A Markov Chain Model for Edge Memories in Stochastic Decoding of LDPC Codes

Kuo-Lun Huang†, Vincent Gaudet‡, and Masoud Salehi†

†Department of Electrical and Computer Engineering, Northeastern University, Boston, Massachusetts
‡Department of Electrical and Computer Engineering, University of Waterloo, Waterloo, Ontario

khuan@ece.neu.edu, vcgaudet@uwaterloo.ca, salehi@ece.neu.edu

Abstract—Stochastic decoding is a recently proposed method for decoding Low-Density Parity-Check (LDPC) codes. Stochastic decoding is, however, sensitive to the switching activity of stochastic bits, which can result in a latching problem. Using Edge Memories (EMs) has been proposed as a method to counter the latching problem in stochastic decoding. In this paper, we introduce a Markov chain model for EMs and study state transitions over decoding cycles. The proposed method can be used to determine the convergence and the required number of decoding cycles in stochastic decoding. Moreover, it can help to study the behavior of decoding process and to estimate the decoding time.

Index Terms—Stochastic decoding, Low-Density Parity-Check (LDPC) codes.

I. INTRODUCTION

Low-Density Parity-Check (LDPC) codes [1] are error-correcting codes with decoding performance close to the Shannon capacity limit [2]. These codes have been considered in several digital communication standards, such as WiMAX (IEEE 802.16) [3] and Digital Video Broadcast (DVB) satellite communications [4]. LDPC codes are often decoded by using a form of iterative belief propagation, such as the Sum-Product Algorithm (SPA), which can be graphically represented by a bipartite Tanner graph [5] with two distinct groups of nodes: variable-nodes and check-nodes. LDPC codes are decoded by passing messages iteratively between variable-nodes and check-nodes over the edges of the Tanner graph. The SPA requires passing probabilities or log-likelihood values between nodes through parallel connections with many paths. This increases the chip area needed for connections as well as the energy consumption. As a result, high-speed implementations of LDPC decoders have been a subject of active research in recent years.

Stochastic computation was introduced in the 1960s as a method to design low-precision digital circuits [6]. In stochastic computing, probabilities are encoded by random sequences of bits. Each bit in these sequences is equal to 1 with the probability to be encoded. This method has been used for iterative decoding of some error-correcting codes. Its advantage is that the decoding operation can be realized using very simple circuits working at high-speed.

The early stochastic LDPC decoders [7][8] are sensitive to the level of random switching activity within the Tanner graph. Several methods have been proposed to improve the performance of the stochastic decoding. Using Edge Memories (EMs) is a way to avoid latching problem [9], which is a major shortcoming of stochastic decoding. In this paper, we present a model to analyze the performance of EMs. This model can be used to determine the convergence and to gain a better understanding of the behavior of the decoding process. The rest of the paper is organized as follows. Section II provides an overview of stochastic computation and the stochastic decoding method. Section III describes the analysis of a Markov chain model for EMs. The simulation results are given in Section IV. Section V gives the conclusions and summaries.

II. STOCHASTIC COMPUTATION AND LDPC DECODING

In stochastic computation, the probabilities are transformed to streams of stochastic bits using Bernoulli sequences. A sequence of \( N \) bits of which \( j \) bits are equal to 1 represents a probability value of \( \frac{j}{N} \). For instance, a length 10 sequence with 8 bits equal to 1 represents a probability of 0.8. The transformation between a probability and a stochastic stream is not unique. Different stochastic streams are possible to represent a given probability. Stochastic representation and computation can be applied to probability operations in Tanner graph to replace the probabilities passed between nodes. Thus, the complex probability operations at variable-nodes and check-nodes can be performed by using simple circuits.

Here, we use a degree 3 node to show how the variable-nodes and the check-nodes exchange stochastic bits in stochastic decoding. Let \( P_a = P(a=1) \) and \( P_b = P(b=1) \) be the probabilities of two inputs, \( a \) and \( b \), of the variable-node. The output probability of the variable-node, \( P_c \), can be computed as

\[
P_c = P_a P_b [P_a P_b + (1 - P_a)(1 - P_b)]
\]

Similarly, at the check-node we have

\[
P_c = P_c (1 - P_a) + (1 - P_c) P_b
\]
In stochastic decoding, each decoding round is called a Decoding Cycle (DC) [9], which does not directly correspond to the iterations in SPA decoding. In each DC, a variable-node receives one bit from the stochastic stream corresponding to the channel probability, and the extrinsic bits from the check-nodes. Then, each variable-node propagates its outgoing 1-bit messages to the connected check-nodes. Check-nodes check the parities and send their 1-bit messages back to the variable-nodes. After completing this exchange of bits between variable-nodes and check-nodes, variable-nodes load in the next bit from the stochastic streams, and start the next decoding cycle.

When using the stochastic decoding approach, there is the possibility that the stochastic decoder is very sensitive to the level of random switching activity. Based on the structure of the variable-node, the latching problem occurs when the input bits of the variable-node are unequal for several decoding cycles. Under this condition, the output bits get stuck in the same value and the variable-node is locked into the hold state. A re-randomization mechanism is required to prevent the latching problem. In [9], Edge Memories (EMs) were introduced to avoid the latching problem.

Edge Memories (EMs)

Edge Memories [9] are $L$-bit shift registers assigned to the outgoing edges of variable-nodes. When input bits of the variable-node are in agreement, the EM is updated with the input bit. On the contrary, the EM is not updated when the variable-node is at hold state. One stochastic bit is randomly picked from the EM and passed through the edge as the outgoing bit. Thus, EMs contain only the bits which are not produced in hold state. With this updating scheme, EMs reduce the chance of latching in the LDPC decoding.

III. A MARKOV CHAIN MODEL FOR EMs

In order to understand the behavior of decoding process, we present a Markov chain model for EMs in stochastic LDPC decoder. First, we define the state of EMs based on the number of 1s in EMs. State 0 denotes the case where the bits in the EM are all-zeros. State 1 means there is only one 1 in the shift register, and state $L$ stands for the all-ones state. Thus, there will be $(L+1)$ states for the EM with length $L$.

Under the infinite codeword length and cycle-free assumption, we conclude that all EMs are identical and independent in the stochastic decoder. Figure 2 shows a Markov chain model for an EM.

Due to the updating scheme used here, an EM is only updated one bit each time. We can see that except state 0 and state $L$, all of other states have three transitional options, staying at the same state, or moving to one of the neighboring states. When the updated bit is 1 and the abandoned bit is 0, the current state, state $i$, will transit to state $(i+1)$. On the contrary, if the updated bit is 0 and the shifted out bit is 1, the state will transit to state $(i-1)$. If the variable-node is locked in the hold state or the updated bit and the abandoned bit of EM are the same, the EM will stay in the same state.

The probabilities of this model are evolved as follows. First, we initialize the EM with the probability received from channel. For each state of EM, the initial probability is corresponding to the channel probability. In each DC, the probabilities of each state are based on the input probability of the variable-node, and the channel probability. Thus, the probabilities of states at time $t$ can be represented as

$$P_t(S) = \begin{bmatrix} P_t(S_0) \\ P_t(S_1) \\ \vdots \\ P_t(S_L) \end{bmatrix}$$

and

$$P_{t-1}(S_i) = P_{t-1}(S_{i-1}) \left[ \frac{L-(i-1)}{L} P_{ch} \left( P_{in,t} \right)^{i-1} \right] + \left( \frac{i-L}{L} \right) P_{ch} \left( P_{in,t} \right)^{i-1} \left[ 1 - P_{ch} \left( 1 - P_{in,t} \right)^{i-1} \right] + \left( \frac{i+1}{L} \right) \left[ 1 - P_{ch} \left( 1 - P_{in,t} \right)^{i-1} \right]$$

for $0 \leq i \leq L, t > 0$.

Here $P_{ch}$ is the probability obtained from the channel, $P_{in,t}$ is the input probability of the variable-node at time $t$, and $d_i$ is the degree of the variable-node.

The first term on the right hand side in (4) is the probability of state transition from state $(i-1)$ to state $(i)$, the second term represents the probability of staying in the same state, and the last term is the probability of moving from state $(i+1)$ to state $(i)$.
The output probability of the variable-node is

\[ P_{out,t} = P_t(S)^T P(B_{out,t} = 1|S) \] (5)

where \( B_{out,t} \) is the outgoing stochastic bit from the variable-node to the check-node at time \( t \).

We also have

\[ P(B_{out,t} = 1|S) = \begin{bmatrix} P(B_{out,t} = 1|S_0) \\ P(B_{out,t} = 1|S_1) \\ \vdots \\ P(B_{out,t} = 1|S_L) \end{bmatrix} \] (6)

where

\[ P(B_{out,t} = 1|S_i) = \frac{i}{L} \left( 1 - P_{ch}(P_{in,t})^{k-1} - (1 - P_{ch}) (1 - P_{in,t})^{k-1} \right) \]

for \( 0 \leq i \leq L, t > 0 \). (7)

The first term in (7) is the probability that the outgoing bit from the variable-node is 1 when all the incoming bits are in disagreement and the bit randomly picked from the EM is 1. The second term is the probability that the output is 1 when all the input bits of the variable-node are 1s.

Because in stochastic decoding parity check is done at the check-nodes, the input probability at the variable-nodes is equal to

\[ P_{in,t} = \frac{1 - (L - 2 P_{out,t})^{k-1}}{2} \] (8)

where \( d_c \) is the degree of the check-node, and \( t > 0 \).

### IV. SIMULATION RESULTS

In this section we study the simulation of the Markov chain model for the EM in stochastic decoding of a \((dv, dc)=(2,3)\) LDPC code. In these simulations, all-zeros codeword is transmitted over an AWGN channel using BPSK modulation. We have selected the length of EMs to be 4 and simulations are carried out at SNR of 3 dB. The state transition matrices at different decoding cycles are shown in Table 1.

After approximately 30 decoding cycles, the state transition matrix approaches the steady state. In the stationary distribution, the transition probability from state 0 to the same state is equal to 1 indicating edge memories approaching to the 0 state, which is in agreement with transmission of the all-zeros codeword.

The stationarity of the Markov chain model is also observed from the limit

![Pin and Pout of variable-node for \((dv, dc)=(2,3)\) LDPC code at SNR=3dB](Fig. 3. Input and output probabilities of the variable-node for a \((dv, dc)=(2,3)\) LDPC code for 30 decoding cycles.)

![Table 1. The transition matrices of a length-4 Edge Memory for a \((dv, dc)=(2,3)\) LDPC code for different decoding cycles.](Tab. 1. The transition matrices of a length-4 Edge Memory for a \((dv, dc)=(2,3)\) LDPC code for different decoding cycles.)
$$\lim_{n \to \infty} M^n = \begin{bmatrix} 1 & 0 & \cdots & 0 \\ & \ddots & \ddots & \vdots \\ & & 1 & 0 & \cdots & 0 \end{bmatrix} \quad (9)$$

Figure 3 shows the input and output probabilities of the variable-node for 30 DCs. These probabilities converge to zero because the all-zeroes codeword is send over an AWGN channel. When the output probability of the variable-node converges to zero (or one), the outgoing stochastic bit from the variable-node has high probability to be 0 (or 1). After the state of the Edge Memories is converges, the output stochastic bit can be determined. Thus, the codeword can be determined and the decoding process can be terminated. This is helpful in determining the maximum number of decoding cycles of stochastic decoder and estimating the decoding time.

V. CONCLUSIONS

We presented a Markov chain model for Edge Memories in stochastic LDPC decoding and determined the state transition matrices of this model. This model can be employed to study the convergence conditions and the behavior of decoding process. This model is helpful in estimating the required number of decoding cycles in stochastic decoding of LDPC codes.

REFERENCES