Mathematical Implementation of MAX LOG MAP Algorithm for Low Power Applications in Turbo Decoders

P. Maniraj Kumar¹, Dr. S. Sutha², V. Krishna Mourthy³

Associate Professor, Dept. of ECE, PSNA College of Engineering and Technology, Dindigul, Tamilnadu, India¹
Assistant Professor, Dept of EEE, University College of Engineering, Panruti, Tamilnadu, India²
M.E Student, Dept. of ECE, PSNA College of Engineering and Technology, Dindigul, Tamilnadu, India³

ABSTRACT: In Turbo decoding the design of Log MAP algorithm is more complicated with the exponential sum. In this paper a mathematical model is designed with a jacobian logarithm which consists of a max function along with the exponential correction function. The complexity of the jacobian logarithm is also reduced by the Max Log Map algorithm by eliminating the correction term. Multistep Log Map algorithm is used to approximate the correction function. A Dual Mode (SB/DB) procedure is used to compute the log likelihood ratio in order to increase the speed of the computation. The performance of this algorithm is similar to the Log Map algorithm. This method will be incorporated into VLSI technology to design a High Speed MAP Decoder (15% to 25% as in existing) with less area Utilization (8% as in existing) and low Power (74.95mW).

KEYWORDS: MAP Algorithm, Log MAP Algorithm, Jacobian logarithm, SB/DB and Log Likelihood ratio.

I. INTRODUCTION

The MAP algorithm which is used in the turbo decoders is operated in the logarithmic domain in order to reduce the computational complexity [1]. The computation of log likelihood ratio with the values of state metrics is more complicated with log exponential sum calculation. The log exponential sum can be simplified by Jacobian algorithm by adding a correction term along with the max operator. The correction function calculation is critical because of the performance and complexity of the turbo decoder. So a lot of methods [2][3][4] are used to simplify the computation to satisfy the performance requirements.

This paper proposes an algorithm to compute the correction function that will be used to calculate the state metrics and log likelihood values. Furthermore this Log Likelihood values can be modified into SB/DB mode with a new set of equations and this will increase the performance parameters of the Turbo decoders.

The rest of the paper is organized as follows: in section II, brief introduction and derivation of basic MAP and Log MAP algorithmic equations. Section III reviews different methods of approximation of the correction function. In Section IV, the SB/DB mode calculations are presented. Section V presents the results and discussions. This paper is finally concluded in the Section VI.

II. DIFFERENT METHODS OF MAP ALGORITHM

In this section different method of MAP algorithms were discussed. The mathematical equations required for the Log Map algorithm is introduced with a max operator and the Max Log Map algorithm is also presented.

A. DIGITAL COMMUNICATION SYSTEM MODEL.

A Digital communication system is represented by figure 1. Map algorithm for the block of binary data input sequence D=D₁, D₂,...,Dₜ is represented in figure1. The encoder consists of two identical recursive systematic Convolutional...
Encoders (RSC) with M memory elements. The code rate of the encoder is \( \frac{1}{2} \) with two outputs which is the sequence of \( X_s = X_{s1}, X_{s2}...X_{sN} \) and the parity bit sequence \( X_p = X_{p1}, X_{p2}...X_{pN} \).

The corresponding received sequence from the channel with the noise is \( Y = Y_s, Y_p \) which consist of pair of received systematic and parity bits.

**B. MAXIMUM A POSTERIORI (MAP) ALGORITHM**

The objective of the MAP algorithm is to calculate Log Likelihood Ratio (LLR) \( \Lambda (D_k) \) of the a posteriori probability (APP) of \( D_k = 1 \) to \( D_k = 0 \). The state of the encoder is represented by \( S_k \) at time \( k \). The state transition is from \( k-1 \) to \( k \). The LLR value can be given by equation 1.

\[
\Lambda (D_k) = \ln \frac{\sum_{S_{k-1}} \sum_{S_k} \gamma_i (y_k, S_{k-1}, S_k) \alpha_{k-1} (S_{k-1}) \beta_k S_k}{\sum_{S_{k-1}} \sum_{S_k} \gamma_i (y_k, S_{k-1}, S_k) \alpha_{k-1} (S_{k-1}) \beta_k S_k}
\]

(1)

The forward state metric \( \alpha_k \) can be expressed as

\[
\alpha (S_k) = \sum_{S_{k-1}} \sum_{i=0}^1 \gamma_i (y_k, S_{k-1}, S_k) \alpha_{k-1} (S_{k-1})
\]

(2)

and the backward state metric \( \beta_k \) can be expressed as

\[
\beta (S_k) = \sum_{S_{k+1}} \sum_{i=0}^1 \gamma_i (y_{k+1}, S_k, S_{k+1}) \beta_{k+1} (S_{k+1}) \alpha_k (S_k)
\]

(3)

The branch transition probabilities \( \gamma_i \) can be calculated by the equation (4).

\[
\gamma_i ((y_{sk}, y_{pk}), S_{k-1}, S_k) = e^{\sqrt{\frac{y_{sk}}{L_{sk}} + \frac{y_{pk}}{L_{sk}} + L_{c} + L_{e} + L_{n}}} \]

(4)

The channel reliability value \( L_c = \frac{2}{\sigma^2} \) with \( \sigma^2 \) is the noise variance and \( L_e \) is the extrinsic information which gives a priori information.

**C. LOG MAP ALGORITHM**

The complexity in the MAP algorithm can be reduced by Log MAP algorithm. To avoid complicated calculations in the MAP algorithm, the entire MAP algorithm is calculated in Logarithmic domain. So the basic equations in the MAP algorithm (2), (3) and (4) are converted into log domain which reduces the number of multiplications and addition operations used. So, \( \bar{\alpha}_k (S_k) = \ln \alpha_k (S_k) \).
The above equation (5),(6) and (7) can be simplified into general form, which is given by the equation (8).

$$ F(x_1, x_2, x_3, \ldots, x_n) = \ln \left( \sum_{j=1}^{n} e^{x_j} \right) $$

(8)

The jacobian logarithm for two variable expressions is given by

$$ \max_x \left( x_1, x_2 \right) = \ln \left( e^{x_1} + e^{x_2} \right) = \max_x \left( x_1 + x_2 \right) + \ln \left( 1 + e^{-|x_1 - x_2|} \right) = \max_x \left( x_1, x_2 \right) + C(x) $$

(9)

The calculation of the correction term C(x) in (9) is more complicated in Log MAP algorithm. The same form of log exponential sum occurs in the calculation of backward state metrics(βk), and LLR A (Dk). So, to improve the performance of the Log Map Algorithm, a simple method of implementation of correction function must be required. The easiest method of finding out the correction function is presented in [5]. But this requires more memory to store the table sequences. A new method used in this calculation is presented in Section III.

D. MAX LOG MAP ALGORITHM

The max log map algorithm introduces an approximation \( \max^*(x1,x2) \approx \max (x1,x2) \). This Max log map algorithm simply omitting the correction term C(x). But the performance of the Log Map algorithm is dropped by 10% compared to Log Map algorithm. Even though the Max Log Map algorithm is least complex, it gives worst BER. So, in Max Log Map algorithm, a simple implementation of correction function is required to improve the performance of the turbo decoders.

III DIFFERENT ALGORITHMS FOR THE CALCULATION OF CORRECTION FUNCTION

In this section a different methods of existing algorithms which are approximates the correction function are discussed. The performance tradeoff for adding the correction elements with these algorithms are also discussed.

In the Linear Log Map algorithm proposed in [6], the MacLaurin Series expansion is used and it is observed that the correction function is approximately zero.

$$ C(x) \approx \max \left( 0, \ln 2 - \frac{1}{2} x \right) $$

(10)
This approximation offers a good result than the Constant Log Map Algorithm presented in [7] where

$$C(x) = \begin{cases} \frac{3}{8}, & -2 \leq x \leq 2 \\ 0, & \text{otherwise} \end{cases}$$  \hspace{1cm} (11)$$

An accurate method of approximation proposed in [8] is Multistep Log Map Algorithm.

$$C(x) \approx \frac{\ln 2}{2^{\lfloor x + 0.5 \rfloor}}$$  \hspace{1cm} (12)$$

This method of correction is more accurate. This method of algorithm employs with shift registers storing the ln(2). For fast computation, a speed registers are required for this algorithm. Figure 2 shows the graphical comparison of various Approximations to the correction function.

![Approximations to the Correction Functions](image)

Figure 2. Comparison of Various Correction functions

IV. DUAL MODE SINGLE BINARY/DOPBLE BINARY (SB/DB) MAP DECODING

The SB/DB Map decoding is introduced by Y.Sen [9] and C.Y Shen [10] for radix 4 Map architecture to achieve high hardware usage and shared storage information. In general, the MAP is composed of branch metrics, forward recursion state metrics, backward recursion state metrics, a priori LLR, a posteriori LLR, and extrinsic information. For the dual-mode MAP decoding, the radix-4 SB MAP decoding and radix-4 DB MAP decoding are employed because of their computational similarity. So, the equations of branch metrics, forward state metrics, backward state metrics, a priori LLR, a posteriori LLR, and extrinsic information are reformulated for SB / DB Decoding.

A) RADIX-4 SB MAP DECODING

The arithmetic operations of the of the Radix-4 SB MAP are described as

$$\alpha_k(S_k) = \max_{S_{k-1}, S_{k-2}} (\gamma_k(S_{k-2}, S_k) + \alpha_{k-2}(S_{k-2}))$$  \hspace{1cm} (13)$$
\[
\beta_k(S_k) = \text{MAX}_{s_{k+1}s_{k+2}} (\gamma'_{k+1}(S_k, S_{k+2}) + \beta_{k-2}(S_{k-2}))
\]

(14)

\[
\gamma'_{k}(S_{k-2}, S_k) = (\Lambda_{\text{apr}, k-1}(u_{k-1}) + y^s_{s(k-1)})x_{i(k-1)} + \sum_{i=1}^{m} y^p_{k-1}x^p_{k-1} + \sum_{i=1}^{m} y^p_{i}x^p_{i}
\]

(15)

\[m \text{ – number of parity bit and the sign bit of posteriori LLR value decides whether } u_k = 0 \text{ or } u_k = 1.\]  

B) RADIX-4 DB MAP DECODING

In DB mode two binary bits are encoded \(u_k = u_k^1 u_k^2\). The arithmetic operations of the of the Radix-4 DB MAP are described as

\[
\alpha_k(S_k) = \alpha_k(S_k) = \text{MAX}_{s_{k-1}} (\gamma'_{k-1}(S_{k-1}, S_k) + \alpha_{k-2}(S_{k-1}))
\]

(16)

\[
\beta_k(S_k) = \text{MAX}_{s_{k+1}} (\gamma_{k+1}(S_k, S_{k+1}) + \beta_{k+2}(S_{k+2}))
\]

(17)

\[
\gamma'_{k}(S_{k-1}, S_k) = \Lambda^{(c)}_{\text{apr}, k}(u_k = z) + 2y^s_{k}x^1_{k} + 2y^s_{k}x^2_{k} + 2\sum_{i=1}^{m} y^p_{i}x^p_{i}
\]

(18)

The reformulated equations for SB/DB are used for the calculation of Log Likelihood Ratio (LLR) calculations. By writing the MAX operations in the equations of LLR value in SB Mode, We can define \(Cs(z')\)

\[
\varepsilon_s(00) = \text{MAX}_{s_{k-2}s_k \neq k-1 = 0, s_{k-1} = 0} (\alpha_{k-2}(S_{k-2}) + \gamma'_{k}(S_{k-2}, S_k) + \beta_{k}(S_{k}))
\]

(19)

\[
\varepsilon_s(01) = \text{MAX}_{s_{k-2}s_k \neq k-1 = 0, s_{k-1} = 1} (\alpha_{k-2}(S_{k-2}) + \gamma'_{k}(S_{k-2}, S_k) + \beta_{k}(S_{k}))
\]

(20)

\[
\varepsilon_s(10) = \text{MAX}_{s_{k-2}s_k \neq k-1 = 1, s_{k-1} = 0} (\alpha_{k-2}(S_{k-2}) + \gamma'_{k}(S_{k-2}, S_k) + \beta_{k}(S_{k}))
\]

(21)

\[
\varepsilon_s(11) = \text{MAX}_{s_{k-2}s_k \neq k-1 = 1, s_{k-1} = 1} (\alpha_{k-2}(S_{k-2}) + \gamma'_{k}(S_{k-2}, S_k) + \beta_{k}(S_{k}))
\]

(22)

Where \(z' = (u_k-1, u_k)\)

Therefore the LLR value can be defined by

\[
\Lambda(D_{z'}(u_{k-1})) = \text{MAX}(\varepsilon_s(10), \varepsilon_s(11)) - \text{MAX}(\varepsilon_s(00), \varepsilon_s(01))
\]

(23)

\[
\Lambda(D_{z'}(u_k)) = \text{MAX}(\varepsilon_s(01), \varepsilon_s(11)) - \text{MAX}(\varepsilon_s(00), \varepsilon_s(10))
\]

(24)

By writing the MAX operations in the equations of LLR value in DB Mode, We can define \(Cd(z')\)

\[
\varepsilon_d(00) = \text{MAX}_{s_{k-1}s_k \neq k = 00} (\alpha_{k-1}(S_{k-1}) + \gamma_{k}(S_{k-1}, S_k) + \beta_{k}(S_{k}))
\]

(25)

\[
\varepsilon_d(01) = \text{MAX}_{s_{k-1}s_k \neq k = 01} (\alpha_{k-1}(S_{k-1}) + \gamma_{k}(S_{k-1}, S_k) + \beta_{k}(S_{k}))
\]

(26)

\[
\varepsilon_d(10) = \text{MAX}_{s_{k-1}s_k \neq k = 10} (\alpha_{k-1}(S_{k-1}) + \gamma_{k}(S_{k-1}, S_k) + \beta_{k}(S_{k}))
\]

(27)
\[
\begin{align*}
\varepsilon_d (11) &= \max_{s_{k-1}, s_k : \alpha_k = 11} (\alpha_{k-1} (S_{k-1}) + \gamma_k (S_{k-1}, S_k) + \beta_k (S_k)) \\
\end{align*}
\]

(28)

Where \( z' = u_k \)
Therefore the LLR value can be defined by

\[
\begin{align*}
\Lambda^0 (D_k (u_k)) &= \varepsilon_d (00) \\
\Lambda^1 (D_k (u_k)) &= \varepsilon_d (10) - \varepsilon_d (00) \\
\Lambda^{10} (D_k (u_k)) &= \varepsilon_d (11) - \varepsilon_d (00) \\
\end{align*}
\]

(29)  (30)  (31)  (32)

By comparing the LLR values of both SB & DB the MAX operations of \( \varepsilon_s (z') \) & \( \varepsilon_d (z') \) are same. Therefore the hardware can be also shared. So, the implementation of this LLR unit in the VLSI tool, the area of utilization is also reduced up to 8% as existing.

V. RESULTS AND DISCUSSIONS

A mathematical model is designed with a jacobian logarithm which consists of a max function along with the exponential correction function. The complexity of the jacobian logarithm is also reduced by the Max Log Map algorithm by eliminating the correction term. Multistep Log Map algorithm is used to approximate the correction function. By this model of correction function the BER of the turbo decoder is greatly reduced. This is shown in the figure 3. A Dual Mode (SB/DB) procedure is used to compute the log likelihood ratio in order to increase the speed of the computation. This method will be incorporated into VLSI technology to design a High Speed MAP Decoder (15% to 25% as in existing) with less area Utilization (8% as in existing) and low Power (74.95mW).

Fig. 7. Comparison of BER for Various Correction functions

The proposed decoder can be designed using Verilog HDL, a descriptive hardware language for architectural module design. This designed architectural module is simulated using Modelsim 5.5e simulating tool, synthesized using Xilinx Project Navigator 10.2i synthesis tool and tested on a Spartan3E family device XC3S500E. This proposed scheme utilized 1972 LUTs and 476 FFs at a maximum frequency of 125MHz. The proposed work results shows that the pipelined metric MAP decoding system incorporated with SB/DB mode leads to lower power consumption in terms of slices, Look Up Tables and Flip Flops.
Table I

<table>
<thead>
<tr>
<th>FPGA Family</th>
<th>Device Specifications</th>
<th>Power Consumption</th>
</tr>
</thead>
<tbody>
<tr>
<td>Spartan-3E</td>
<td>Xc3s1200E</td>
<td>158.9mw</td>
</tr>
<tr>
<td>Spartan-3E</td>
<td>Xc3s500E</td>
<td>81.8mw</td>
</tr>
<tr>
<td>Spartan-3E</td>
<td>Xc3s250E</td>
<td>52.27mw</td>
</tr>
</tbody>
</table>

The various devices in the Spartan-3 family are tested against their frequency and power and tabulated in Table 1.

Table 2 illustrates the investigation parameters, such as number of slices, LUTs, etc. required in SB and in DB mode. The values are graphically plotted and depicted in Fig. 9.

Table II

<table>
<thead>
<tr>
<th>Parameters</th>
<th>SB Mode</th>
<th>DB Mode</th>
</tr>
</thead>
<tbody>
<tr>
<td>Slices</td>
<td>128</td>
<td>98</td>
</tr>
<tr>
<td>LUTs</td>
<td>72</td>
<td>64</td>
</tr>
<tr>
<td>Gate counts</td>
<td>1800</td>
<td>1785</td>
</tr>
<tr>
<td>IOBs</td>
<td>21</td>
<td>21</td>
</tr>
</tbody>
</table>
The optimum performance of Log Map Algorithm involves complex operations. By neglecting the correction function in the Max Log Map algorithm gives the capacitive loss and hence the correction term must be added which gives the optimal Log Map algorithm. The shared dual mode MAP decoder achieves low computational costs and low storages. This MAP processor achieves high throughput rates and high area efficiency.

VI. CONCLUSION

REFERENCES


