International Journal of Information, Control and Computer Sciences
ISSN: 2517-9942
Vol:2, No:5, 2008

Parallel Joint Channel Coding and Cryptography

Nataša Živić, and Christoph Ruland

***Abstract*—**Method of Parallel Joint Channel Coding and Cryptography has been analyzed and simulated in this paper. The method is an extension of Soft Input Decryption with feedback, which is used for improvement of channel decoding of secured messages. Parallel Joint Channel Coding and Cryptography results in improved coding gain of channel decoding, which achieves more than 2 dB. Such results are an implication of a combination of receiver components and their interoperability.

***Keywords*—**Block length, Coding gain, Feedback, L-values, Parallel Joint Channel Coding and Cryptography, Soft Input Decryption.

I. Introduction

CHANNEL coding uses redundant information for the recognition or correction of errors that occur during the data transfer over a noisy channel. Cryptography provides secure information transfer: it protects against eavesdropping or manipulation of transmitted information, or masquerading of data origin.

The cooperation between channel coding and cryptography has been researched in [1] and [2]: using channel decoding for the improvement of decryption results and, vice versa, using cryptography for the improvement of channel decoding. This concept is called Joint Channel Coding and Cryptography.

A message with a cryptographic check value is transmitted over a noisy channel using channel coding and decoding. The decryption of the cryptographic check value is very delicate, because if one bit or more of the input of decryption is wrong, about 50 % of decrypted bits are false, and the verification of cryptographic check value fails. Therefore, all bits of the message and the cryptographic check value have to be correct at the input of decryptor. The solution for such problem uses the method of correction which is called Soft Input Decryption [1]: if the decoder is not able to reconstruct the original message and cryptographic check value because of a noisy channel or inefficiency of the channel decoding algorithm, it is possible to correct the message with the cryptographic check value using side information of the channel decoder in form of so called *L*-values.

Improvement of channel decoding can be accomplished using a message with its cryptographic check value which has been corrected by Soft Input Decryption. This method uses corrected *L*-values as feedback information to the channel decoder for improved decoding of those bits which have not been yet corrected [2]. The feedback method is iterative, because *L*-values corrected in one round are used for the correction of bits in the next iteration.

The idea of inversion of the least probable bits (with the lowest reliability values) originated from Chase decoding algorithms [3] in 1972, which were the generalization of the GMD (Generalized Minimum Distance) algorithms from 1966 [4]. These algorithms have been applied to a binary (n, k) linear block code and are referenced as LRP (Least Reliability Positions) algorithms [5].

Manuscript received on April 14, 2008. and accepted on Jun 2, 2008.

Nataša Živić is with the Institute for Data Communications Systems, Electrical Engineering and Informatics Department, University of Siegen, Siegen, 57076, Germany (e-mail: natasa.zivic@uni-siegen.de).

Christoph Ruland is the manager of the Institute for Data Communications Systems and a professor at the Electrical Engineering and Informatics Department, University of Siegen, Siegen, 57076, Germany (e-mail: christoph.ruland@uni-siegen.de).

II. Soft Input Decryption

Soft Input Decryption is the basic technique which is used in this paper. The main component is a decryptor which uses soft output of the channel decoder as soft input [1].

The cryptographic mechanism [6] which is used by encryptor and decryptor generates and verifies cryptographic check values (hash values [7], digital signatures [8][9][10], MACs [11], H-MACs [12]) providing data integrity, data origin authentication and non repudiation [13][14].

The algorithm of Soft Input Decryption is presented in Fig. 1 and functions as follows:

The decryption is successful, if the verification of the cryptographic check value is positive, i.e. the output of the decryptor is "true". In case that the verification is negative, the soft output of the channel decoder is analyzed and the bits with the lowest $|L|$-values are flipped (XOR "1") [1]. Afterwards, the decryptor performs the verification process and proves the result of the verification again. If the verification is again negative, bits with another combination of the lowest $|L|$-values are changed. This iterative process is finished when the verification is successful or the needed resources are consumed.

If attempts for correction fail, the number of errors is too large as a result of a very noisy channel or an attack, so that the resources are not sufficient to try enough combinations of flipping bits of low $|L|$-values.

It case that the attempts for correction of SID block succeed,

but the corrected cryptographic value is not equal to the original one, it means that a collision happens. This case has an extremely low probability when cryptographic check values are chosen under security aspects.

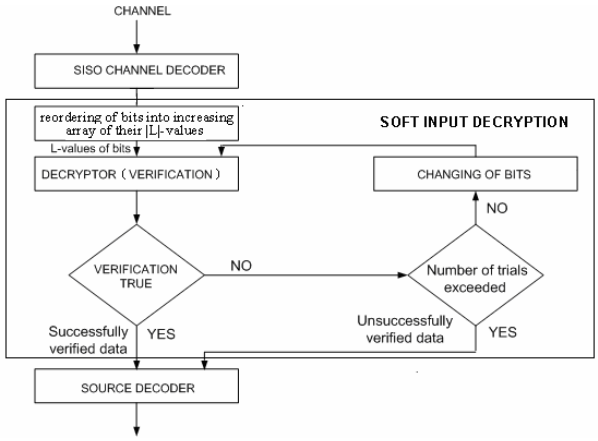

Fig. 1 Algorithm of the Soft Input Decryption

Soft Input Decryption is block oriented. The block which is taken from sequential input bits to the channel encoder and should be corrected by Soft Input Decryption after channel decoding is called SID block (Soft Input Decryption block). The SID block may have different contents depending on cryptographic mechanisms and scenarios [1].

III. Parallel Joint Channel Coding and Cryptography

Joint Channel Coding and Cryptography uses Soft Input Decryption with feedback [2]. The input of the encryptor is a data block, which may be part of a data stream. The data block is split in two parts of the same length, message *ma* and message *mb*, both of length of *m*. Each of both messages is extended by a cryptographic check value *na* and *nb,* both of length *n*, using a cryptographic check function RCF (generation of a digital signature, MAC/H-MAC) – see Fig. 2.

In general, the lengths of message parts *ma* and *mb* and the lengths of cryptographic check values *na* and *nb* do not have to be the same. In [2] it is shown, that different lengths of *ma*, *mb*, *na* and *nb* have only minor influence on BER and that equal lengths for *ma* and *mb*, as well as for *na* and *nb*, show the best results. Therefore, equal lengths of message parts as well as cryptographic check values are used in this paper.

Block *a* consists of the message part *ma* and the redundancy check value *na*:

$$a = a_1 a_2 \ldots a_{m+n} = ma_1 ma_2 \ldots ma_m\, na_1 na_2 \ldots na_n \quad (1)$$

Block *b* consists of the message part *mb* and the redundancy check value *nb*:

$$b = b_{1b} b_2 \ldots b_{m+n} = mb_1 mb_2 \ldots mb_m\, nb_1 nb_2 \ldots nb_n \quad (2)$$

Interleaving of block *a* and block *b* forms the assembled message *u*:

$$u = a_1 b_1\, a_2 b_2 \ldots a_{m+n}\, b_{m+n} \quad (3)$$

u is encoded by a convolutional or turbo code (inner code), modulated and transferred over the noisy channel.

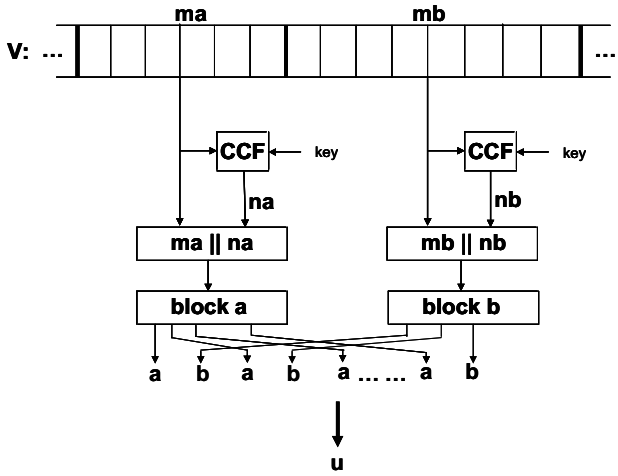

Fig. 2 Interleaving of blocks *a* and *b* into message *u*

After demodulation of the received message, Joint Channel Coding and Cryptography is applied in 3 steps (Fig. 3).

Step 1:
- channel decoding with resulting BER_{cdl}
- segmentation and de-interleaving of the output *u'* of the decoder into block *a´* and block *b´*, and
- parallel Soft Input Decryption with feedback of block *a´* and block *´b*.

The following cases depend on the results of step 1:

Case 1.
the results of the first Soft Input Decryption (1. SID) of block *a'* and Soft Input Decryption of block *b'* are correct, i.e. BER after 1. SID is 0:

$BER_{1.SID} = 0$

u is corrected and no other actions are necessary.

Case 2.
the result of Soft Input Decryption of block *a'* is correct, but block *b'* could not be corrected. So, a half of bits are corrected (belonging to block *a'*), and another half of bits (belonging to block *b'*) have *BER* as after channel decoding:

$BER_{1.SID} = ½\ BER_{cdl}$

Step 2 of Case 2.
The second step consists of feedback [2] from block *a* corrected by Soft Input Decryption to block *b*. L-values of block *a* block are set to ±∞, L-values of block *b* are set to 0, which represent unknown bits [2]. The SISO decoder decodes

u again with these L-values as input. Resulting *BER* after step 2 is $BER_{feedback}$.

Step 3 of Case 2.
The third step is a second Soft Input Decryption (2. SID). block *b´* is tried to be corrected by Soft Input Decryption. Resulting *BER* after this step is $BER_{2.SID}$. As step 3 is the last step of the algorithm, total *BER* is equal to $BER_{2.SID}$.

Case 3.
The result of Soft Input Decryption of block *b'* is correct, but block *a'* could not be corrected. As in Case 2.:

$BER_{1.SID} = ½\ BER_{cd1}$

Step 2 and 3 of Case 3.
These steps correspond to steps 2 and 3 of Case 2., but with difference that the symbols a´ and b´ are exchanged.

Case 4.
Neither the result of Soft Input Decryption of block *a'* nor the result of Soft Input Decryption of block *b'* is correct: BER is equal to BER of the convolutional or turbo decoder (BER of the inner code, BER_{cd1}).
No further actions are possible.

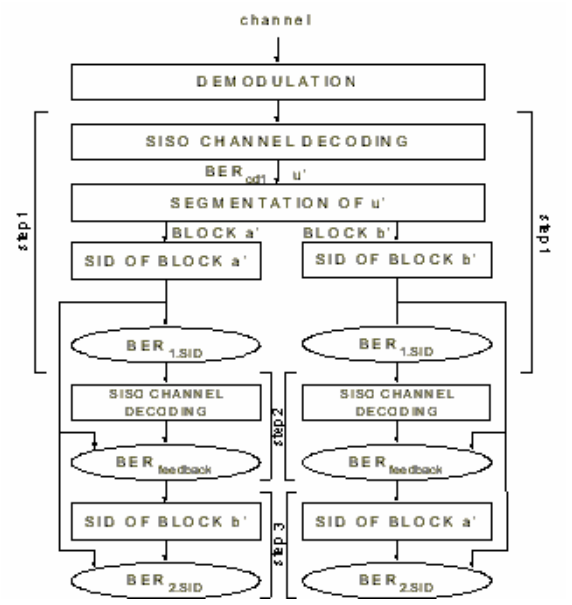

Fig. 3 Algorithm of Parallel Joint Channel Coding Cryptography

IV. Results of Simulations

Parallel Joint Channel Coding and Cryptography has been simulated with the block *a* and block *b*, each of length of 320 bits. Simulations are performed using convolutional encoder of a code rate r = 1/2 and constraint length m = 2 (Fig. 4), BPSK modulation, AWGN channel and SISO decoding using MAP algorithm [15]. MAP decoding algorithm was chosen, because it gives better results in scope of Soft Input Decryption than another well known SISO decoding algorithm: SOVA [16][17]. The reason for that is, that MAP is based on correction of single bits, which is needed for Soft Input Decryption, and SOVA founds the most probable sequence of bits.

For each point of resulting curves, 50 000 simulations are performed, which is more than enough for reliable results [18].

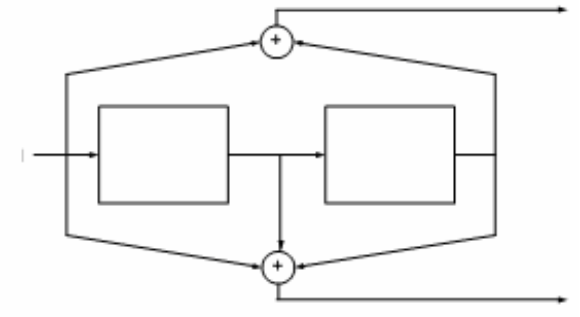

Fig. 4 Convolutional encoder $r = 1/2$, $m = 2$

BER after each step of the algorithm is shown in Fig. 5. The coding gain increases with increasing of E_b/N_0.

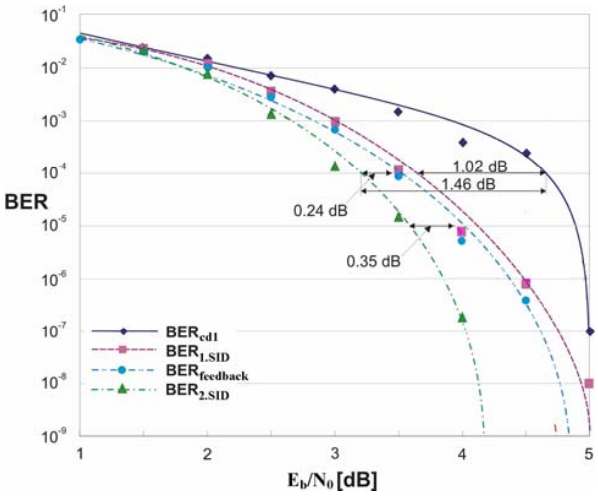

Fig. 5 BER after each step of Parallel Joint Channel Coding and Cryptography Algorithm, in comparison to ½ channel decoding

The coding gain after the 3. step of Parallel Joint Channel Coding and Cryptography algorithm reaches 1.46 dB for BER of 10^{-4}. Obviously, the second Soft Input Decryption has a significant influence on coding gain, adding 0.24 dB coding gain to the coding gain of the 2. step of the algorithm.

V. Impact of Block Lengths

Simulations in this Chapter are performed in order to examine the impact of lengths of blocks *a* and *b* to the coding gain of the Parallel Joint Channel Coding and Cryptography.

Parameters of simulations are the same as in Chapter IV.

The used block lengths are presented in Table I:

TABLE I
BLOCK LENGTHS USED FOR SIMULATIONS

Nr. test	1	2	3	4	5
length (block *a*) = length (block *b*)	128	160	256	320	640

The results of simulations are presented in Fig. 6, showing significant difference of BER for different lengths of blocks. Coding gain is bigger for shorter blocks, as expected, because of higher efficiency of Soft Input Decryption and feedback [19]. Therefore, the biggest coding gain is achieved for the shortest blocks *a* and *b* (128 bits each): over 2 dB. Vice versa, for blocks of 640 bits length, coding gain reaches 1.04 dB.

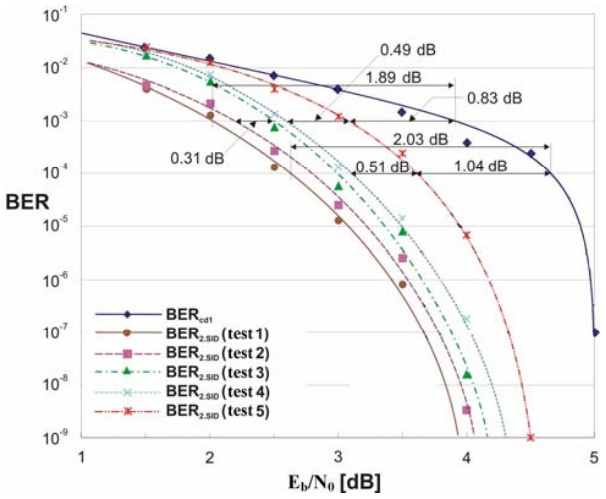

Fig. 6 Coding gains of Parallel Joint Channel Coding and Cryptography Algorithm of different block lengths, in comparison to ½ channel decoding

VI. CONCLUSION AND FUTURE WORK

This paper analyzes Parallel Joint Channel Coding and Cryptography as an efficient algorithm for correction of channel decoding results of messages which are protected by security mechanisms.

Performed simulations have shown, that coding gain increases after every of three steps of Parallel Channel Coding and Cryptography algorithm.

Efficiency of Parallel Channel Coding and Cryptography strongly depends on used block lengths. Simulations using different block lengths have shown, that coding gain varies from above 1 dB – for longer blocks, to above 2 dB – for shorter blocks.

Future work should include analysis of the influence of different convolutional encoders (with different coding rates) to Parallel Channel Coding and Cryptography. Further on, by adding new iteration steps by assembling more than two blocks of messages and cryptographic check values, decoding results would be improved through the turbo effect of decoding.

REFERENCES

[1] N. Živić, C. Ruland, "Soft Input Decryption", *4th Turbocode Conference, 6th Source and Channel Code Conference*, VDE/IEEE, Munich, April 2006.
[2] N. Živić, C. Ruland, "Feedback in Joint Coding and Cryptography", *7th International ITG Conference on Source and Channel Coding VDE/IEEE*, Ulm, January 2008.
[3] D. Chase: "A Class of Algorithms for Decoding Block Codes with Channel Measurement Information", *IEEE Trans. Inform. Theory, IT-18*, pp. 170-182, January 1972.
[4] G.D.Jr. Forney: "Generalized Minimum Distance Decoding", *IEEE Trans. Inform. Theory, IT-12*, pp. 125-131, April 1966.
[5] S. Lin, D.J. Costello: *Error Control Coding*, Pearson Prentice Hall, USA, 2004
[6] C. Ruland, *Informationssicherheit in Datennetzen*, Datacom Verlag, Bergheim, 1993.
[7] ISO/IEC 10118-1, *Information technology – Security techniques – Hash-functions – Part 1: General*, 2000.
[8] ISO/IEC 14888-1, *Information technology – Security techniques – Digital signatures with appendix – Part 1: General*, 1998.
[9] ISO/IEC 9796-2, *Information technology – Security techniques – Digital signatures giving message recovery – Part 2: Discrete logarithm based mechanisms*, 2006.
[10] C. Ruland, "Realizing digital signatures with one-way hash function", *Cryptologia*, *Vol XVII, Number 3*, July 1993.
[11] ISO/IEC 9797-1, *Information technology – Security techniques – Message Authentication Codes (MACs) – Part 1: Mechanisms using a block cipher*, 1999.
[12] ISO/IEC 9797-2, *Information technology – Security techniques – Message Authentication Codes (MACs) – Part 2: Mechanisms using a hash- function*, 2000.
[13] ISO/IEC 9798-1, *Information technology – Security techniques – Entity authentication mechanisms – Part 1: General*, 1997.
[14] ISO/IEC 13888-1, *Information technology – Security techniques – Non-repudiation – Part 1: General*, 2004.
[15] L. Bahl, J. Jelinek, J., Raviv, F. Raviv, "Optimal decoding of linear codes for minimizing symbol error rate", *IEEE Transactions on Information Theory*, IT-20, March 1974.
[16] J. Hagenauer, P. Höher: "A Viterbi algorithm with soft-decision outputs and its applications", *Proc. IEEE GLOBECOM `89*, Dallas, Texas, USA, pp. 1680-1686, November 1989.
[17] F. – H. Huang: "Evaluation of Soft Output. Decoding for Turbo Codes", *thesis at the Faculty of the Virginia Polytechnic Institute*, May 1997, http://scholar.lib.vt.edu/theses/available/etd-71897-15815/unrestricted/
[18] M. Jeruchim, P. Balaban, K. S. Shanmugan, "Simulation of Communication Systems", *Kluwer Academic/Plenum Publ*, New York, 2000.
[19] N. Živić, C. Ruland: "Channel Coding as a Cryptography Enhancer", *WSEAS Transactions on Communications*, http://www.worldses.org/journals/communications/communications-2008.htm, Vol. 7, March 2008.