k) = \sum_s \sum_j f(s, i, k) \cdot F_{n+1}^N(r - s, j, l) g(j, l, i),$$

where $n < N$,

$$f(s, i, k) = F_n^N(s, i, k)$$

and

$$g(j, l, i) = P[D_{n+1} = j, z_{n+1} = l | D_n = i].$$

The conditional probability distribution $g(j, l, i)$ is easily calculated as follows:

$$g(j, l, i) = P[D_{n+1} = j, z_{n+1} = l | D_n = i]$$

$$= P[D_n = i] P[D_{n+1} = j | D_n = i] \cdot P[z_{n+1} = l | D_n = i],$$

where we have used the fact that the random variables $D_{n+1}$ and $z_{n+1}$ are statistically independent. Let us assume that we have the branch metric set $\{+1, -4, -9\}$, cf. [4]; then

$$P[D_n = i | D_{n+1} = j, z_{n+1} = l] = \begin{cases} 1, & i = j, l = 1 \\ 1, & i = j - l, l \geq 0 \\ 0, & \text{elsewhere.} \end{cases}$$

Starting from the conditional probability distribution $f(s, i, k)$, which can be determined by the methods given in [4], we can use (8) and recursively obtain the conditional probability distribution $F_n^N(r, i, k)$ for any $n, 0 < n < N$. Finally, we calculate the probability distribution for the number of computations in $K \geq 1$ subtrees as

$$P[C_1^K = r] = \sum_i \sum_k F_1^K(r, i, k) h(i, k),$$

where

$$h(i, k) = P[D_n = i, z_n = k] = P[D_n = i] P[z_n = k].$$

IV. DISCUSSION OF NUMERICAL RESULTS AND SIMULATIONS

The problem of resolving ties among the cumulative metrics is discussed in detail in [4]. When there was no obvious way to resolve ties the pessimistic policy “in case of ties extend an incorrect node first” was used to obtain an upper bound on the distribution function for $C_1$, and the optimistic policy “in case of ties extend the correct node first” was used to obtain a lower bound. Because of the close agreement between the simulated $C_1$-curves and the corresponding lower bounds in [4], we will use these lower bounds as our conditional distribution $f(s, i, k)$ in (8).

By combining the branching processes methods described in [4] and the recurrent equation (8), we evaluate theoretical lower bounds on the distribution function of the random variable $C_1$, $P[C_1^K \geq r]$ for the case where transmission takes place over a binary symmetric channel (BSC) with crossover probability $p = 0.045$. Since we are only considering rate $R = 1/2$ codes, this crossover probability corresponds to transmission at the computational cutoff rate $R_0$. The bounds are evaluated for the Fano


A New Class of Check-Digit Methods for Arbitrary Number Systems

H. PETER GUMM

Abstract—For arbitrary number systems we present a new check-digit method that detects all single-digit errors and all transpositions of adjacent digits using a single check digit from the given number system. In previous methods at least one type of transpositional error had to remain undetected. The key to this method lies in using the dihedral groups together with appropriate transformations in the important cases, where the numbers are represented in base 2r with r odd.

I. INTRODUCTION

Empirical studies have shown [9], [10] that the most common typing errors that occur when data are entered on a keyboard are

- single-digit errors (one digit wrong)
- format errors (one digit inserted or left out)
- transpositions (interchanging of two adjacent digits)

To detect such errors, the original string of data is supplied with one check digit, where digit now means a “digit” in the chosen number system, i.e., the check digit is numerical for numerical data and may be alphanumerical for alphanumerical data. Various check-digit methods have been designed for the decimal number system. Each method is able to detect all “single-error mistakes,” but they fail to detect all transposition errors. At least one (and often not more than one) erroneous transposition is undetectable with those methods, see [1]-[3].

The format errors principally cannot be detected by only one check digit, and every method that detects single-digit errors will automatically detect about 90 percent of all format errors. The use of two check digits as proposed in [3] and [11] is not advised since the previously mentioned studies have also shown that the absolute number of errors that occur roughly doubles when the number of digits increases by two. Thus generally specifying and checking for a fixed format seems appropriate to detect format errors. We therefore concentrate on methods to detect single-digit errors and transposition errors.

Clearly, for numbers to the base 2 such a check digit method is impossible since the numbers 00, 01, 10 would have to be supplied with mutually different check digits from the set {0, 1}.

If a number n has the digits $d_1, d_2, \ldots, d_r$ in base r, i.e., $n = \sum d_i r^i$, then using $p = -\sum d_i$ (in modulo r arithmetic) for a check digit will detect every single-digit error. The secured number then has the digit representation $d_1, d_2, \ldots, d_r, p$, and

REFERENCES

[1] I. M. Jacobs and E. R. Berlekamp, “A lower bound to the distribution of the Fano metric set { +1, - 3, - 7} and for the “unmatched” sets { +1, - 5, - 11} and { +1, - 3, - 7}. The bounds are shown in Figs. 2-4 together with lower bounds on the distribution function of the random variable $C_e$, $P[C_e \geq r]$ and simulations of $P[C_e \geq r]$; and $P[C_{eu} \geq r]$ for the fixed nonsystematic optimum distance profile (ODP) convolutional code with memory length $M = 23$ and $d_{ac} = 25$ [7]. As many as 100 000 frames of 500 information symbols augmented by a tail of 1000 zeros were transmitted at rate $R = R_0$ and decoded with different metrics.

Finally, it is interesting to notice the wide gap between the $C_e$ and $C_{eu}$ curves for the metric set { +1, - 3, - 7}. When estimating the crossover probability it does not pay to be pessimistic!

Fig. 4. Computational distribution functions obtained from simulations and branching processes bounds for the metric set { +1, - 3, - 7}. This set is pessimistic for the simulated channel.

Since our fixed code has a better distance profile than a random code, the simulated curves are slightly below the corresponding calculated values. We also notice the close agreement between all curves for the Fano metric set { +1, - 3, - 7} and the well-known Pareto asymptote (slope -1).

II. SIMULATION OF DECODING

At this point we can use a Monte Carlo simulation to test our method as described in Section II. Clearly, the computations are too formidable to permit an exhaustive search through all possible numbers. The method is evaluated for the Fano metric set { +1, -3, -7) and { +1, -5, -11). The bounds are shown in Figs. 2-4 together with lower bounds on the distribution function of the random variable $C_e$, $P[C_e \geq r]$ and simulations of $P[C_e \geq r]$; and $P[C_{eu} \geq r]$ for the fixed nonsystematic optimum distance profile (ODP) convolutional code with memory length $M = 23$ and $d_{ac} = 25$ [7]. As many as 100 000 frames consisting of 500 information symbols augmented by a tail of 1000 zeros were transmitted at rate $R = R_0$ and decoded with different metrics.

Since our fixed code has a better distance profile than a random code, the simulated curves are slightly below the corresponding calculated values. We also notice the close agreement between all curves for the Fano metric set { +1, - 3, - 7} and the well-known Pareto asymptote (slope -1). Finally, it is interesting to notice the wide gap between the $C_e$ and $C_{eu}$ curves for the metric set { +1, - 3, - 7}. When estimating the crossover probability it does not pay to be pessimistic!

REFERENCES

[1] I. M. Jacobs and E. R. Berlekamp, “A lower bound to the distribution of the Fano metric set { +1, - 3, - 7} and for the “unmatched” sets { +1, - 5, - 11} and { +1, - 3, - 7}. The bounds are shown in Figs. 2-4 together with lower bounds on the distribution function of the random variable $C_e$, $P[C_e \geq r]$ and simulations of $P[C_e \geq r]$; and $P[C_{eu} \geq r]$ for the fixed nonsystematic optimum distance profile (ODP) convolutional code with memory length $M = 23$ and $d_{ac} = 25$ [7]. As many as 100 000 frames consisting of 500 information symbols augmented by a tail of 1000 zeros were transmitted at rate $R = R_0$ and decoded with different metrics.

Finally, it is interesting to notice the wide gap between the $C_e$ and $C_{eu}$ curves for the metric set { +1, - 3, - 7}. When estimating the crossover probability it does not pay to be pessimistic!