Technische Universität München
Lehrstuhl für Nachrichtentechnik
Prof. Dr. sc. techn. Gerhard Kramer

Master's Thesis

Polar FEC chain development in Software for 5G

Vorgelegt von:
Yadhunandana Rajathadripura Kumaraiah

München, November 2018

Betreut von:
Dipl.-Ing. Advisor

Master's Thesis am
Lehrstuhl für Nachrichtentechnik (LNT)
der Technischen Universität München (TUM)
Titel : Polar FEC chain development in Software for 5G
Autor : Yadhunandana Rajathadripura Kumaraiah

Yadhunandana Rajathadripura Kumaraiah
Schröfelhofstraße 14 WG 02/02
81375 München
yadhu.kumaraiah@tum.de

Ich versichere hiermit wahrheitsgemäß, die Arbeit bis auf die dem Aufgabensteller bereits bekannte Hilfe selbständig angefertigt, alle benutzten Hilfsmittel vollständig und genau angegeben und alles kenntlich gemacht zu haben, was aus Arbeiten anderer unverändert oder mit Abänderung entnommen wurde.

München, xx.xx.20xx

.....

Ort, Datum (Yadhunandana Rajathadripura Kumaraiah)

Contents

1 Introduction and Motivation **3**

2 Polar Codes Background **5**

3 Encoding FEC Chain **7**

4 Decoding FEC Chain **9**

4.1 Decoding algorithms 9

5 Conclusion **11**

Bibliography **I**

List of Figures

1.1 PDF $p_N(N)$ of the number N of times that the head side is up. 3

3.1 Polar Encoding FEC chain for PDCCH/PBCH 8

List of Tables

Abstract

The abstract comes here.

1 Introduction and Motivation

In 1948, scientists at the Bell Laboratories achieved two landmark research results: Claude E. Shannon published his paper *A mathematical theory of communication* [Sha48], and John Bardeen, Walter Brattain and William Shockley announced the invention of the *transistor effect.*
[Ari09]
A binomial distribution is shown in Figure 1.1.

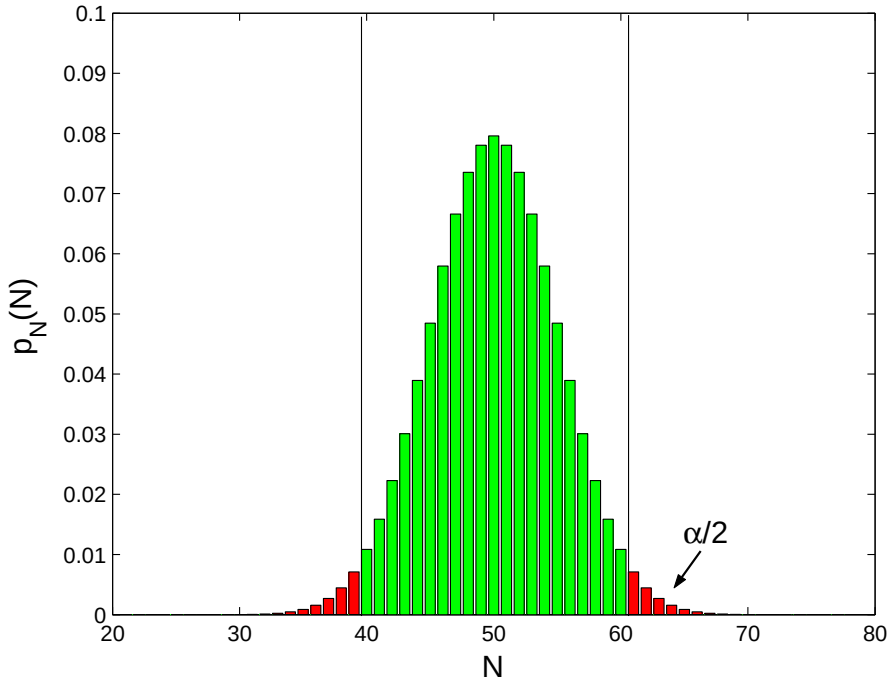

Figure 1.1: PDF $p_N(N)$ of the number N of times that the head side is up.

For further information, the reader is referred to [CT91, Cla01, Sac92, **?**, BGT93, Che03, Onk02, Hag01, Wei].

Traditionally FEC chains are developed in hardware i.e FPGAs or ASICs to achieve low latency and high throughput. Development in FPGA/hardware requires more time and

costly. With recent advances in General Purpose Processors it is possible to achieve required latency and throughput with software implementations without custom hardware. Software implementations are flexible and easy to maintain compared hardware implementations.
However algorithms need to be adopted/optimized to efficiently implement in software.
Recent advances in the modern processors such as SIMD units can be utilized to achieve low latency and high throughput.

2 Polar Codes Background

Polar codes

3 Encoding FEC Chain

Work I have done Until now.
Optimizations to the original implementations until now. Generic optimizations - Using optimization primitives such as likely and unlikely. - Aligning memory to 32 bytes so copying of data can be vectorized. Polar transform optimization - Replace binary additions with xor. instead of addition and then modulus two. - division and multiplications by left and right shift operations. - Avoided copy operations in polarTransform operations. Optimization in getting reliability indices. - Avoided remove and erase operations which have huge overhead. Wrote a efficient mechanism(reduced the latency by 176 us). - Instead removing and erasing I mark the element as removed. - Since the reliability indexes won't change. I built a look up table in place of searching all (1024)indices reduced the latency by 40us - Avoided copying operations of interleaved indexes. - Unrolled the loop to reduce the jumps. Rate matching optimizations. - optimization in subblock interleaving, Rewrote the logic to avoid E number of division and modulus operations. - Unrolled the for loops in subblock inteleaving method. - Implemented optimal version of bit selection, Avoided E number of modulus operations which are very costly. - Again optimization primitives for helping the branch predictor.
Fast version of Encoding API's. In the original implementation of the polar encoding each of the bit is treated as 32 bit integer. This is highly inefficient when the goal is to process multiple bits at time. With each bit considered as 32 bit integer SIMD instructions won't provide any performance improvement. Reason is SIMD instruction can process multiple bits at time. avx2 instructions 256bits at a time. if we have 32 bits to represent a single bit. we can process only 8 bits at time. Which doesn't significantly improve the performance. To avoid this disadvantage and make use of SIMD capability. each 64 bit integer is considered as 64 bits of data. so one avx2 instruction can process 256 data bits in a single instruction. - Built a look up table to avoid last eight stages of polar encoding instead of traversing till end of tree. - Implemented SIMD instruction based encoding. Encoding happens within 0.6 us for N = 512. - Implemented optimal version of CRC calculation which can calculate CRC for PDCCH chain within 0.8 us. Original implementation was taking 7 us. - Implemented a bit interleaver which can deal with this format of data.

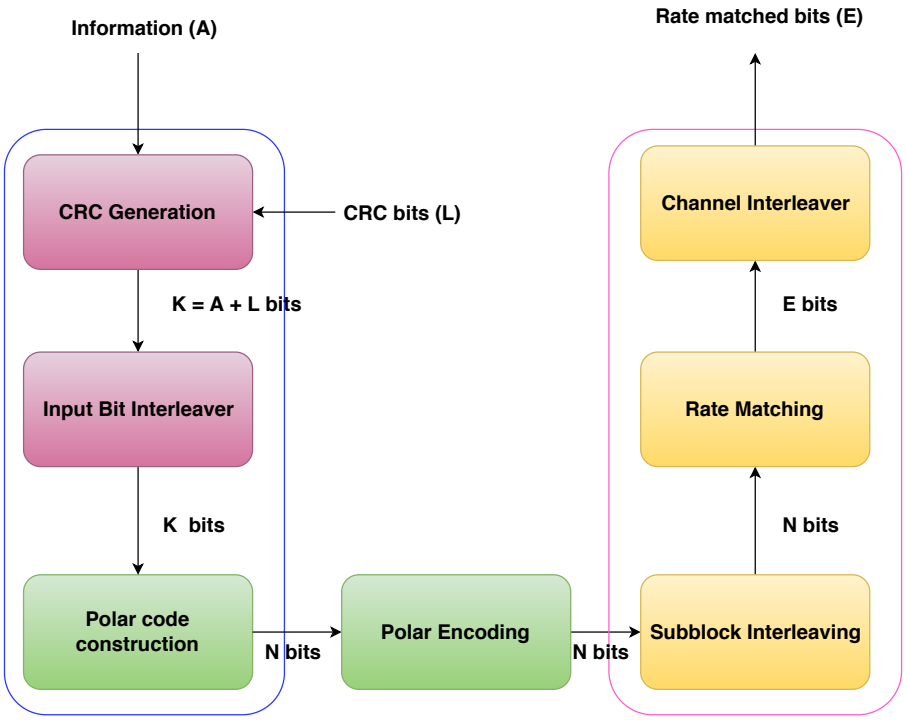

Figure 3.1: Polar Encoding FEC chain for PDCCH/PBCH

4 Decoding FEC Chain

Decoding is of serial nature, has lot of latency.

4.1 Decoding algorithms

5 Conclusion

Bibliography

[Ari09] E. Arikan, "Channel polarization: A method for constructing capacity-achieving codes for symmetric binary-input memoryless channels," in *IEEE Transactions on Information Theory*, vol. 55, no. 7, 2009, pp. 3051–3073.

[BGT93] C. Berrou, A. Glavieux, and P. Thitimajshima, "Near shannon limit error-correcting coding and decoding: Turbo-codes," in *IEEE International Conference on Communications*, Geneva, Switzerland, May 1993.

[Che03] G. Chechik, "Types, super-types and the mutual information distribution," Leibniz Center, School of computer sciene and engineering, The Hebrew university of Jerusalem, Tech. Rep. 2002-61, 2003.

[Cla01] D. Clayton, "Population association," in *Handbook of Statistical Genetics*, D. Balding, M. Bishop, and C. Cannings, Eds. Chichester: John Wiley & Sons, 2001, pp. 519–540.

[CT91] T. Cover and J. Thomas, *Elements of Information Theory*, ser. Wiley series in telecommunications. New York: John Wiley & Sons, 1991.

[Hag01] J. Hagenauer, "Informationstheorie und Quellencodierung," Manuskript zur Vorlesung, Lehrstuhl für Nachrichtentechnik, Technische Universität München, 2001.

[Onk02] P. Onkamo, "Genetic mapping of complex traits: the case of type 1 Diabetes," Ph.D. dissertation, University of Helsinki, Helsinki, Jan. 2002.

[Sac92] L. Sachs, *Angewandte Statistik*, 7th ed. Berlin: Springer-Verlag, 1992, ch. Die Auswertung von Mehrfeldertafeln, pp. 608–611.

[Sha48] C. E. Shannon, "A mathematical theory of communication," *Bell System Technical Journal*, vol. 27, pp. 379–423, 623–656, July, October 1948.

[Wei] E. Weisstein. Delta function. [Online]. Available: http://mathworld.wolfram.com/DeltaFunction.html