Halmstad University Post-Print

Packet combining and doping in concatenated hybrid ARQ schemes using iterative decoding

Elisabeth Uhlemann, Tor M. Aulin, Lars K. Rasmussen and Per-Arne Wiberg

N.B.: When citing this work, cite the original article.

©2003 IEEE. Personal use of this material is permitted. However, permission to reprint/republish this material for advertising or promotional purposes or for creating new collective works for resale or redistribution to servers or lists, or to reuse any copyrighted component of this work in other works must be obtained from the IEEE.

Uhlemann E, Aulin T, Rasmussen L, Wiberg P. Packet combining and doping in concatenated hybrid ARQ schemes using iterative decoding. In: The IEEE Wireless Communications and Networking, 2003. WCNC 2003. 2003 IEEE. Piscataway, NJ: IEEE; 2003. p. 849-854. IEEE Wireless Communications and Networking Conference, 2003, vol. 2.

DOI: <http://dx.doi.org/10.1109/WCNC.2003.1200482>

Copyright: IEEE

Post-Print available at: Halmstad University DiVA

<http://urn.kb.se/resolve?urn=urn:nbn:se:hh:diva-403>

Packet Combining and Doping in Concatenated Hybrid ARQ Schemes Using Iterative Decoding

Elisabeth Uhlemann^{*†}, Tor M. Aulin[†], Lars K. Rasmussen^{†‡} and Per-Arne Wiberg^{*}

^{*}Computer and Communications Lab.
Halmstad University, Box 823
SE-301 18 Halmstad, Sweden
{bettan, pelle}@ide.hh.se

[†]Department of Computer Engineering
Chalmers University of Technology
SE-412 96 Göteborg, Sweden
aulin@ce.chalmers.se

[‡]Inst. for Telecommunications Research
University of South Australia
Mawson Lakes SA 5095, Australia
lars.rasmussen@unisa.edu.au

Abstract — We consider serially concatenated block codes in a hybrid ARQ scheme using iterative decoding. The extrinsic information generated in the iterative decoding process is saved and used when a retransmission is decoded. Two different strategies are examined; one using the extrinsic information only in the very first iteration, whereas the other uses it in all subsequent iterations until another retransmission arrives. The latter can be seen as turbo or concatenated code combining whereas the former, where the extrinsic information is used only once may be seen as code doping, providing an alternative perspective. The strategy of saving the extrinsic information is also compared to traditional type-III, equal gain diversity combining. Using the extrinsic information from previous retransmissions is shown to improve performance not only in terms of bit error rate but also in terms of throughput and convergence speed and requires only negligible additional decoder complexity. The performance of this strategy is however not as good as simple equal gain combining. As a consequence, the investigated schemes are not competitive alternatives, however, the code doping procedure can be used in conjunction with traditional diversity combining schemes, improving further on convergence speed.

I. INTRODUCTION

The first time a turbo code, i.e. a parallel concatenated convolutional code, was used in a hybrid ARQ (HARQ) scheme was in [1]. Here, the extrinsic information was saved and used when decoding a retransmitted packet. Since then, a series of schemes have been suggested and investigated [2]–[8]. The strategy of saving and using extrinsic information may in principle be seen as a form of packet combining, which traditionally is divided into two categories, namely diversity combining and code combining. Code combining is generally done within the decoder, e.g., a rate compatible code in a type-II HARQ scheme [9], [10]. Diversity combining such as equal gain combining has been referred to as a type-III HARQ scheme, but is in principle a special case of a type-II HARQ scheme where simple repetition codes are used for incremental redundancy [11]. Due to the simple repetition code, such diversity combining can be done by trivial averaging over observables, prior to decoding.

For concatenated codes with iterative decoding, different packet combining techniques are available, e.g., concatenated

code combining, equal gain combining and combinations of the two since concatenated code combining is possible in conjunction with traditional diversity combining. Here, we focus on a HARQ scheme based on serially concatenated block codes (SCBC). Different packet combining techniques specifically adapted to the SCBC HARQ scheme are thus investigated in the following.

The retransmission scheme is intended for wireless real-time applications, where a packet must be delivered prior to a specified deadline [12]. A truncated ARQ scheme is therefore considered [13], where only a limited number of retransmissions is allowed. The decoder result following the final allowed retransmission is always delivered, regardless of quality. In this scenario, we are particularly interested in the performance in terms of bit error rate (BER) as a function of time, or equivalently, the number of retransmissions.

The paper is organized as follows. In Section II, the system model is described in some detail. Section III defines concatenated hybrid ARQ schemes, describes an inherent retransmission criterion based on averaged log-likelihood ratios, and defines a series of packet combining techniques unique to iterative decoding. Performance results based on simulations are presented and discussed in Section IV and concluding remarks completes the paper in Section V.

II. SYSTEM MODEL

We consider a SCBC in a HARQ scheme using iterative decoding. The SCBC encoder is shown in Fig. 1.

Specifically, we consider a SCBC based on two binary Reed-Solomon RS(7,3) codes with a pseudo-random interleaver of size 945 [14]. To provide a simple scenario for

Fig. 1. Encoder for a serially concatenated code.

This work was funded by the national Swedish Real-Time Systems research initiative ARTES (www.artes.uu.se), supported by the Swedish Foundation for Strategic Research. Part of this work was done while E. Uhlemann was visiting University of South Australia in 2002.

performance comparison, binary phase shift keying (BPSK) is used to transmit coded bits over an additive white Gaussian noise (AWGN) channel with two-sided power spectral density equal to $N_0/2$. Throughout this work, an error free feedback channel is assumed.

III. CONCATENATED HYBRID ARQ

The term *concatenated hybrid ARQ* (CHARQ), denotes here a HARQ scheme using concatenated codes as the error control code, but also using “concatenation” between retransmissions. Soft information from previous decoding attempts may now be used in the iterative decoding process of a retransmission.

A. Retransmission Criterion - Convergence Behavior

A common approach for stopping an iterative decoding process is to allow for a fixed number of iterations. This may lead to unnecessary iterations or performance degradation, if the process is terminated prematurely. Performance based stopping rules related to the convergence behavior of the iterative decoder have been considered in [15]–[18]. In [15], a stopping criterion based on thresholding the average log-likelihood ratios (LLRs) is found to be efficient. Further, a HARQ scheme is also suggested in [15], using the LLR threshold as both a stopping criterion and a retransmission criterion, hence no additional error detecting code is needed. We adopt the same approach for the CHARQ scheme applied here. The LLR threshold is set to ± 10 and the maximum allowed number of iterations to seven.

In Fig. 2, the BER is given as a function of E_b^0/N_0 for CHARQ schemes allowing between zero and four retransmissions. E_b^0 refers to the bit energy of the regular scheme with no retransmissions. For high signal to noise ratio, $\text{SNR} = E_b^0/N_0$, all the curves tend towards the regular scheme without ARQ, since the probability of a retransmission tends to

Fig. 2. BER as a function of E_b^0/N_0 for CHARQ schemes with different truncation length. Erroneous packets are discarded.

zero. The same behavior is observed for very low SNR, in this case because the probability of a retransmission tends to one. Virtually all transmissions are rejected here and thus there is nothing to gain from a retransmission, since the final decoder output will always be accepted. The region between these two extreme values, however, is the working area of the ARQ schemes and a noticeable gain can be observed. The reason for showing the different truncation lengths in one figure is to see the improvement when more time is allocated to allow for additional retransmissions. At high SNR, a loss in performance due to the stopping criterion can be observed. This is due to the threshold being fixed for all values of SNR. However, the stopping criterion is good in the working area of the ARQ schemes since we, in this area, obtain fast convergence at virtually no performance loss [14].

In Fig. 3 the throughput of the same CHARQ schemes can be seen. We have defined the throughput as the number of accepted packets over the total number of transmitted packets, implying that the throughput of the regular scheme, without ARQ is equal to one. The throughput is a direct measure of the actual code rate and it can be observed that there is indeed a reduced code rate within the working area of the ARQ schemes. The decrease in code rate is directly related to the increase in bit energy caused by the retransmissions.

B. Packet Combining Techniques - Decoding Strategies

The basic scheme used here is a type-I HARQ scheme. We can therefore directly apply equal gain diversity combining [19]. This implies that the receiver averages over the demodulator output metrics from all received copies of a packet to produce a combined packet for decoding. This is equivalent to including an additional repetition code within the scheme.

Using iterative decoding, another kind of packet combining technique is possible. When the decoding process is terminated

Fig. 3. Throughput as a function of E_b^0/N_0 for the different CHARQ schemes in Fig. 2. Note that the regular scheme without ARQ has throughput =1.

and a retransmission is requested, we have in fact side information available in terms of extrinsic information from the previous transmission. This information can be saved and used in the decoding process of a retransmission. For example, using the notation of Fig. 4, if the extrinsic information of the output code bits from the outer decoder, Oco , is saved, it can be used as *a priori* information, Iui , when decoding the *new copy*, r_1 , of the packet received from a retransmission. The extrinsic information fed to the inner decoder from the previous transmission, which henceforth will be referred to as *inner extrinsic information*, is denoted Iui_0 in Fig. 4. It should be noted that the inner extrinsic information, saved from the previous transmission, will have a new updated value in the next iteration. Two different scenarios are therefore considered. In the first scenario the inner extrinsic information is used just once – in the first iteration. In the second scenario, the inner extrinsic information from the previous transmission is used in all subsequent iterations of the newly received packet. It will thus be added to the new updated extrinsic information available after the first iteration of the new packet.

Alternatively, we see from Fig. 4 that we can save the extrinsic information on the output information bits of the *outer* decoder, Ouo . This information is then used as *a priori* information on the same information bits, Oui , for the outer code when decoding after a retransmission. We will refer to this information as *outer extrinsic information* and it is denoted Oui_0 in Fig. 4. Using the outer extrinsic information we make use of a previously unconnected input, hence no updated value will be available after the first iteration. We regard the same two cases; using the outer extrinsic information only in the first iteration and using the outer extrinsic information in all subsequent iterations of the decoding of a new packet until yet another retransmission occurs.

The process of saving the inner extrinsic information and using it once requires no extra memory since this is information that is already saved to be used and updated

between the decoders. It is simply not reset when a new copy of a packet arrives. When we save and use the inner extrinsic information in all subsequent iterations, extra memory for storing the information is required together with the extra addition operation, much like when equal gain combining is used. Saving and using the outer extrinsic information once or always may also increase the decoder complexity as we are now making use of a previously not connected decoder input.

The strategy of saving the extrinsic information can be seen as turbo code combining or concatenated code combining. Prior inner/outer extrinsic information implies prior knowledge, as if we received the systematic bits of the inner/outer code in a previous transmission. In some sense this gives us a combined lower rate code even though we are using a type-I ARQ scheme.

The strategy of saving the extrinsic information can alternatively be seen as a doping procedure to speed up convergence of the iterative decoder [20]. The concept of doping is based on providing side information to the decoding process. The extrinsic information from the decoding process of the previous transmission constitutes useful side information, which can lead to a constructive bias, speeding up convergence. With this interpretation in mind, equal gain diversity combining may be used in conjunction with the saved inner or outer extrinsic information.

IV. PERFORMANCE RESULTS

The following retransmission protocol is applied:

1. Use iterative decoding on the received packet. Check the LLR threshold after each iteration. If the stopping criterion is fulfilled go to 3. If seven iterations are reached, go to 2.
2. If four retransmissions are made, go to 3, else request a retransmission. Do doping/code combining if any. Go to 1.
3. Output hard quantized values.

We first address a scheme where the inner extrinsic information is saved and used in the decoding process of a retransmission. In Fig. 5, the BER is plotted versus E_b^0 / N_0 , where E_b^0 again is the bit energy for the regular scheme with no retransmissions. The dotted curves represent CHARQ schemes where the inner extrinsic information is used once whilst the dash-dotted curves are CHARQ schemes using the inner extrinsic information in all the subsequent iterations. It can be seen that the BER in the working area of the ARQ systems is improved when the inner extrinsic information is used as compared to the case of no combining in Fig. 2. Note that the curves representing a regular scheme with no ARQ in both Fig. 2 and Fig. 5 are the same since no packet combining is made for these curves and the same stopping criterion is used. Using the inner extrinsic information always seems to be better for very low SNR, whereas using it only once is best around SNR=0.5 dB.

Fig. 4. Code combining or doping with inner and outer extrinsic information. *I* and *O* stands for inner and outer decoder, whereas *c* denotes code bits, *u* information bits and *i* output and input respectively

Fig. 5. BER as a function of E_b^0/N_0 for different CHARQ schemes. The inner extrinsic information is used in either one or all subsequent iterations.

When extrinsic information is used in all subsequent iterations it is similar to doing packet combining, i.e., when diversity or code combining is applied, two or more packets are combined for decoding during the entire decoding process. Such combining will in general improve performance, when all packets (individually) otherwise would have been rejected. However, this may not be the case if one of these packets were of sufficient reliability to be accepted by itself. Combining such a packet with one or more rejected packets will invariably reduce reliability as compared to only using the acceptable packet. The old rejected packets, however, still constitute valuable side information that can be used to set off the convergence process in the right direction. That it is in fact the *right direction* in most cases can be seen by the improvement in BER for the CHARQ schemes using the inner extrinsic information once as compared to the CHARQ schemes of Fig. 2. From results related to Fig. 5, it can be concluded that for $\text{SNR} < 0.5$ dB, most individual packets are rejected, providing full benefits of packet combining. For $\text{SNR}=0.5$ dB, a significant number of packets is acceptable without combining, and thus, better performance is obtained through doping, i.e., using the extrinsic information once, rather than through full packet combining.

In Fig. 6 the BER of a scheme saving the outer extrinsic information is shown and again the dotted curves represent CHARQ schemes where the outer extrinsic information is used once whilst the dash-dotted curves are CHARQ schemes using the outer extrinsic information in all the subsequent iterations. Also in this case using the extrinsic information always is better for very low SNR whereas using it once is better around $\text{SNR}=0.5$ dB. When comparing the results in Fig. 6 to Fig. 5 there is a slight difference between saving the outer and inner extrinsic information, i.e., affecting the inner or the outer decoder leads to slightly different BER performance. The effects of inner and outer doping or code combining

Fig. 6. BER as a function of E_b^0/N_0 for different CHARQ schemes. The outer extrinsic information is used in either one or all subsequent iterations.

respectively, depend on individual decoder transfer characteristics, which in turn are code dependent, and hence their relationship will require more investigation in order to determine the preferred strategy.

Obviously, we are not only concerned with performance in terms of BER, but also in terms of throughput. In our case, however, the number of retransmissions together with the number of iterations made for each retransmission is as important as the traditional throughput since we are concerned with real-time performance. We want to know if the packet combining technique considered improves both the throughput and the convergence speed, i.e. two schemes may need the same number of retransmissions, but one of them may require less iterations of the last retransmission, hence it is superior in a time critical application, where we want to deliver the result as fast as possible. Since the number of iterations in our considered schemes is limited to seven and the number of transmissions to five, the maximum number of iterations ever occurring is 35.

We start by comparing the schemes saving and using the inner and outer extrinsic information once to that where no packet combining technique is used. The relative frequency of occurrence of each iteration is plotted in Fig. 7. Note that after seven iterations a retransmission occurs. Consequently, iteration number eight in Fig. 7 corresponds to the first iteration of a newly received retransmission. Hence, the first seven iterations are the same for the different schemes, as no retransmissions have yet occurred. The light gray columns in Fig. 7 correspond to the scheme discarding erroneous copies of a packet, hence when a new packet arrives the iterative process is restarted. This can be seen as a staircase characteristic with a step every seventh iteration. The dark gray and the black columns correspond to the schemes saving and using the extrinsic information once. The number of iterations decreases smoother since the decoder does not have to restart every time

Fig. 7. Relative frequency of occurrence for each iteration plotted at SNR=0 dB. A scheme without packet combining is compared to schemes saving and using the inner and outer extrinsic information once.

a retransmission arrives. It can be noted that the scheme using the outer extrinsic information has a slightly better performance than the one using the inner extrinsic information. Consequently, doping with the extrinsic information does not only reduce the BER, but also convergence is faster – resulting in higher throughput.

Fig. 7 is plotted at SNR=0 dB, however, in this area using the extrinsic information in all subsequent iterations resulted in the lowest BER. Consequently, in Fig. 8 the relative frequency of occurrence for each iteration for the CHARQ schemes using the inner and outer extrinsic information in all subsequent iterations are compared. We notice again that the convergence speed is slightly improved using the outer extrinsic information as opposed to the inner. However, the most important difference is noticed when comparing Fig. 8 to Fig. 7, namely that if the extrinsic information is used in all subsequent iterations, the performance is improved not only in terms of BER but also in terms of convergence speed.

At SNR=0.5 dB the overall throughput is higher, but the performance relations between the schemes in terms of convergence speed is approximately the same as for SNR=0 dB, i.e. the scheme with the best performance is still the one using the outer extrinsic information always. Recall that the schemes using the extrinsic information once have a better performance in terms of BER at 0.5 dB.

When equal gain combining is applied, additional memory is required to store erroneous copies used in the combining process, but the gain in terms of BER is considerable as can be seen in Fig. 9. It is observed that the BER increases when the SNR increases. This seems intuitively wrong, but is simply an artifact of using an SNR measure which is proportional to the code bit energy rather than the information bit energy. The $\text{SNR} = E_b^0 / N_0$ used in Fig. 2, Fig. 5, Fig. 6 and Fig. 9 is defined with reference to a regular scheme without retransmissions, whereas an ARQ scheme has an effective

Fig. 8. Relative frequency of occurrence for each iteration plotted at SNR=0 dB. A scheme saving and using the inner extrinsic information always is compared to a scheme using the outer extrinsic information always.

SNR related to the average number of retransmissions. The effective SNR is 3 dB higher when two packets have been combined using equal gain combining as compared to a single packet at the same reference SNR. Going from 0 to 1 dB, more and more transmissions are considered reliable after the first transmission as opposed to the second transmission [14], and thus, the effective SNR decreases, even though the reference SNR increases. In this case the effective SNR decreases faster than the reference SNR increases, leading to the behavior observed in Fig. 9. This together with the imperfect stopping criterion explains the increase in BER observed between 0 and 1 dB.

It can be concluded that the equal gain combining scheme is a more effective packet combining scheme than concatenated code combining using saved extrinsic information, both in terms of BER and in terms of throughput. Trying to use concatenated code combining together with equal gain

Fig. 9. BER as a function of E_b^0 / N_0 for different CHARQ schemes. Traditional equal gain combining is used.

combining only results in an increased BER and negligible throughput improvements.

However, using the saved extrinsic information once may also be seen as doping and as such it may be used together with equal gain combining. Doing so has no noticeable impact on the BER as the equal gain combining scheme in itself already uses all available information, but at the same time the BER is not increased [14]. However, the throughput and convergence speed improves. The reason for this is that the effective SNR is already so high for the equal gain combining scheme that code doping only has negligible effect on BER. Convergence speed is however, in general improved by doping.

Additional results, together with a more detailed analysis can be found in [14]. It should also be noted that type-II code combining schemes based on more powerful codes than repetition codes perform significantly better. For iterative decoding schemes, especially incremental redundancy schemes as the one suggested in [7] are of interest.

V. CONCLUSIONS

Serially concatenated block codes with iterative decoding are incorporated into a hybrid ARQ scheme, exploiting the observed convergence behavior. The stopping criterion for the iterative decoding process is used as a retransmission criterion, constructing a concatenated hybrid ARQ scheme with iterative decoding. The goal is to reject packets with slow or no convergence behavior early and request a retransmission to hopefully receive a packet with fast convergence behavior.

The extrinsic information used in the iterative decoder can be saved from one transmission and then used in the decoding process of a retransmission. This can be done either only in the first iteration or in all subsequent iterations until the next retransmission. The latter can be viewed as turbo or concatenated code combining. Its performance improvement in terms of BER as well as throughput and convergence speed is noticeable at very low additional cost in terms of decoder complexity. Its improvements in performance are most clear at very low SNR.

It has, however, also been observed that a traditional equal gain, repetition code combining scheme is a more effective packet combining scheme than concatenated code combining using saved extrinsic information, both in terms of BER and in terms of throughput.

Using the extrinsic information once can be viewed as a code doping operation providing useful side information, both for equal gain combining and concatenated code combining. It is concluded that doping with prior extrinsic information does improve the performance in terms of BER and throughput by setting off the convergence process in the right direction. Its

improvement in performance is most noticeable at medium SNR.

REFERENCES

- [1] K. R. Narayanan and G. L. Stüber, "A novel ARQ technique using the turbo coding principle," *IEEE Commun. Letters*, pp. 49-51, March 1997.
- [2] J. Hámorský, U. Wachsmann, J. B. Huber and A. Cizmár, "Hybrid automatic repeat request scheme with turbo codes," *Int. Symp. on Turbo Codes*, Brest, France, pp. 247-250, Sept. 1997.
- [3] A. Banerjee, D. J. Costello Jr. and T. E. Fuja, "Bandwidth efficient hybrid ARQ schemes using turbo codes," *Proceedings of the IEEE Int. Symp. Info. Theory*, ISIT 2000, p. 188, June 2000.
- [4] J. Li and H. Imai, "Performance of hybrid-ARQ protocols with rate compatible turbo codes," *Int. Symp. Turbo Codes*, Brest, pp. 188-191, Sept. 1997.
- [5] T. Ji and W. E. Stark, "Concatenated punctured turbo Reed-Solomon codes in a hybrid FEC/ARQ DS/SSMA data network," *Proc. IEEE Veh. Tech. Conf.*, pp. 1678-1682, May 1999.
- [6] J. Hámorský and L. Hanzo, "Performance of the turbo hybrid automatic repeat request system type-II," *Proc. Info. Theory and Networking Workshop*, ITW '99, p. 51, July 1999.
- [7] D. N. Rowitch and L. B. Milstein, "On the performance of hybrid FEC/ARQ systems using rate compatible punctured turbo (RCPT) codes," *IEEE Trans. Commun.*, vol. 48, pp. 948-959, June 2000.
- [8] Y. Wu and M. C. Valenti, "An ARQ technique using related parallel and serial concatenated convolutional codes," *Proc. Int. Conf. Commun.*, ICC 2000, pp. 1390-1394, June 2000.
- [9] J. Hagenauer, "Rate-compatible punctured convolutional codes (RCPCC) and their applications," *IEEE Trans. Commun.*, vol. 36, pp. 389-400, April 1988.
- [10] S. Lin and D. J. Costello Jr., *Error Control Coding: Fundamentals and Applications*, Prentice-Hall, Inc., Englewood Cliffs, N.J., 1983.
- [11] D. M. Mandelbaum, "An adaptive-feedback coding scheme using incremental redundancy," *IEEE Trans. Info. Theory*, vol. 20, pp. 388-389, 1974.
- [12] E. Uhlemann, T. M. Aulin, L. K. Rasmussen and P.-A. Wiberg, "Concatenated hybrid ARQ - A flexible scheme for wireless real-time communication", *IEEE Real-Time and Embedded Tech. and Appl. Symp.*, (San Jose, California), pp. 35-44, Sept. 2002.
- [13] E. Malkamäki and H. Leib, "Performance of truncated type-II hybrid ARQ schemes with noisy feedback over block fading channels," *IEEE Trans. Commun.*, pp. 1477-1487, Sept. 2000.
- [14] E. Uhlemann, *Hybrid ARQ using serially concatenated block codes for real-time communication - an iterative decoding approach*, Licentiate Thesis, Chalmers University of Technology, Göteborg, Sweden, Oct. 2001. Available at <http://www.hh.se/staff/bettan/>.
- [15] A. C. Reid, T. A. Gulliver and D. P. Taylor, "Convergence and errors in turbo-decoding," *IEEE Trans. Commun.*, pp. 2045-2051, Dec. 2001.
- [16] J. Hámorský and U. Wachsmann, "Criteria for minimum bit error rate decoding of turbo codes," *Proc. Tagungsband der Kleinheubacher 1995*, pp. 607-616, Oct. 1995.
- [17] M. Moher and T. A. Gulliver, "Cross-entropy and iterative decoding," *IEEE Trans. Info. Theory*, vol. 44, pp. 3097-3104, Nov. 1998.
- [18] R. Y. Shao, S. Lin and M. Fossorier, "Two simple stopping criteria for turbo decoding," *IEEE Trans. Communications*, vol. 47, pp. 1117-1120, Aug. 1999.
- [19] M. Schwartz, W. R. Bennett and S. Stein, *Communication Systems and Techniques*, McGraw-Hill, Inc. 1966.
- [20] S. ten Brink, "Code doping for triggering iterative decoding convergence," *Proc. IEEE Int. Symp. Info. Theory*, p. 235, June 2001.