http://ansinet.com/itj

ITJ

ISSN 1812-5638

INFORMATION TECHNOLOGY JOURNAL

ANSInet
Asian Network for Scientific Information
308 Lasani Town, Sargodha Road, Faisalabad - Pakistan

Information Technology Journal 12 (22): 7056-7060, 2013
ISSN 1812-5638 / DOI: 10.3923/itj.2013.7056.7060

A Decoding Algorithm for Data Service in HAPS Communication System

Mingxiang Guan and Le Wang
School of Electronic Communication Technology,
Shenzhen Institute of Information Technology, Shenzhen, 518172, China

Abstract: An information system formed by HAP (High Altitude Platform) will be a new generation-system for the wireless communications and HAPS (HAP Station) communication system combines the advantages of both terrestrial and satellite communication systems and avoids, to different extents, their disadvantages. Turbo code has been shown to have ability to achieve performance that is close to Shannon limit. It has been adopted by various commercial communication systems. Both universal mobile telecommunications system (UMTS) TDD and FDD have also employed turbo code as the error correction coding scheme. It outperforms convolutional code in large block size but because of its time delay, it is often only used in the non-real-time service. In this study, we discuss the encoder and decoder structure of turbo code in mobile communication System. In addition, various decoding techniques, such as the Log-MAP, Max-log-MAP and SOVA algorithm for non-real-time service are deduced and compared.

Key words: High altitude platform, decoder, beyond 3G mobile communication system, decoding algorithm

INTRODUCTION

Mobile communication technologies have rapidly developed in the past two decades. The fundamental problem of mobile communication lies on finding ways to enable people to communicate by all means whenever and everywhere, including mobile communication. Recently, a novel form of mobile communication has emerged and is called the High-altitude Platform Station (HAPS) (Li *et al*., 2011; Liu *et al*., 2009). In general, a HAPS network is composed of quasi-static high-altitude platform stations at the low altitude (20-100 km), with a certain payload and with the residence time of 5-10 years. In the near-Earth space, HAPS adopts a stable communication platform as a microwave relay station and forms into a communication system with ground control units, access equipment and various wireless users. A turbo code can be thought as a refinement of the concatenated encoding structure and an iterative algorithm for decoding the associated code sequence (Pietrobon, 1998). The codes are constructed by applying two or more component codes to different interleaved versions of the same information sequence) (Kaza and Chakrabarti, 2004; Hagh *et al*., 2006). For any single traditional code, the final step at the decoder yields hard-decision decoded bits (or, more generally, decoded symbols). In order for a concatenated scheme such as a turbo code to work properly, the decoding algorithm should not limit itself to pass hard decisions among the decoders (Lee and Park, 2006; Boutillon *et al*., 2003). To best exploit the information learned from each decoder, the decoding algorithm must effect an exchange of soft rather than hard decisions (Yeo and Anantharam, 2003). For a system with two component codes, the concept behind turbo decoding is to pass soft decisions from the output of one decoder to the input of the other and to iterate this process several times to produce better decisions. For a system with two component codes, the concept behind turbo decoding is to pass soft decisions from the output of one decoder to the input of the other and to iterate this process several times to produce better decisions. Figure 1 shows high level block diagram of turbo encoder and decoder which are used in simulation.

CODE THEORY

The mathematical foundations of hypothesis testing rests on Bayes' theorem which is derived from the relationship between the conditional and joint probability of events A and B, as follows:

$$P(A\,|\,B)P(B) = P(B\,|\,A)P(A) = P(A,B) \tag{1}$$

A statement of the theorem yields the a posteriori probability (APP), denoted $P(A|B)$:

$$P(A\,|\,B) = \frac{P(B\,|\,A)P(A)}{P(B)} \tag{2}$$

Corresponding Author: Mingxiang Guan, School of Electronic Communication Technology, Shenzhen Institute of Information Technology, Shenzhen, 518172, China

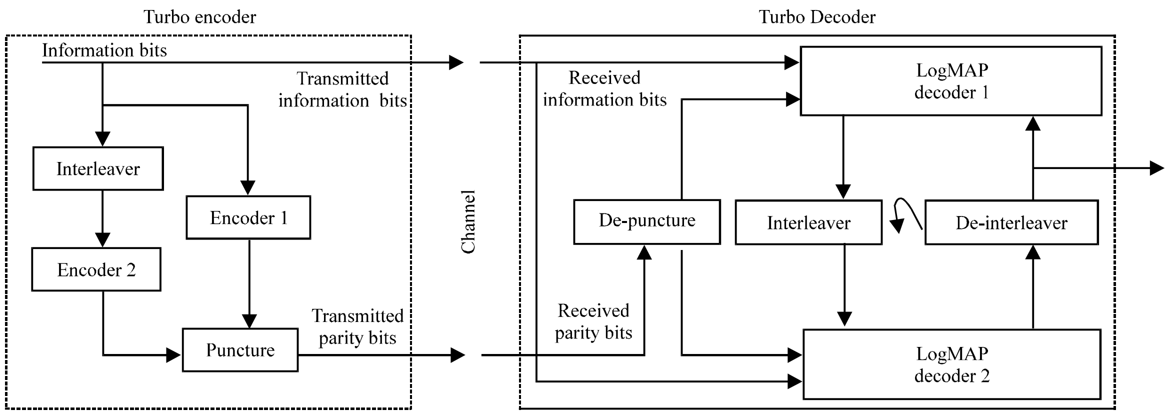

Fig. 1: Turbo decoder schemes

Which allows us to infer the APP of an event A conditioned on B, from the conditional probability, P(B|A) and the a priori probabilities, P(A) and P(B). For communications engineering applications having additive white Gaussian noise (AWGN) in the channel[10], the most useful form of Bayes' theorem expresses the APP in terms of a continuous-valued random variable x in the following form:

$$P(d=i\,|\,x)=\frac{P(x\,|\,d=i)P(d=i)}{P(x)}\quad i=1,\cdots,M \tag{3}$$

And:

$$P(x)=\sum_{i=1}^{M}P(x\,|\,d=i)P(d=i) \tag{4}$$

where, d = i represents data d belonging to the ith signal class from a set of M classes and p(x| d = i) represents the probability density function (pdf) of a received continuous-valued data-plus-noise signal, x, conditioned on the signal class d = i, p(x) is the pdf of the received signal x over the entire space of signal classes. In Eq. 3, for a particular received signal, p(x) is a scaling factor since it has the same value for each class. Lower case p is used to designate the pdf of a continuous-valued signal and upper case P is used to designate probability (a priori and APP). Equation 3 can be thought of as the result of an experiment involving a received signal and some statistical knowledge of the signal classes to which the signal may belong. The probability of occurrence of the ith signal class P(d = i), before the experiment, is the a priori probability. As a result of examining a particular received signal, we can compute the APP, P(d = i|x) which can be thought of as a "refinement" of our prior knowledge.

Maximum a Posteriori (MAP) rule takes into account the a priori probabilities, as seen below in Eq. 6. The MAP rule is expressed in terms of APPs as follows:

$$P(d=+1\,|\,x)\underset{H_2}{\overset{H_1}{\gtrless}}P(d=-1\,|\,x) \tag{5}$$

Equation 5 states that one should choose the hypothesis H_1 (d = +1) if the APP P(d = +1| x) is greater than the APP P(d =-1| x). Otherwise, choose hypothesis H_2 (d = -1). Using the Bayes' theorem in Eq. 3, the APPs in Eq. 5 can be replaced by their equivalent expressions, yielding:

$$p(x\,|\,d=+1)P(d=+1)\underset{H_2}{\overset{H_1}{\gtrless}}p(x\,|\,d=-1)P(d=-1) \tag{6}$$

Equation 6 is generally expressed in terms of a ratio, called the likelihood ratio test, as follows:

$$\frac{p(x\,|\,d=+1)}{p(x\,|\,d=-1)}\underset{H_2}{\overset{H_1}{\gtrless}}\frac{P(d=-1)}{P(d=+1)} \tag{7}$$

By taking the logarithm of the likelihood ratio in Eq. 7, we obtain a useful metric called the Log-likelihood Ratio (LLR). It is the real number representing a soft decision out of a detector, designated L(d |x), as follows:

$$\begin{aligned}L(d\,|\,x)&=\log\Big[\frac{p(d=+1\,|\,x)}{p(d=-1\,|\,x)}\Big]\\&=\log\Big[\frac{p(x\,|\,d=+1)P(d=+1)}{p(x\,|\,d=-1)P(d=-1)}\Big]\end{aligned} \tag{8}$$

$$L(d \mid x) = \log\left[\frac{p(x \mid d=+1)}{p(x \mid d=-1)}\right] + \log\left[\frac{P(d=+1)}{P(d=-1)}\right] \tag{9}$$

$$L(d \mid x) = L(x \mid d) + L(d) \tag{10}$$

where, L(x|d) is the LLR of the channel measurements of x under the alternate conditions that d = +1 or d = -1may have been transmitted and L(d) is the a priori LLR of the data bit d. To simplify the notation, we represent Eq.10 as follows:

$$L'(\hat{d}) = L_c(x) + L(d) \tag{11}$$

where, the notation L_c (x) emphasizes that this LLR term is the result of a channel measurement made at the detector. Equations 3 through 11 were developed with only a data detector in mind. Next, the introduction of a decoder will typically yield decision-making benefits. For a systematic code, it can be shown that the LLR (soft output) $L(\hat{d})$ out of the decoder is equal to:

$$L(\hat{d}) = L'(\hat{d}) + L_e(\hat{d}) \tag{12}$$

where, $L'(\hat{d})$ is the LLR of a data bit out of the detector (input to the decoder) and $L_e(\hat{d})$ called the extrinsic LLR, represents extra knowledge that is gleaned from the decoding process. The output sequence of a systematic decoder is wade up of values representing data and parity. Equation 12 partitions the decoder LLR into the data portion represented by the detector measurement and the extrinsic portion represented by the decoder contribution due to parity. From Eqs.11 and 12, we write:

$$L(\hat{d}) = L_c(x) + L(d) + L_e(\hat{d}) \tag{13}$$

The soft decision $L(\hat{d})$ is a real number that provides a hard decision as well as the reliability of that decision. The sign of $L(\hat{d})$ denotes the hard decision; that is, for positive values of $L(\hat{d})$ decide +1, for negative values decide -1. The magnitude of $L(\hat{d})$ denotes the reliability of that decision.

ITERATIVE DECODER DESIGN

For the first decoding iteration of the soft input/soft output decoder in Fig. 2, one generally assumes the binary data to be equally likely, yielding an initial a priori LLR value of L(d) = 0 for the third term in [8]. The channel pre-detection LLR value, $L_c(x)$, is measured by forming the logarithm of the ratio of λ_1 and λ_2, seen in Fig. 2. The output $L(\hat{d})$ of the Fig. 2 decoder is made up of the LLR

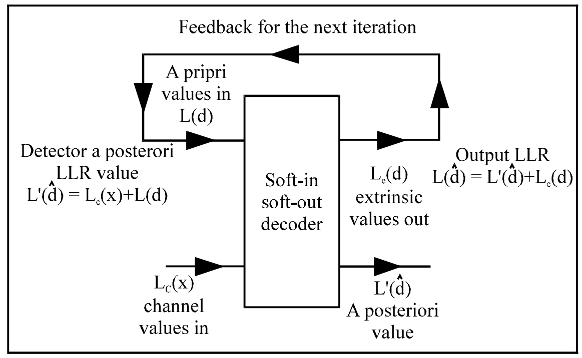

Fig. 2: Soft input/soft output decoder

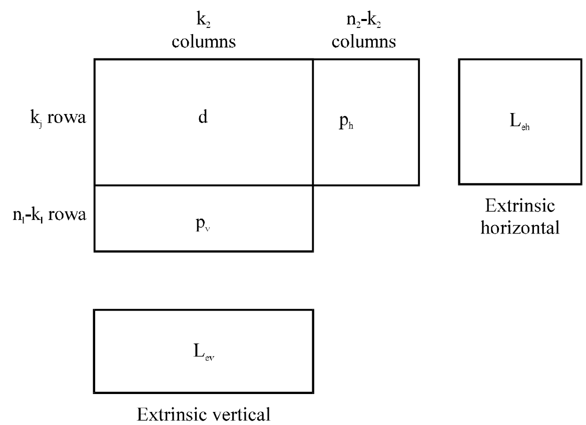

Fig. 3: Two-dimensional product code

from the detector, $L'(\hat{d})$ and the extrinsic LLR output. $L_e(\hat{d})$, representing knowledge gleaned from the decoding process. As illustrated in Fig. 3, for iterative decoding the extrinsic likelihood is fed back to the decoder input, to serve as a refinement of the a priori value for the next iteration.

Consider the two-dimensional code (product code) depicted in Fig. 3. The configuration can be described as a data array made up of k_1 rows and k_2 columns. Each of the k_1 rows contains a code vector made up of k_2 data bits and n_2-k_2 parity bits. Similarly, each of the k_2 columns contains a code vector made up of k_1 data bits and n_1-k_1 parity bits. The various portions of the structure are labeled d for data, p_h for horizontal parity (along the rows) and p_v for vertical parity (along the columns). Additionally, there areblocks labeled L_{eh} and L_{ev} which house the extrinsic LLR values learned from the horizontal and vertical decoding steps, respectively. Notice that this product code is a simple example of a concatenated code. Its structure encompasses two separate encoding steps, horizontal and vertical. The iterative decoding algorithm for this product code proceeds as follows:

- Set the a priori information

$$L(d) = 0 \tag{14}$$

- Decode horizontally, obtain the horizontal extrinsic information as shown below:

$$L_{eh}(\hat{d}) = L(\hat{d}) - L_c(x) - L(d) \tag{15}$$

- Set:

$$L(d) = L_{eh}(\hat{d}) \tag{16}$$

- Decode vertically, obtain the vertical extrinsic information as shown below:

$$L_{ev}(\hat{d}) = L(\hat{d}) - L_c(x) - L(d) \tag{17}$$

- Set:

$$L(d) = L_{ev}(\hat{d}) \tag{18}$$

- If there have been enough iterations to yield a reliable 7 decision, go to step 7; otherwise, go to step 2
- The soft output is:

$$L(\hat{d}) = L_c(x) + L_{eh}(\hat{d}) + L_{ev}(\hat{d}) \tag{19}$$

PERFORMANCE OF THE DECODING ALGORITHMS

SOVA the ML path is found in Table 1. The recursion used is identical to the one used for calculating of α in Log-MAP algorithm. Along the ML path hard decision on the bit u_k is made. $L(u_k|y)$ is the minimum metric difference between the ML path and the path that merges with ML path and is generated with different bit value u_k. In $L(u_k|y)$ calculations accordingly to Log-MAP one path is ML path and other is the most likely path that gives the different u_k. In SOVA the difference is calculated between the ML and the most likely path that merges with ML path and gives different u_k. This path but the other may not be the most likely one for giving different u_k. The output of SOVA just more noisy compared to Log-MAP output (SOVA does not have bias). The SOVA and Log-MAP have the same output. The magnitude of the soft decisions of SOVA will either be identical of higher than those of Log-MAP. If the most likely path that gives the different hard decision for u_k, has survived and merges with ML path the two algorithms are identical. If that path does not survive the path on what different u_k is made is less likely than the path which should have been used.

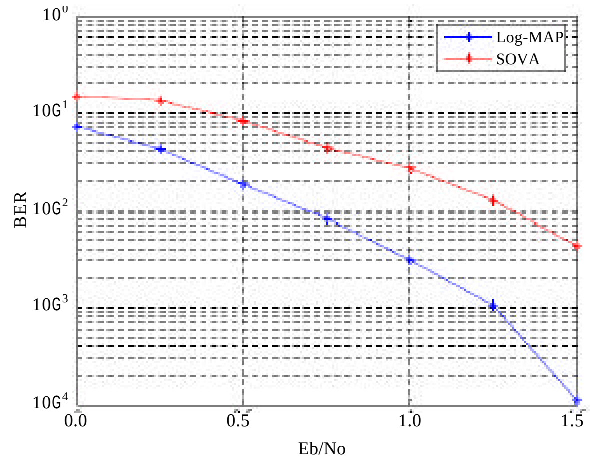

Fig. 4: BER performance of Log-MAP algorithm VS SOVA algorithm

Table 1: Comparison of complexity of different decoding algorithms

Operations	MAP	Log-MAP	SOVA
additions	$4\times2^M+6$	$12\times2^M+6$	$4\times2^M+9$
max-ops		$4\times2^M-2$	$2\times2^M-1$
multiplications	$10\times2^M+8$	8	4
look-ups	4(exp)	$4\times2^M-2$	

The forward recursion in Log-MAP and SOVA is identical but the trace-back depth in SOVA is either less than or equal to the backward recursion depth. Log-MAP is the slowest of the three algorithms but have the best performance among these three algorithms. In our TD-CDMA simulation, we implemented the Log-MAP and SOVA decoder to get best performance (with Log-MAP) or fastest speed (with SOVA). Figure 4 gives the performance comparison between Log-MAP and SOVA and it shows that Log-MAP is 1.2 dB better than SOVA at the BER of 10^{-2}, with the code block size of 260 bits and 7 iterations.

CONCLUSION

This study illustrate the turbo decoder principle and the derivation of Log-MAP and SOVA algorithms. Log-MAP algorithm is shown to achieve the best performance with good complexity tradeoff. SOVA algorithm has less computation complexity with about 1.2 dB performance degradation compared with Log-MAP at the BER of 10^{-2}. We also demonstrate that the performance of turbo code is directly proportional to the interleaver size and number of iteration in the turbo decoder. Finally, it is also shown that the performance is affected by the scale of the input soft bits power but the effect is negligible when the scaling factor is larger than 0.8.

ACKNOWLEDGMENT

This study is supported by project of shenzhen science and technology innovation committee (JCYJ20130401095947215), Guangdong natural science foundation of China (S2011010006111), key lab of SZIIT (SYS201004) and innovation team of SZIIT (CXTD2-003). The author would like to thank Prof. Liang Yongsheng for his revision of the text, the editor and the anonymous reviewers for their contributions that enriched the final paper.

REFERENCES

Boutillon, E., W.J. Gross and P.G. Gulak, 2003. VLSI architectures for the MAP algorithm. IEEE Trans. Commun., 51: 175-185.

Hagh, M.M., M. Salehi, A. Sharma and Z. Zvonar, 2006. Implementation issues in turbo decoding for 3GPP FDD receiver. Wireless Personal Commun., 39: 165-182.

Kaza, J. and C. Chakrabarti, 2004. Design and implementation of low-energy turbo decoders. IEEE Trans. Very Large Scale Integr. Syst., 12: 968-977.

Lee, D.S. and I.C. Park, 2006. Low-power log-MAP decoding based on reduced metric memory access. IEEE Trans. Circuits Syst. I: Regul. Pap., 53: 1244-1253.

Li, S., D. Grace, Y. Liu, J. Wei and D. Ma, 2011. Overlap area assisted call admission control scheme for communications system. IEEE Trans. Aerospace Electron. Syst., 47: 2911-2920.

Liu, Y.M., D. Grace and P.D. Mitchell, 2009. Exploiting platform diversity for GoS improvement for users with different high altitude platform availability. IEEE Trans. Wireless Commun., 8: 196-203.

Pietrobon, S.S., 1998. Implementation and performance of a turbo/MAP decoder. Int. J. Satellite Commun., 16: 23-46.

Yeo, E. and V. Anantharam, 2003. Iterative decoder architectures. IEEE Commun. Mag., 41: 132-140.