An Analog Circuit for Implementation of Arithmetic Coding

S. M. Reza Dibaj, Farshad Lahouti and Nasser Masoumi

School of Electrical and Computer Engineering
University of Tehran, Tehran, Iran
m.dibaj@ece.ut.ac.ir, lahouti@ut.ac.ir, nmasoumi@ut.ac.ir

Entropy coders are one of the important components of most compression standards. Specifically, arithmetic coding provides a high compression performance and is a key element for improved coding efficiency of recent multimedia communication standards such as H.264, and JPEG2000. However, efficient implementation of arithmetic codes in different applications is challenging.

At the arithmetic encoder, each symbol corresponds to an interval proportionate to its probability within an area updated by its preceding symbols. The bits transmitted are obtained based on a point within this interval. The important arithmetic encoding tasks include (disregarding source modeling): (a) Interval update and arithmetic operations, (b) Carry propagation and bit moves and (c) Interval renormalization.

In this work, inspired by the recent progress in channel decoding based on analog circuits, we present an analog circuit for implementation of binary arithmetic codes that facilitates a reduced circuit complexity and a higher speed when compared to the digital counterparts. The proposed solution is motivated by the iterative nature of the arithmetic coding algorithm. Specifically, the scheme performs the interval splitting by a simple resistance divider. Switches, based on the input symbols, select one of the resistance divider outputs for charging a capacitor that acts as an analog memory. This, together with the divider circuit, iteratively updates the arithmetic code intervals. Because of the limited practical precision of the circuit components, we consider a recently presented block implementation of arithmetic coding [1] to restart the en/decoding process periodically and avoid any potential errors.

The decoding is performed by reproducing the encoding process, the difference is only in the symbol search. The symbol search can be done simply by using a comparator circuit that compares the received codeword with the cumulative distribution of symbols.

Our simulations show that the time needed to encode one bit, using the proposed analog arithmetic coder, is less than 2ns. The best reported time to accomplish the same task based on a digital implementation of arithmetic coding is about 20ns [2]. Moreover, the complexity of the analog implementation is substantially smaller, as a digital implementation is challenging due the required multiplication and renormalization operations [3]. However, the limited precision of analog components requires a constrained input block length, and hence affects the possible compression gains. Specifically, considering a random error of 10^{-4} in the circuit elements, we assumed a short input block of 10 bits. In this case, the reduction of compression rate of the employed block arithmetic coding with respect to entropy, is between 5% to 10%, corresponding to highly skewed and uniform binary input distributions, respectively.

[1] S. M. R. Dibaj, F. Lahouti, "A low complexity block implementation of arithmetic coding", *Proceedings of European Space Agency On Board Payload Data Compression Workshop*, Noordwijk, Holland, June 2008.

[2] A. Said, "Comparative analysis of arithmetic coding computational complexity", *Proceedings of the Data Compression Conference*, March 2004

[3] A. Moffat, R. M. Neal, and I. H. Witten, "Arithmetic coding revisited," *ACM Transactions on Information Systems*, vol. 16, no. 3, pp. 256-294, 1998