Multiple Bit Release Sliding Window Turbo Decoding

Yassir Nawaz, R.Venkatesan and Paul Gillard

Faculty of Engineering and Applied Science, Memorial University of Newfoundland
St. John's, NF, CANADA A1B 3X5
E-mail: {yassir, venky}@engr.mun.ca, paul@cs.mun.ca

Abstract: *Sliding window implementations of decoding algorithms are used to reduce the memory requirements in turbo decoders. In this paper we propose several modifications to the conventional sliding window implementations of SOVA, bi-directional SOVA and Max-Log-MAP based turbo decoders. The proposed modifications allow multiple bits to be released in a single decoding window thus reducing the computational complexity and increasing the decoding speed of turbo decoders. These improvements in speed and complexity are achieved without any performance degradation and at the expense of a modest increase in hardware.*

Keywords: Sliding window, finite length decoding, turbo codes, SOVA, MAP

1. INTRODUCTION

Turbo codes introduced in [1], can achieve performance close to Shannon limit for very long frame lengths. A turbo encoder consists of two parallel concatenated recursive convolutional (RSC) encoders that are connected by an interleaver. Iterative decoding is employed in the decoder where each component decoder has the ability to use a priori information as well as provide reliability information for each decoded information bit. The two main algorithms used in turbo decoding are the Soft Output Viterbi Algorithm (SOVA) [2] and Maximum a posteriori (MAP) [3] algorithm. Several algorithms based on SOVA and MAP have been introduced to reduce the complexity and in some cases to increase the performance of turbo decoders, the most significant being modified SOVA [4], bi-directional SOVA [5], Max-Log-MAP [6] and Log-MAP [7] algorithms.

The performance of turbo codes depends heavily on frame length and deteriorates rapidly as the frame length decreases. A long frame length, however, means a long decoding trellis for which the memory requirements as well as decoder complexity are excessive from an implementation view point. Sliding window or finite-length window decoding can significantly reduce the memory requirements and the complexity of the decoder. Sliding window implementations of both SOVA and MAP algorithms reduce the decision depth of the trellis to around five times the encoder constraint length which eliminates the need to store the trellis for the entire frame in memory.

In this paper we present single and multiple-bit release sliding window decoding of SOVA, bi-directional SOVA and Max-Log-MAP algorithms along with their performance comparison. Section 2 defines some basic parameters and explains conventional single-bit release sliding window decoding for the above algorithms. The proposed modifications and multiple-bit release sliding window decoding is explained in section 3. Section 4 compares the multiple-bit release algorithms presented in section 3 and shows that multiple-bit release bi-directional SOVA significantly outperforms Max-Log-MAP for sliding window implementation.

2. SLIDING WINDOW DECODING

In this section we explain our implementation of generalized sliding window decoding for SOVA and MAP algorithms. In trellis based decoding, the number of trellis stages required to make reliable decisions determines the length of the decoding window and is referred to as the decision depth *D*. In both SOVA and MAP algorithms this decision depth is about five times the constraint length of the component convolutional encoder. A generalized sliding window decoding process is shown graphically in Figure 1. After the bits decoded in a window are released, next stage of the trellis is built and the window slides forward.

2.1 SOVA and Bi-directional SOVA

To explain sliding window SOVA algorithm we define the following terms.

D_{SOVA} Decision depth of trellis for SOVA.

T_{SOVA} Traceback depth of trellis for SOVA. T_{SOVA} is the total number of trellis stages, where the discarded path merging with ML path is considered to find the reliability value of decoded bit.

For single bit release SOVA, forward recursion starts by building the first D_{SOVA} stages of the trellis. This is followed by SOVA traceback at each stage of the trellis in the current window. T_{SOVA} in this case equals D_{SOVA}. The decoded bit at the first stage of the trellis is released and the decoding window slides forward by one trellis stage. The decoded bit at the

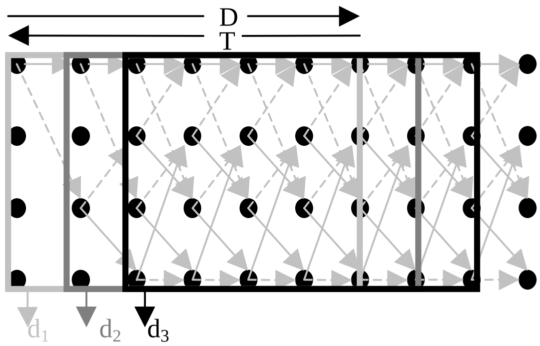

Fig 1: One bit release sliding window decoding

second trellis stage is released in this window followed by another slide of the window and so on. Decoding of bi-directional SOVA is the same as simple SOVA except that sliding window in backward SOVA starts from the last stage in the trellis and moves in the opposite direction, thus releasing the bits in reverse order.

2.2 MAP and Max-Log-MAP

The parameters for MAP and Max-Log-MAP algorithms are defined as follows.

$\boldsymbol{D}_{\boldsymbol{MAP}}$ Same as $\boldsymbol{D}_{\boldsymbol{SOVA}}$

$\boldsymbol{T}_{\boldsymbol{MAP}}$ Number of trellis stages in the backward recursion which is the same as $\boldsymbol{D}_{\boldsymbol{MAP}}$ or $\boldsymbol{D}_{\boldsymbol{SOVA}}$.

Forward recursion in MAP algorithm is similar to the forward recursion in SOVA, the only difference being the actual calculation of path metrics. Max-Log-MAP however has forward recursion equivalent to that of SOVA. The forward recursion in both MAP and Max-Log-MAP is followed by a backward recursion instead of a traceback as was the case in SOVA. This backward recursion is identical to the forward recursion but proceeds from last stage in the decoding window to the first stage. $\boldsymbol{T}_{\boldsymbol{MAP}}$ therefore is always the same as $\boldsymbol{D}_{\boldsymbol{MAP}}$ in sliding window decoding of MAP and Max-Log-MAP. After the release of the decoded bit, the window slides forward in the same manner as explained previously for SOVA.

The bit error rate (BER) performance comparison of sliding window single-bit release SOVA, bi-directional SOVA and Max-Log-MAP is presented in Figure 2. A Turbo encoder with pseudorandom interleaver and RSC component encoders of constraint length 3 has been used. The value of $\boldsymbol{D}_{\boldsymbol{SOVA}}$ and $\boldsymbol{T}_{\boldsymbol{SOVA}}$ for the component decoders is 15. While both Max-Log-MAP and bi-directional SOVA are better than simple SOVA, it is interesting to note that bi-directional SOVA is consistently better than Max-Log-MAP. Similar results were also reported for normal or full length decoding of SOVA and Max-Log-MAP in [5].

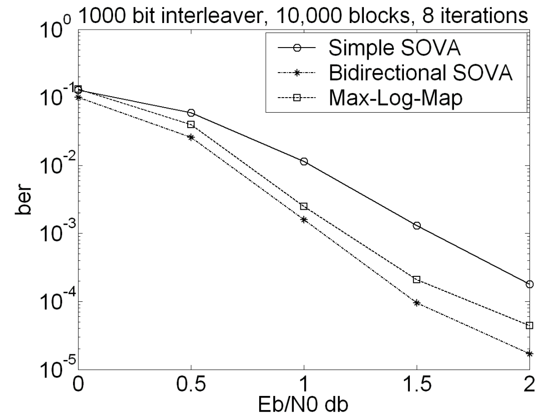

Fig 2: BER performance comparison of Simple SOVA, bi-directional SOVA and Max-Log-MAP

3. MULTIPLE BIT RELSEASE SLIDING WINDOW DECODING

In this section we propose modifications to the generalized sliding window implementation of SOVA and MAP described in section II. These modifications allow multiple bits to be released in one decoding window.

3.1 SOVA and bi-directional SOVA

In order to facilitate the release of N bits in one decoding window we propose the following modifications.

1. Increase the decision depth of trellis by N-1

$$\boldsymbol{D}_{mult\_SOVA} = \boldsymbol{D}_{SOVA} + (N\text{-}1)$$

   where $1 \leq N \leq \boldsymbol{D}_{SOVA}$

2. Keep $\boldsymbol{T}_{mult\_SOVA}$ same as $\boldsymbol{T}_{SOVA}$ and use the same ML and discarded paths in the decoding of all the N bits in a decoding window.
3. After N bits in a window have been decoded, slide the window forward by N trellis stages.

Figure 3 shows the N bit release sliding window decoding graphically. The decoding process begins with a forward recursion which builds $\boldsymbol{D}_{mult\_SOVA}$ trellis stages and finds the ML path. This is followed by the SOVA traceback in which $\boldsymbol{T}_{mult\_SOVA}$ discarded paths, one at each stage in the SOVA traceback depth are considered. The ML path and the discarded path at each trellis stage in the SOVA traceback are not only used to calculate the soft value of the decoded bit at the first stage of the trellis but also for the decoded bits at trellis stages k=n, where $2 \leq n \leq N$. Once the SOVA traceback is complete and N bits are released, the window slides forward by N trellis stages.

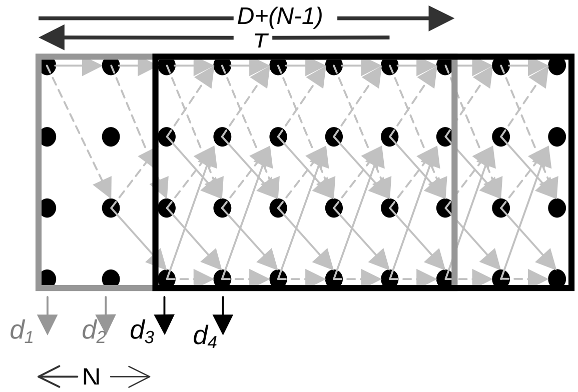

Fig 3: Multiple bit release sliding window decoding

3.1.1 The Effect of Proposed Modifications on Decoder Complexity and Performance

The number of computations in the forward recursion remains unaffected by the modifications. However, the number of SOVA tracebacks is reduced by a factor of N which implies fewer computations and an overall increase in decoder speed. This increase comes at the expense of additional hardware i.e. memory to store elongated trellis and logic to enable the release of N bits simultaneously. The amount of additional hardware required is proportional to N.

In order to determine the effect of proposed modifications on the decoder performance, we analyze how they will affect the reliability of individual bits released in a decoding window. An increase in the trellis decision depth implies more reliable ML paths for the first N-1 bits in the window. The ML path for the N^{th} bit has the same length as in one bit release implementation and therefore its reliability is unaffected. On the other hand, keeping SOVA traceback length unchanged means that all but first bit in the decoding window now have a reduced traceback. A reduced traceback implies fewer discarded or alternate paths available in the decoding process. It is clear from the above discussion that while first modification tends to increase the reliability of individual bits the later has the opposite effect. However the two effects are not uniform and decoded bits at different positions are affected differently. These differences can be exploited to increase the overall performance of a turbo decoder.

Figure 4 shows the performance of multiple bit release sliding window decoding for Simple SOVA. It can be seen that the performance of the turbo decoder improves as we increase N from 1 to 4. For N=8 the performance is approximately the same as for N=1. However, for the extreme case of N=15 performance degrades significantly. Similar results for bi-directional SOVA are presented in Figure 5. In this case the decoder performance for N=8 is also consistently better than N=1. Interestingly even the 15 bit (i.e. all the bits in the traceback depth) release

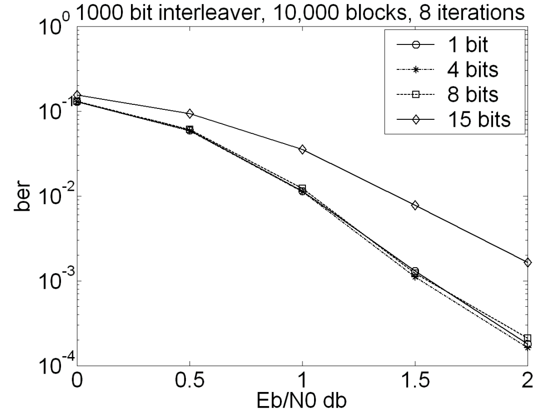

Fig 4: BER performance comparison of multiple bit release sliding window Simple SOVA

sliding window bi-directional SOVA performs as well as 1 bit release bi-directional SOVA.

3.2 MAP and Max-Log-MAP

A multiple bit release sliding window MAP and Max-Log-MAP can be implemented in a fashion similar to that of SOVA explained above. The key difference is the length of backward recursion which in the case of MAP and Max-Log-MAP is the same as forward recursion i.e. $\boldsymbol{D_{mult\_MAP}} = \boldsymbol{T_{mult\_MAP}}$.

3.2.2 The Effect of Proposed Modifications on Decoder Complexity and Performance

The number of computations in the forward recursion of MAP and Max-Log-MAP remains unaffected by the modifications. The number of computations in the backward recursion increases in each window however, the number of total windows is reduced by a factor of N, leading to an overall reduction in computations.

The performance of the multiple bit release sliding window Max-Log-MAP algorithm is shown in Figure 6. Increase in the length of both forward and backward recursions increases the reliability of decoded bits resulting in a consistent improvement in performance with an increase in N.

4. COMPARISON OF MAX-LOG-MAP AND BI-DIRECTIONAL SOVA

The forward recursion in Max-Log-MAP and bi-directional SOVA is identical but the traceback depth in bi-directional SOVA is either less than or equal to the backward recursion depth in Max-Log-MAP. Furthermore SOVA traceback is much faster than Max-Log-MAP backward recursion which suggests that bi-directional SOVA can achieve faster decoding speeds when compared to the Max-Log-MAP. The BER performance comparison of multiple bit release

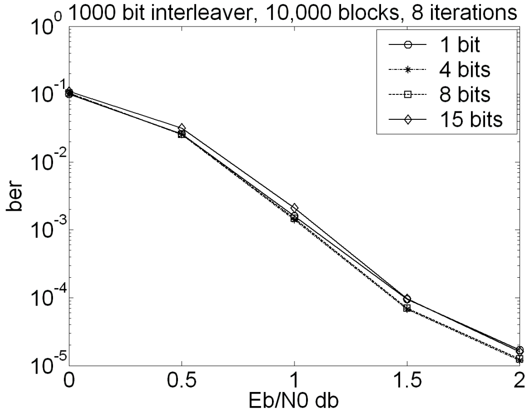

Fig 5: BER performance comparison of multiple bit release sliding window bi-directional SOVA

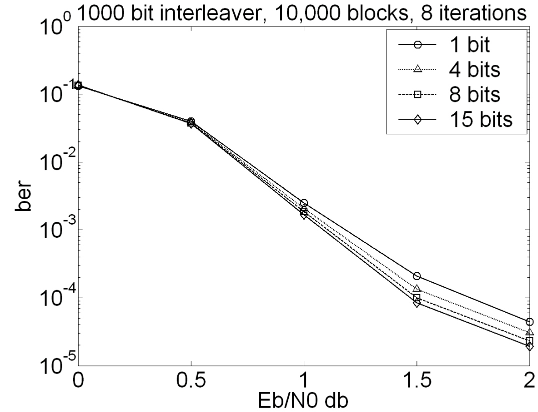

Fig 6: BER performance comparison of multiple bit release sliding window Max-Log-MAP

sliding window Max-Log-MAP and bi-directional SOVA for N=8 and N=16 is presented in Figure 7 and Figure 8. Bi-directional SOVA consistently outperforms Max-Log MAP for N=8 and both implementations have a similar performance for N=15.

5. CONCLUSION

In this paper we have presented modifications to the conventional one bit release sliding window decoding of SOVA and MAP to facilitate the release of multiple bits in a decoding window. The results obtained argue strongly in favor of multiple bit release implementations due to their reduced computational complexity, improved performance and faster decoding speed.

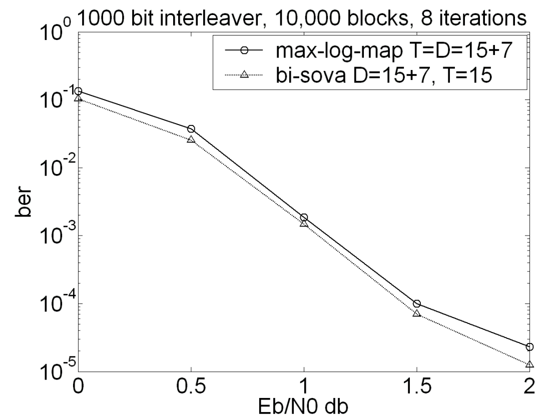

Fig 7: BER performance comparison of eight bit release sliding window bi-directional SOVA and Max-Log-MAP

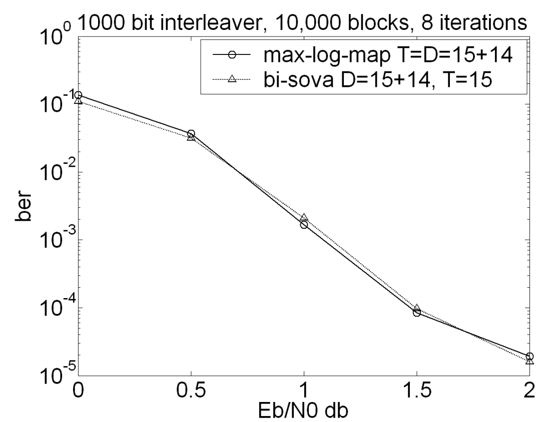

Fig 8: BER performance comparison of fifteen bit release sliding window bi-directional SOVA and Max-Log-MAP

REFERENCES

[1] C.Berrou and A.Glavieux, "Near optimum error-correcting coding and decoding. Turbo-codes," *IEEE Trans. Commun.*, vol. 44, no. 10, pp. 1261-1271, 1996.

[2] J.Hagenauer and P.Hoeher, "A Viterbi algorithm with soft-decision outputs and its applications," *IEEE Globecom*, pp.1680-1686, 1989.

[3] L.R.Bahl, J.Cocke, F.Jelinek, and J.Raviv, "Optimal decoding of linear codes for minimizing symbol error rate," *IEEE trans. Inform. Theory*, vol. VOL. NO?, pp.284-287, Mar.1974.

[4] M.P.C.Fossorier, F.Burkert, S.Lin, and J.Hagenauer, "On the equivalence between SOVA and Max-Log-MAP decodings," *IEEE Comm. Letters*, vol. 2, no.5, pp.137-139, May. 1998.

[5] J.Chen, M.Fossorier, S.Lin and C.Xu, "Bi-directional SOVA decoding for Turbo-codes," *IEEE Commun. Letters*, vol. CL-4, pp.405-407, Dec. 2000.

[6] J.A.Erfanian, S.Pasupathy, and G.Gulak, "Reduced complexity symbol detectors with parallel structures for ISI channels," *IEEE trans. Commun.*, vol.42, pp. 1661-1671, 1994.

[7] P.Robertson, E.Villebrun, and P.Hoeher, "A comparison of optimal and sub-optimal MAP decoding algorithms operating in the log domain," in *Proc. Int. Conf. Communications*, pp. 1009-1013, June. 1995.