Efficient Bernoulli Probability Distribution Estimation for Arithmetic Coding

Vladimir Čeperković, Milan Prokin
University of Belgrade – School of Electrical Engineering
Belgrade, Serbia

Dragana Prokin
The School of Electrical and Computer Engineering of Applied Studies
Belgrade, Serbia

Abstract—**State-of-the-art arithmetic coding utilizes finite state machines to select symbol probability estimates from the predefined finite set, resulting in inherently limited precision and increased code length. This paper presents novel method for Bernoulli probability distribution estimation, based on low pass filtering with varying dominant pole. This solution uses integer-only arithmetic, without multiplication or division operations, thus providing significant reduction in required resources of IoT devices.**

Keywords-data compression; probability estimation; arithmetic coding; varying dominant pole filter

I. INTRODUCTION

Global transition to information society led to an exponential grow of the amount of data produced, resulting in a constantly increasing requirements for processing capacities, storage size, and communication bandwidth. Data compression is a technique commonly employed to dramatically simplify their fulfillment, by exploiting redundancy in source data.

A block diagram of state-of-the-art one-dimensional (audio) encoder is shown in Fig. 1. An input signal is first filtered and decomposed. The actual process depends on the prior knowledge about signal type. The resulting data samples are processed by the quantizer, which produces quantized data samples in case of lossy compression, or just passes received data samples to the probability estimator and the entropy encoder. The probability estimator uses internal model to determine the symbol probabilities within the specified contexts and feed them to the entropy encoder. Finally, the entropy encoder removes redundancy from the samples, using their estimated probabilities.

It is well established that arithmetic [1]-[3] and range coders [4] may be used to remove all the redundancy that can be described in a digital message. Thus, the efficiency of the entropy coder depends only on the ability of the probability estimator to accurately model probabilities of the symbols being encoded. However, the complexity of the probability estimator is limited by the computational resources, and by the available memory. Single pass encoding and decoding is a common requirement, where the probability estimates are continuously produced based only on samples preceding the one being encoded. Common design approach is to use finite state machine (FSM), tightly coupled with entropy encoder, to select a probability estimate from a predefined finite set.

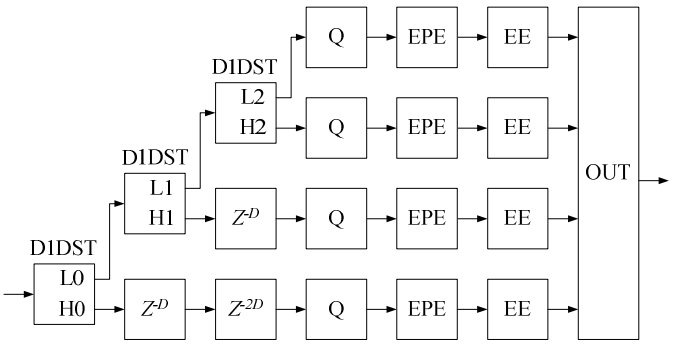

Figure 1. State-of-the-art one-dimensional (audio) encoder

Arithmetic Q-coder, presented in [5]-[8], utilizes FSM with just 60 states. The current state is updated sporadically, only during arithmetic coder normalization procedure. Similarly, the arithmetic Z-coder presented in [9]-[12] utilizes FSM with 256 states, where the current state can be changed after each symbol. In well-known image compression standard JPEG 2000, MQ arithmetic coder is utilized, which is similar to QM-coder adopted in the original JPEG image compression standard described in [13]-[15]. The MQ probability estimation utilizes FSM with 92 effective states, where the current state is changed after each symbol using a tuned Markov model.

However, all those probability estimators suffer from inherently limited precession, that limits the coding efficiency. The goal of this paper is to propose a novel probability distribution estimation method, whose precision is limited only by available information, without increasing computational complexity.

This paper is structured as follows. The theoretical background of the proposed probability distribution estimator is presented in section II. Proposed probability distribution estimator is presented in section III. Section IV describes experimental verification setup and results. Finally, brief conclusion is stated in section V.

This work was partially supported by Ministry of Education, Science and Technological Development of Republic of Serbia. under grant nos. TR32039 and TR32047.

978-1-7281-6949-1/20/$31.00 ©2020 IEEE

II. BACKGROUND

Consider a sequence of symbols x_i from the symbol set $S = \{S_0, S_1\}$, whose first k symbols are known. Assuming that the symbol sequence is generated by a Bernoulli process with time-invariant probability distribution $p(x)$, the probability of occurrence of the symbol x can be estimated using the maximum likelihood probability estimator (MLE) as in

$$\hat{p}_k(x) = \frac{\sum_{i=1}^{k} eq(x, x_i)}{k}. \quad (1)$$

where the $\hat{p}_k(x)$ is the probability distribution estimated from the first k symbols, and eq is the symbol equality function defined as

$$eq(a,b) = \begin{cases} 1, & a = b \\ 0, & a \neq b \end{cases}. \quad (2)$$

However, this approach is limited when tracing non-stationary probability distributions, that better model the real-life behavior of symbol sequences. The adaptation to a probability distribution change is slow and, in general, does not converge to actual value. Another problem is computational cost of the integer division, since k is an arbitrary integer. To overcome those issues, a non-stationary process model must be used, where the probability distribution $p_i(x)$ of the symbol x_i depends on the sample index i. Now, consider an auxiliary binary sequence, u_i, formed from the symbol sequence as follows

$$u_i = eq(x_i, S_0). \quad (3)$$

The binary sequence u_i can be represented as a sum of two stochastic signals, the expected value of the sample u_i and the zero-mean noise n_i. However, since the expected value of the sample u_i is the probability of the sample x_i being equal to the S_0 and, the signal u_i is equal to the probability $p_i(S_0)$ with added noise, as in

$$u_i = p_i(S_0) + n_i. \quad (4)$$

Assuming that the $p_i(S_0)$ and the n_i are two independent stochastic processes, the power spectral density of the noise n_i is constant over frequency, which can be verified from the autocorrelation function. The noise variance is a function of the probability $p_i(S_0)$ as in

$$E\{n_i^2\} = p_i(S_0)(1 - p_i(S_0)). \quad (5)$$

Thus, the worst-case noise power is equal to the signal power, occurring when the $p_i(S_0)$ is 0.5. However, because the symbol probability distribution $p_i(x)$ is slowly changing, the power spectral density of the signal $p_i(S_0)$ is concentrated in low frequency range, i.e. $p_i(S_0)$ is a narrow band signal. Consequently, the signal to noise ratio can be significantly improved by filtering the auxiliary sequence u_i with low pass filter, thus obtaining the probability $p_i(S_0)$ estimate.

The filter bandwidth is a compromise between ability to follow probability distribution changes and the noise rejection. However, the initial condition problem has also to be accounted for. Namely, without prior knowledge, the filter output has to be initialized to the uniform probability distribution. Unfortunately, this results in poor estimator performance if the actual probability distribution is not close to uniform, when the filter output will be slow to converge.

The problem arises because the initial conditions can be represented as an additive step disturbance to the $p_i(S_0)$ at index zero. Because the bandwidth of this disturbance far exceeds the bandwidth of the stochastic process model, it will also exceed the filter bandwidth. In order to overcome this problem, without compromising the estimator precision, the variation of the dominant pole of the filter is proposed, from maximum to minimum value. Thus, the filter would have higher bandwidth in the beginning of the symbol sequence x_i, to quickly identify initial conditions, but considerably smaller bandwidth later, to reject the noise.

III. PROBABILITY DISTRIBUTION ESTIMATOR DESIGN

In order to use probability distribution estimate in an arithmetic or range coder, the probability estimate must be scaled to an integer, h_k, in the range $[0, T]$. The total T is a design parameter of the entropy encoder, that is commonly chosen to be a power of two in order to replace multiplication and division operations by left shift and right shift operations respectively. The parameter T value is high enough for integer representation of probability distribution to be at least 16-bit wide, resulting in negligible loss of precision.

Furthermore, the entropy coder cannot encode symbols whose estimated probability is zero. However, because the probability distribution is specified by a single signal, this can be easily accomplished by limiting the scaled filter output to the $[1, T\text{-}1]$ range.

The main difference between an efficient integer-only implementation of the probability distribution estimator and the theoretical background method is in the introduction of scaling by factor T before filtering of the auxiliary sequence u_i. This is accomplished at no computational cost by scaling the output of the equality function (2). Consequently, the filter applied to the auxiliary sequence can be implemented as an integer-only low-pass filter with unity gain. The output of the filter is saturated to the input range of the entropy coder $[1, T\text{-}1]$. The remaining probability estimate, i.e. the probability of occurrence of the symbol S_1, is calculated so that the sum of the estimated probabilities is one.

Without prior knowledge, the filter output is initialized to the uniform probability distribution, i.e. $T/2$.

A. *Maximum Likelihood Filter*

It should be noted that it is not possible to design an exact maximal likelihood estimator for the non-stationary stochastic process, as a closed form of the probability distribution model $p_i(x)$ is unknown. However, knowing that $p_i(x)$ is narrow-band process, the estimator is designed so that in limit it becomes equivalent to the maximal likelihood probability estimator (7). Thus, if the N is the maximal estimator length determining steady state estimator bandwidth, the estimator is defined as in

$$h_k = \frac{\sum_{i=k-\min(k,N)+1}^{k} T \cdot eq(x_i, S_0)}{\min(k, N)}. \tag{6}$$

Unfortunately, the estimator (6) requires division by an arbitrary integer. To overcome this issue, observe that the maximum likelihood estimator (6) is a moving average filter with length equal to the minimum of the number of elements in the sequence k and the maximal filter length N. However, if only power-of-two filter lengths are allowed, the estimator becomes determined by

$$\begin{aligned} w &= \lfloor \log_2(\min(k, N)) \rfloor \\ h_k &= 2^{-w} \cdot \sum_{i=k-2^w}^{k} T \cdot eq(x_i, S_0) \end{aligned}. \tag{7}$$

The estimator (7) is integer only approximation of the maximum likelihood probability estimator, without multiplication or divisions, that can be efficiently implemented in either hardware or software. Note that both estimators (6) and (7) are inherently implementing a filter with varying dominant pole.

B. *First Order Filter*

Another approach is to design a simple first order filter, with a varying dominant pole, that fulfills the requirements derived in the theoretical background. In order to keep efficient implementation, without multiplication and division operators, the filter pole is constrained to the form $1-2^{-a}$, where a is an integer. This ensures that a multiplication by a pole can be replaced by a subtraction and shift right operation. The proposed first order varying pole filter is defined as

$$\begin{aligned} a &= \lfloor \log_2(\min(k, N)) \rfloor \\ h_k &= (1-2^{-a}) h_{k-1} + 2^{-a} \cdot T \cdot eq(x_k, S_0) \end{aligned}. \tag{8}$$

The pole varying method is an extension of the method applied in the estimator (7), where individual contribution of each auxiliary sample u_i is halved whenever the length k of input sequence is doubled.

IV. EXPERIMENTAL VERIFICATION

To proposed probability distribution estimator is implemented in software, whose functional correctness was verified with both artificial and natural symbol sources. Fig. 2 provides a representative example of the power spectral density of an auxiliary sequence u_i. The frequency is normalized relative to Nyquist limit. Three characteristic regions are observable. The Dirac pulse at the zero frequency is a consequence of the non-zero average probability of the symbol S_0, which is 0.49 in the present example. Simultaneously, this is the only spectral component observed by the stationary maximum likelihood probability estimator (1). The power spectral density of the signal $p_i(S_0)$ is almost flat at 10dB level in the low band, up to the cut-off point at $2 \cdot 10^{-4}$. Finally, starting from normalized frequency $1 \cdot 10^{-3}$, only the noise remains at -7dB level. Although the sequence in question is a worst-case example regarding the signal to noise ratio, the power spectral density levels still differ by 17dB, and with 20dB difference in the transition area.

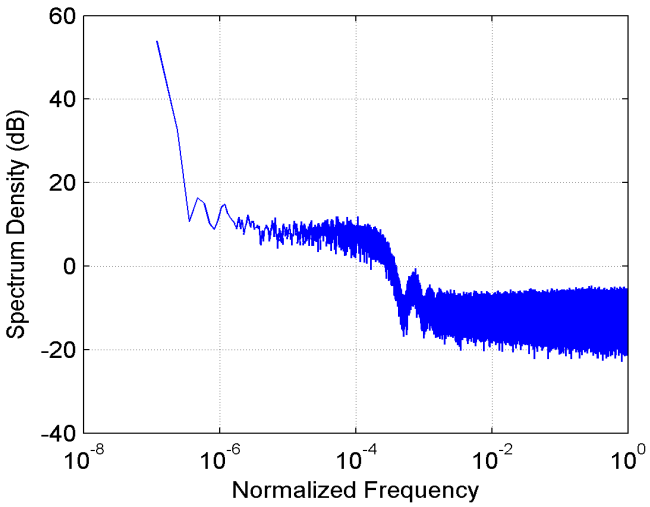

Figure 2. The power spectral density of the auxiliary sequence

Based on previous results, a parameterized discrete model of the symbol probability $p_i(S_0)$ was created for testing as in

$$Z\{p_i(S_0)\} = c_0 + c_1 \cdot \frac{0.005}{z - 0.995} \cdot (R(z) + \nu(z)). \tag{9}$$

The model parameters c_0 and c_1 are selected for each test independently, while the $\nu(z)$ is additive white noise. The reference $R(z)$ is a stochastic process with uniform distribution, changing once every 4000 samples of the $p_i(S_0)$ signal, corresponding to one second of simulation time. Fig. 3 presents an example of a simulated run with the stationary maximum likelihood probability estimator (1). As expected, the estimate is quickly to converge to the initial conditions but fails to track changes in probability distribution.

For comparison, Fig. 4 presents the same simulation, but repeated using the proposed estimators, with the maximum likelihood filter (MLF) and the first order filter (FOF). Both estimators follow changes in the underlying probability distribution.

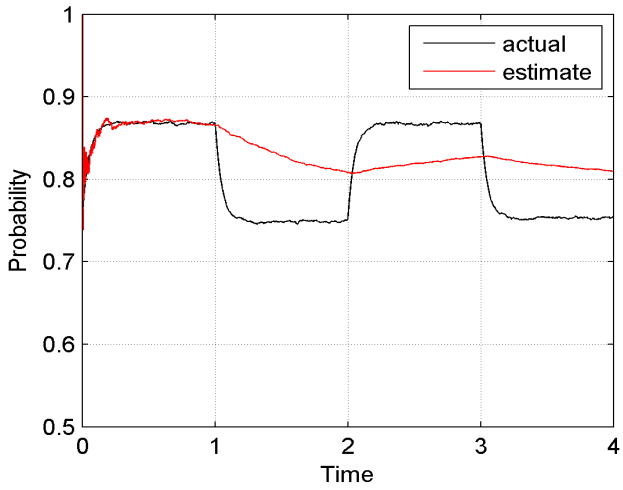

Figure 3. Simulation with stationary probability estimator

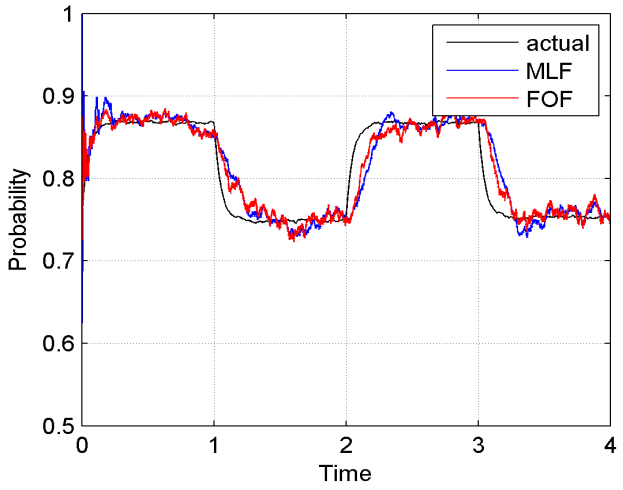

Figure 4. Simulation with proposed estimators

Finally, all three probability estimators were evaluated with the range coder in large number of sequences. The resulting average symbol code lengths are presented in Table I. The results strongly support the use of the proposed method with the FOF variant, especially considering it has negligible processing requirements compared to the other two methods.

TABLE I. CODE LENGTH STATISTIC

Method	Code Length	
	Mean	*Variance*
Stationary probability estimator	0.81256	0.003548
Proposed non-stationary, MLF	0.79955	0.003519
Proposed non-stationary, FOF	0.79741	0.003533

V. CONCLUSIONS

This paper presents novel method for Bernoulli probability distribution estimation, for use with arithmetic coding. The method uses low-pass filter with varying dominant pole applied to the auxiliary binary sequence to continuously produce estimated values, without inherent limitations in precision. This solution uses integer-only arithmetic without multiplication or division operations, thus providing significant reduction in required computational resources and enabling application in embedded systems with relatively small processing throughput, such as IoT devices compressing one-dimenstional data.

REFERENCES

[1] I. H. Witten, R. M. Neal, J. G. Cleary, “Arithmetic coding for data compression,” Commun. ACM, vol. 30, no. 6, pp. 520-540, June 1987.

[2] A. Moffat, R. M. Neal, I. H. Witten., “Arithmetic coding revisited”, Proc. Data Compression Conf., Snowbird, UT, pp. 202-211, Mar. 1995.

[3] A. Moffat, R. M. Neal, I. H. Witten, “Arithmetic coding revisited,” ACM Trans. Inform. Syst., vol. 16, no. 3, pp. 256-294, July 1998.

[4] G. N. N. Martin, “Range encoding: an algorithm for removing redundancy from a digitised message”, Proc. Video & Data Recording Conference, Southampton, 1979.

[5] J. L. Mitchell, W. B. Pennebaker, “Software implementations of the Q-coder,” IBM J. Res. Develop., vol. 21, no. 6, pp. 753-774, Nov. 1988.

[6] W. B. Pennebaker, J. L. Mitchell, G. G. Langdon, R. B. Arps., “An overview of the basic principles of the Q-coder adaptive binary arithmetic coder,” IBM J. Res. Develop., vol. 32, no. 6, pp. 717-726, Nov. 1988.

[7] W. B. Pennebaker, J. L. Mitchell, U.S. Patent 4,935,882, June 1990.

[8] W. B. Pennebaker, J. L. Mitchell, U.S. Patent 4,933,883, June 1990.

[9] L. Bottou, P. G. Howard, Y. Bengio, “The Z-coder adaptive binary coder,” Proc. Data Compression Conf., Snowbird, UT, pp. 13-22, Mar. 1998.

[10] Y. Bengio, L. Bottou, P. G. Howard, U.S. Patent 6,188,334, Feb. 2001.

[11] Y. Bengio, L. Bottou, P. G. Howard, U.S. Patent 6,225,925, May 2001.

[12] Y. Bengio, L. Bottou, P. G. Howard, U.S. Patent 6,281,817, Aug. 2001.

[13] G. K. Wallace, “The JPEG still picture compression standard,” IEEE Trans. Consum. Electron., vol. 38, no. 1, pp. 18-34, Feb. 1992.

[14] F. Ono, M. Denki, K. Kaisha, U.S. Patent 5,059,976, Oct. 1991.

[15] F. Ono, M. Denki, K. Kaisha, U.S. Patent 5,307,062, April 1994.