ActaEnergetica
POWER ENGINEERING QUARTERLY

Architecture Framework of High Throughput for the Soft Decision Decoding

Ernst Huenges
Professor, The GFZ German Research Centre for Geosciences, Potsdam, Germany

Mohamed K. Hassan
College of Engineering and Islamic Architecture, Umm Al-Qura University, Makkah, Saudi Arabia

Keywords
High Throughput, Soft Decision Decoding,

Abstract
In this research, we examine the Viterbi method for decoding information that has been convolutionally encoded. Using soft-decision and hard-choice decoding algorithms, we examine potential power-saving techniques that might allow receivers to decode convolutionally coded signals with less power usage. The goal of this study is to only employ the Viterbi decoder when the transmitted message contains errors. A straightforward decoder is employed if there are no mistakes. The MATLAB profiler ® tool was used to research the suggested method. A 2 dB performance gain over the identical solution utilising hard-decision decoding is offered by soft-decision decoding. Digital transmission methods need physical media, such as cables, optical fibres, or even radio channel propagation, which cannot be completely trusted and can modify the data being communicated. The design of computer systems and communication systems now includes error-correcting codes for protection and correction.

Received by the editor: 20.03.2022
Received in revised form: 05.06.2022
Accepted: 24.06.2022

1. INTRODUCTION

For academics, protecting information on computer networks, communications systems, and data storage is a trouble. As a rule, there are two methods for guaranteeing that the data sent is basically as secure as could really be expected. The first is to raise the outflow power, however this approach is incredibly expensive. The second is the inclusion of overt repetitiveness to a message that will be sent or stored. The guideline of mistake adjusting codes is to add excess data to the data to be secured.

Blunder remedying codes are utilized to ensure that mistakes brought about by loud aggravation of data sent in correspondence diverts or stored in computerized storage are recognized and revised however much as could reasonably be expected. Decoding direct codes is commonly a NP-Hard undertaking, and various decoders have been made to find and fix botches. A decoder's quality is evaluated in view of how well it acts as far as touch blunder rate and transient complexity.

Prior to conveying a message, an encoder in a correspondence framework makes codewords by adding excess data to the client vectors' data, and a decoder endeavors to recognize the communicated message that is probably going to have been sent from the grouping of gotten data. The cycle blunder rate (BER) that a coding-decoding framework can ensure at a specific signal-to-noise ratio (SNR), run-time complexity, and equipment assets required are indicators of the framework's quality.

The decoding algorithms can be isolated into two social affairs: hard decision algorithms and soft decision algorithms. Rather than soft decision algorithms, which work directly on got pictures and normally utilize the Euclidian distance as an estimation to pick the most plausible conveyed codeword, hard decision algorithms work on the equal kind of the got data and use the Hamming distance to unravel.

A. Soft Decision Decoding

The following is how a hard and soft decision decoder differ from one another.

- Hard decision decoding analyzes the got codeword to all potential codewords and picks the one with the littlest Hamming distance.

In soft decision decoding, the codeword that outcomes in the littlest Euclidean distance is picked after the got codeword is contrasted with any remaining conceivable codewords. Accordingly, the soft decision decoding works with better decision-production by giving greater dependability information (determined Euclidean distance or determined log-probability proportion)

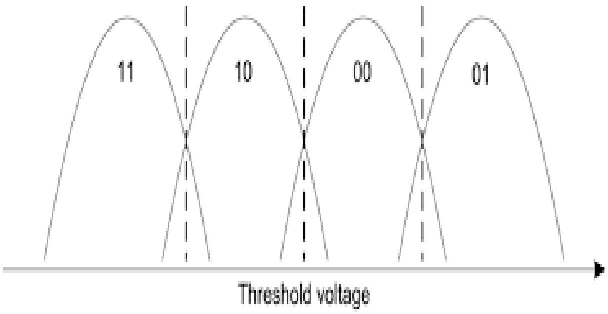


Figure.1: Soft-decision decoding

2. LITERATURE REVIEW

Various methods managing the hard choice unraveling calculations have as of late been portrayed. NESWDA (New Productive Condition Weight Unraveling Calculation), distributed in [2], is one of the different techniques made that is utilized to disentangle double methodical Quadratic Buildup QR(47,24,11) codes with up to five slip-ups. The creators of [3] have made a Cyclic Weight (CW) calculation for translating a similar code, which depends on the heaviness of conditions and properties of cyclic codes. The creators of [4] have likewise given a table query translating calculation to unravel up to five flaws for a similar code. Hash techniques are utilized to build a system in view of the connection among disorders and examples of correctable blunders; in [6], the creators offer a mathematical translating calculation to address all examples of four or less blames in the twofold QR (41, 21, 9) code. The drawn out quadratic buildup twofold codes are the just nontrivial broadened parallel cyclic codes that are invariant under the projective unique direct gathering, the creators have exhibited in [7]. (PSL). In [8], it is proposed to disentangle quadratic buildup codes utilizing hashing search to recognize erroneous examples. As well as proposing equipment engineering to carry out the Lagrange interjection equation, the creators of [9] translated specific double QR codes utilizing the created Berlekamp-Massey (BM) calculation and Chien search.

We find a correlation of the BCH(15, 7, 2) and BCH(255, 231, 3) codes created by the creators of [10] as well as their exhibitions for BCH codes. For Reed Solomon (RS), BCH, and Low Thickness Equality Really look at Codes, the creators of [11] have introduced an exhibitions research and a union of different novel calculations found in this field (LDPC). A profound learning way to deal with improve the conviction engendering calculation was put out in [12]. The creators changed the current conviction engendering calculation by allotting loads to the edges of the Leather expert chart. An iterative hard choice deciphering procedure for double straight block codes over a parallel symmetric channel is given by the creators in [13]. (BSC). A few hard choice decoders in light of hereditary calculations (GA) are created in [14-17]. The first is the HDGA (Hard choice Decoder in view of Hereditary Calculations) [14]. The subsequent one is the Piece Flipping deciphering calculation (BF) [15-16]. It was initially produced for LDPC codes and later summed up to direct obstruct codes. A decoder known as ARDecGA (Fake Reliabilities based Unraveling calculation by Hereditary Calculations) is depicted in [17]. It utilizes a summed up equality really take a look at framework to figure a vector of fake reliabilities of the twofold gotten succession h and utilizations a hereditary calculation to track down the parallel word with the most noteworthy probability to fit this vector. In view of estimation of disorders, the arithmetical hard choice decoder [18, 19] of Berlekamp-Massey has a compelling strategy to find every correctable shortcoming and is appropriate to BCH codes. This last decoder has been altered for use with quadratic buildup codes [20]. In [21], we presented two brand-new hash-based hard choice decoders that are speedy and successful for ongoing correspondence systems. The first, named HSDec, has exceptionally low fleeting intricacy contrasted with contenders yet needs more memory than the subsequent one, called HWDec; interestingly, the latest one requirements more noteworthy run time yet little memory.

A soft decoder (SD1) utilized for BCH encoding in remote body region networks (WBANs) is one of many examinations managing soft decision decoders and was distributed in [22]. To find the blunder area depends on tests and condition estimations. In [23], a new soft BCH decoder (SD2) was acquainted with increment equipment complexity and blunder rectification capacities. In [24], the creator introduced his Soft Stage Decoding Calculation, which is a Soft In-Hard Out (SIHO) variation of MacWilliams' Change Decoding Calculation (McPD) (SPDA). [25] proposes a variant of his SPDA that utilizes iterative decoding. To help synchronization of computerized correspondence systems, Shim proposed forward mistake remedy codes for correspondence diverts in [26]. In [27], two double space soft-decision decoders are proposed utilizing smaller hereditary algorithms (cGA) in bigger tournaments.

3. METHODOLOGY

Quantitative and applied research methods are used in this study. Quantitative research is based on quantity or measurement of quantity, whereas applied research seeks to find solutions to pressing problems facing society or business/industrial organizations[22]. A quantitative research approach is used in this study to ensure accurate results and comprehensive coverage of the topics to be resolved. Subcategories of quantitative research strategies are inference, experimentation, and simulation methods. The goal of inferential research approaches is to create databases from which population characteristics and relationships can be inferred. The research environment is much more tightly controlled with experimental methods. In this situation, change some factors to see how they affect others. Simulation approaches involve creating artificial environments that can generate relevant data and information.

A simulation/empirical approach is used in this study to investigate power saving in communication systems by observing and evaluating the performance of new algorithms compared with conventional systems, and in which soft-decision decoding is to see if it compares to better performance. Hard-decision decoding truncates the dB gain, as shown in [15]. Identification, investigation, confirmation and further development of theoretical ideas are some of the main goals of the approach. The other is to learn from collective experience in this area. The use of appropriate test cases, data collection methods and analysis methods is emphasized. Evaluation and analysis of energy saving potential of new algorithms using bit error performance and execution time as a basis.

4. RESULT & DISCUSSION

As referenced before, the time data is utilized to analyze the normal power utilization of the two decoders. As a guideline, power utilization is supposed to be conversely relative to execution time. We utilized the MATLAB® Profiler tool to work with a decoder with a data length of 10,000 pieces. Profiler data was gathered for individual bundles of 10,000 pieces every when bit mistakes were brought about by tireless AWGN channel noise. Simulations were performed for Eb/No in the reach 1 to 13 dB. The Eb/N0 esteem stayed steady all through each run. The outcomes uncover the relating time and power necessities for every decoder, as displayed in the table and figure below.

Table.1: Total execution time plotted for Normal Viterbi

Eb/No	Total CPU Time
1	40
2	35
3	38
4	42
5	45
6	44

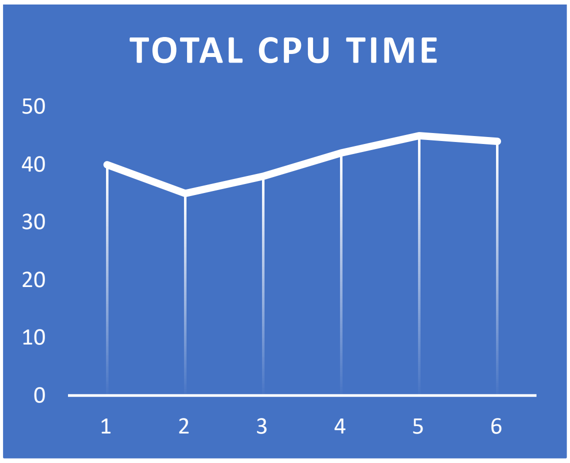


Figure.2: Total execution time plotted for Normal Viterbi

Table.2: Total execution time plot for the Simple Decoder

Eb/No	Total CPU Time
1	1
2	5
3	8
4	10
5	12
6	18

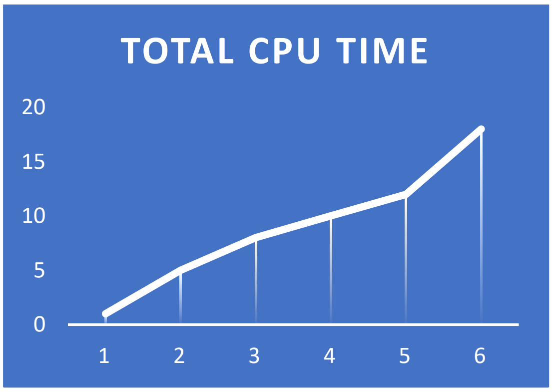


Figure 3. Total execution time plot for the Simple Decoder

Table.3: Total execution time plot for the Modified Viterbi

Eb/No	Total CPU Time
1	10
2	16
3	20
4	22
5	24
6	28

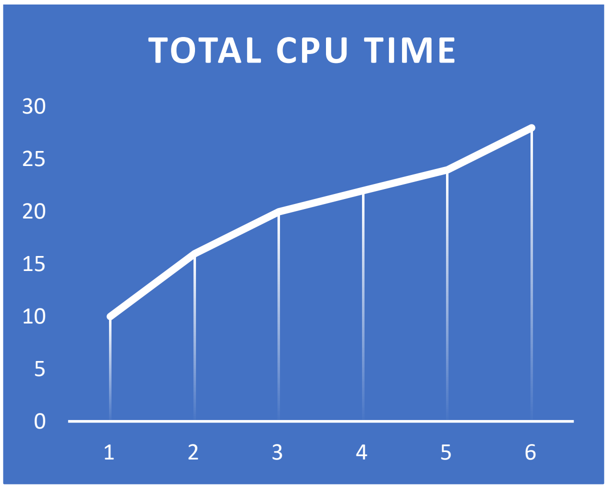


Figure 4. Total execution time plot for the Modified Viterbi

Table.4: Total execution time diagram for the proposed switching decoder

Eb/No	Total CPU Time
1	12
2	14
3	18
4	35
5	38
6	45

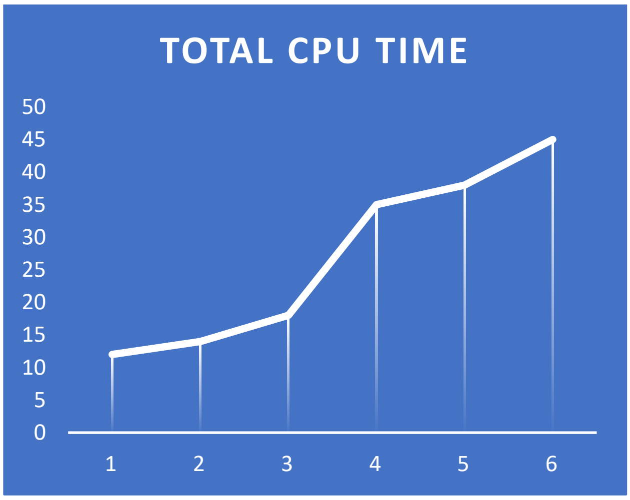
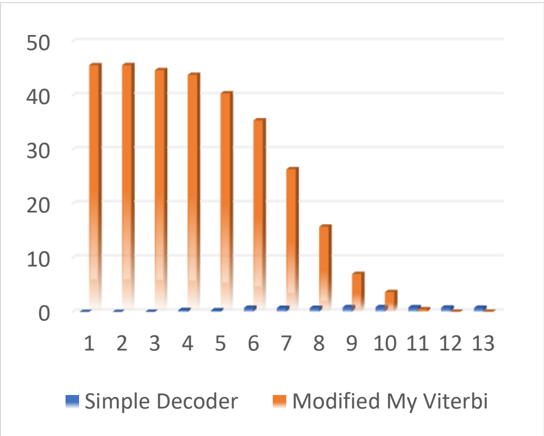


Figure 5. Total execution time diagram for the proposed switching decoder

Table.5: Comparative Timing Analysis of the Proposed Decoder with Conventional Viterbi Decoder

EbNo	Time (cpu seconds)			
	My Viterbi Decoder	Proposed Decoder		
		Simple Decoder	Modified My Viterbi	Total
1	46.2226	0	45.5634	45.5637
2	43.3474	0	45.5769	45.5787
3	42.6647	0.0346	44.6538	44.6354
4	45.4637	0.3411	43.7853	43.6897
5	47.3631	0.2772	40.3759	40.3462
6	44.6353	0.7426	35.3732	35.2632
7	44.4363	0.7349	26.3462	27.2364
8	45.5436	0.7297	15.7284	16.2462
9	44.3635	0.8593	7.0020	8.6276
10	45.3527	0.8573	3.6422	3.4410
11	44.3645	0.8389	0.4622	1.5288
12	44.2643	0.7833	0.0472	1.6468
13	44.2680	0.7537	0.0456	0.9988



CONCLUSION

The figure above shows that as EbNo builds, how much pieces decoded by the adjusted My Viterbi decoder diminishes. Lessening the quantity of pieces likewise essentially decreases the time expected to disentangle the pieces. The total time taken by the proposed decoder can be determined by adding the time taken by the superior My Viterbi decoder and the basic decoder. Accordingly, Eb/N0 decides how long it requires for the proposed decoder to finish. At large Eb/N0 levels, where the essential decoding part does the majority of the decoding, less time is expected to finish the decoding. The versatile Viterbi decoding area performs more decoding as the Eb/N0 esteem diminishes. Subsequently, it requires a long investment to finish the decoding process. It was tracked down that when Eb/N0 is 3 dB, the planning of the exchanging decoder is almost equivalent to that of the customary Viterbi decoder. Below 3 dB, exchanging and customary Viterbi decoders take about a similar measure of time. The power investment funds start at around 5 dB, instead of 2.3 dB (for the soft-decision execution), contrasted with the aftereffects of a similar method utilizing hard-decision decoding. This study upholds the case that soft-decision decoding really depends on 3 dB better than hard-decibel decoding.

FUTURE WORK

The decoding ability of the proposed method in this research effort has been significantly improved. However, there are other directions in which the current work can be extended. A parallel decoding framework can be used in ultra-high-speed communication systems in the Gbps range, but the actual and exact limits of this framework are not yet known. Considering the practical application of this work, this is very important. Finding optimization methods other than convex optimization and evaluating their performance are important additional steps that can be taken to move the work forward. Similarly, new optimization techniques may be discovered that are particularly suitable for use in the RS decoding framework.

REFERENCES

1. Jacobsmeyer, M.J., (1996). Introduction to Error Control Coding. Pericle Communciations Company. [Online]. Available at: [Accessed 10 June 2011]
2. Jin, J., Chi-Ying Tsui., (2006). A low power Viterbi decoder implementation using scarce state transition and path pruning scheme for high throughput wireless applications. Proceedings of the 2006 international symposium on Low power electronics and design, Germany, 4-6 Oct. 2006, pp. 406 – 411.
3. Shaker, S.W., (2009). Design and Implementation of Low-Power Viterbi Decoder for SoftwareDefined

WiMAX Receiver. 17th Telecommunications forum TELFOR, Belgrade, 24-26 Nov 2009, pp. 468-471.
4. Kothari C. R. (2004). Research Methodology: Methods and Techniques, New Age International (P) Ltd., New Delhi.
5. Anjali, K. S., (2010). Energy saving Viterbi Decoder for Forward Error Correction in Mobile Networks. M.Sc Thesis, University of Manchester.
6. Michelle, A. (2002). Steps in Empirical Research, PPA 696 Research Methods, California State University. [Online]. Available at: [Accessed 11 July 2011].
7. Clark, G. C. Jr. and Cain. J. B., (1981). Error-Correction Coding for Digital Communications. New York: Plenum Press.
8. Sklar, B., (2001). Digital Communications – Fundamentals and Applications. 2nd ed. New Jersey: Prentice Hall
9. Basilead Library. (2006), What is Empirical Research?. Tutorials and Research Guides. [Online]. Manor College. Available at: [Accessed 11 July 2011]
10. Seki K.; Kubota S.; Mizoguchi M. and Kato S., (1994) "Very low power consumption Viterbi decoder LSIC employing the SST (scarce state transition) scheme for multimedia mobile communications," Electronics-Letters, IEE, Vol.30, no.8, April p.637-639.
11. Lang L.; Tsui C.Y. and Cheng R.S., (1997) "Low power soft output Viterbi decoder scheme for turbo code decoding," Conference-Paper, ISCAS „97(Cat. No97CH35987). IEEE, New York, NY, USA, 4 vol. Lxvi+2832 pp. 1369-1372.
12. Oh D. and Hwang S., (1996) "Design of a Viterbi decoder with low power using minimum-transition traceback scheme," Electronic-Letters, IEE, Vol.32, No.22, pp. 2198-2199.
13. Viterbi, A. J., (1967). Error bounds for convolutional codes and an asymptotically optimum decoding algorithm. IEEE Trans. Information Theory., vol. 13 no.2, pp.260-269.
14. Garrett D. and Stan M., (1998) "Low power architecture of the soft-output Viterbi algorithm," Electronic-Letters, Proceeding 98 for ISLEPD "98, p 262-267.
15. Fleming, C., (2002). Tutorial on Convolutional Coding with Viterbi Decoding. Spectrum Applications. [Online]. Available at: [Accessed 8 June 2011]