

 Open access • Proceedings Article • DOI:10.1109/VLSISP.1993.404497

Systematic design optimization of a competitive soft-concatenated decoding system

— [Source link](#) 

O.J. Joeressen, G. Schneider, H. Meyr

Institutions: RWTH Aachen University

Published on: 20 Oct 1993 - IEEE Workshop on VLSI Signal Processing

Topics: Concatenated error correction code, Serial concatenated convolutional codes, Sequential decoding, Convolutional code and List decoding

Related papers:

- [Design Space Exploration of Hard-Decision Viterbi Decoding: Algorithm and VLSI Implementation](#)
- [Optimal decoding of linear codes for minimizing symbol error rate](#)
- [Improved error control techniques for data transmission](#)
- [Influence of LSI and VLSI technology on the design of error-correction coding systems](#)
- [Transform Decoding of Reed-Solomon Codes. Volume I. Algorithm and Signal Processing Structure](#)

Share this paper:    

View more about this paper here: <https://typeset.io/papers/systematic-design-optimization-of-a-competitive-soft-1cg9nltbc9>

Systematic Design Optimization of a Competitive Soft-Concatenated Decoding System

*Olaf J. Joeressen, Gregor Schneider, Heinrich Meyr
RWTH Aachen, Lab. for Integrated Systems for Signal Proc.
ISS - 611810, Templergraben 55, D-52056 Aachen, Germany
Tel: +49-241-807632, email: joeresse@ert.rwth-aachen.de*

Due to the advances in VLSI technology complete digital communication systems can today be implemented on single application specific VLSI circuits. The optimum choice of implementation parameters, such as signal wordlengths, is a critical design task since poor parameter choices can lead to costly designs. On the other hand, the high number of parameters to be selected span a large search space that is very difficult to handle. We present a new systematic approach to parameter selection in this paper and apply this approach to the design optimization of a decoding system for a concatenated coding scheme. Two convolutional codes are concatenated and both are decoded by soft decision decoding. This is facilitated by means of soft output decoding of the inner code. The performance of the scheme is better than that of the well known standard code with 64 states for moderate BER at equivalent implementation cost. The proposed coding scheme is thus an attractive alternative whenever high bit error rate performance is a prerequisite, e.g. for digital HDTV transmission.

1. Introduction

It has been known for a long time that code concatenation can be an attractive alternative to using a single code. The majority of the literature on concatenated coding has however focused on concatenating either block codes or an inner convolutional code and an outer block code [1]. This was due to the fact that, firstly, no decoders which provide soft outputs were available at reasonable complexity for the inner processing step and, secondly, for block codes efficient codes and efficient decoding algorithms based on hard quantized input samples were already available for a long time. Convolutional codes in conjunction with Viterbi decoding have been employed successfully for the inner decoder wherever soft quantized values are available as input [2], while soft input decoding of block codes, with the notable exception of errors and erasures decoding, is less common due to complexity. It has however been noted that decoding schemes with a hard deciding inner decoder (hard concatenating) can be improved if information about the decoding process is passed between subsequent decoders [3]–[5].

A prominent member of the class of soft output decoding algorithms is the Soft Output Viterbi Algorithm (SOVA), a modification of the Viterbi algorithm, which was developed by Hagenauer and Höher [6]. The algorithm allows not only the most likely path sequence to be found (as the Viterbi algorithm) but in addition delivers a reliability value for each decoded bit. It has been shown that the algorithm is well suited for VLSI implementation [7,8] which

gives rise to further investigation of schemes employing SOVA. In this paper we investigate a soft concatenated scheme for two convolutional codes which employs SOVA for the inner and the Viterbi algorithm (VA) for the outer decoding step. The decoder for the proposed coding scheme is naturally more complex to design than a single decoder. In particular, more implementation parameters have to be selected, which has proved to be a difficult task in VLSI system design [9]. Special attention is thus paid in the course of the paper to the problem of choosing the set of implementation parameters which leads to the most efficient VLSI implementation. A new systematic optimization approach for the joint optimization of implementation parameters of digital signal processing hardware is presented and applied to the proposed scheme.

2. System Outline and Performance

The goal of our work was to investigate whether soft concatenated Viterbi coding is an alternative to a single convolutional code with respect to implementation cost and performance. While in [3] schemes with rate $1/4$ were investigated, we decided to take the well known standard code with constraint length $K = 7$ and rate $r = 1/2$ [2] as our benchmark. Due to implementation cost, codes of rate $1/2$ (and $K < 7$) were punctured [10] and concatenated. Figure 1 outlines the resulting transmission system.

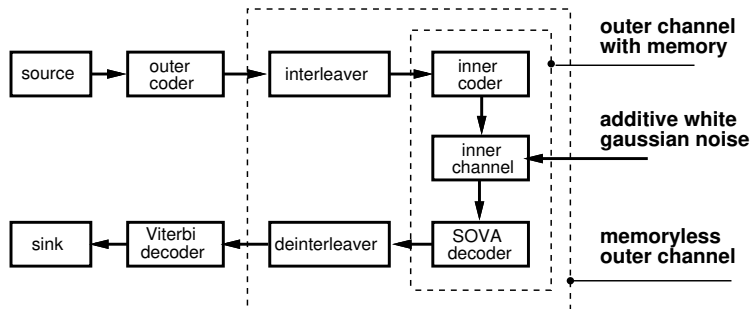


Figure 1. System Outline

The simulated inner channel modulation scheme is BPSK and the SNR figures are given in energy per information bit versus single sided noise power spectral density. Several schemes were investigated and Figure 2 shows the result for concatenating punctured codes of constraint length $K = 5$ as compared to our benchmark. The inner and outer code are derived from the same original code which leads to implementation advantages.

The performance of the concatenated scheme improves if the inner code rate approaches $1/2$. The scheme with inner rate $r_i = 4/7$ and outer rate $r_o = 7/8$ equals the performance of the benchmark at bit error rate (BER) $2 * 10^{-4}$, whereas the scheme with $r_i = 2/3$ and $r_o = 3/4$ is about $0.4dB$ worse. We found no further improvement for inner code rates below $r_i = 4/7$ with punctured codes as listed in [10]. The concatenated schemes exhibit a steeper overall characteristic and the best scheme provides approximately $0.5dB$ additional coding gain at $BER = 10^{-5}$ compared to the benchmark scheme ($K = 7, r = 1/2$). Thus applications which require error rates below 10^{-4} are likely to benefit from the scheme.

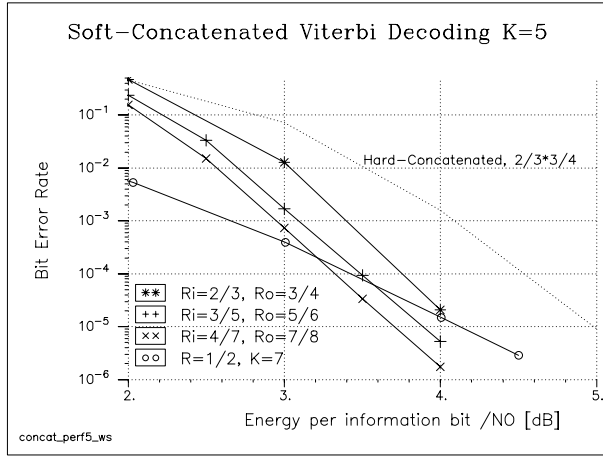


Figure 2. Performance of Soft-Concatenated Viterbi Decoding (K=5)

3. Implementation Parameters

Of course Figure 2 tells only half the story if a VLSI implementation is considered, because the included results represent the pure algorithm performance without considering the effect of implementation parameters such as limited wordlengths. Figure 3 gives an overview of our decoding path and the most important implementation parameters. Note that puncturing masks and constraint length are not mentioned since they are assumed to be fixed in accordance with the performance results of Figure 2 for $r_i = 4/7$.

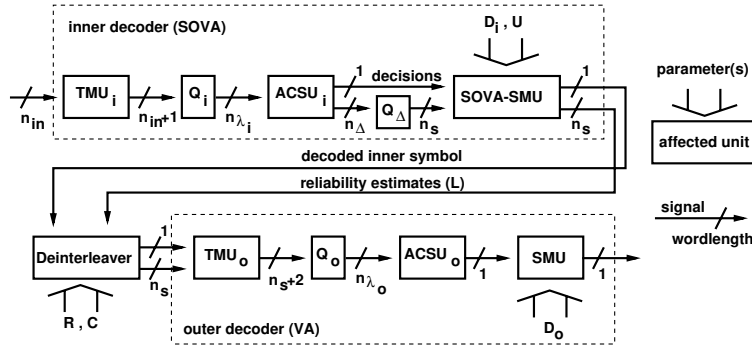


Figure 3. System Parameters

The blocks are the transition metric units (TMU), add compare select units (ACSU), survivor memory units (SMU), quantizers (Q) and the deinterleaver. The first group of implementation parameters are the wordlengths of the involved signals. Table 1 explains the notation of wordlengths in Figure 3.

We assume there are no additional quantizers in the basic blocks. Thus some wordlengths can be derived from others. Consider the input and output of the TMU of the outer decoder (TMU_o). The input is made up of decoded symbols

Wordlength	Signal(s)
n_{in}	input samples
n_{λ_i}	branch metrics of inner ACS (λ_i)
n_{Δ}	path metric differences (Δ)
n_s	quantized Δ , symbol reliability (L)
n_{λ_o}	branch metrics of outer ACS (λ_o)
1	all decision bits and decoded symbols

Table 1. Wordlengths and Signals

from the inner decoder (one bit) and reliability estimates with wordlength n_s , which together form a bit metric of wordlength $n_s + 1$. Since our original code is of rate 1/2, two of these metrics are required to calculate a branch metric of wordlength $n_s + 2$. The outer metric quantizer Q_o provides the metrics λ_o to the outer ACS ($ACSU_o$) and may, if appropriate, reduce the wordlength to a value $n_{\lambda_o} \leq n_s + 2$. The other group of parameters denote the sizes of implementation structures. Table 2 below explains the notation.

Parameter	Meaning
D_i	survivor depth of inner decoder
U	update depth of inner decoder
R	rows of deinterleaver
C	columns of deinterleaver
D_o	survivor depth of outer decoder

Table 2. Remaining Parameters

The update depth U is a parameter of the SOVA-SMU which affects the quality of the reliability estimates L of the decoded bits. While the original formulation of the SOVA requires the path comparison and update operation for the depth of the SMU (D_i in our case), it has been shown that U can be chosen significantly smaller [8]. It becomes clear from the variety of parameters that parameter optimization, with the goal of finding the overall best implementation for a given acceptable performance loss, becomes a critical task due to the large search space. In particular it is impossible to simulate the performance for each possible parameter set. A tool for automatic optimization would thus need to automatically start simulations for parameter sets that are determined by the optimization program [11]. The major disadvantages of such an approach are that, firstly, relatively complex software is required to run the optimization and the required simulations and, secondly, no information about useful ranges of single parameters is provided.

To avoid these disadvantages, we base our design flow on the assumption that the overall implementation loss is the sum of implementation losses found for varying a single parameter while the others remain fixed to a certain reference parameter set. This allows an approximate picture of the design space to be obtained with limited simulation effort. Although the above assumption often provides a good approximation of the design space, the determined optimum parameter set needs to be verified with respect to the achieved performance. If significant differences to the expected result are found, further optimization

steps can be performed iteratively with the determined parameter set of the previous iteration as the new reference parameter set.

The following section shows how we optimized our design. We started with the unquantized design and very large setting of the structural parameters as the reference parameter set. We then varied single parameters to determine the equivalent implementation loss with respect to this parameter. Subsequently, the optimum parameter set was determined based on the obtained data with respect to implementation cost and verified by simulation. The optimization objective was to find the best solution (in terms of the area consumption of the chip) which provides an implementation loss of less than $0.2dB$. The decoder throughput was not within the scope of the optimization since this parameter is usually predefined by the application. Furthermore, only the wordlength of the branch metric influences this parameter and the dependency of the throughput on this parameter is weak.

4. Simulation Data and Area Models

Since we deal with implementation losses in the range of $0.01dB$ special attention was paid to the problem of obtaining reliable data. To ensure sufficient accuracy of the results of the Monte Carlo simulations, the simulation length was adjusted to average over a minimum of 7000 bit errors. The resulting charts are sufficiently smooth, but one should keep in mind, that there is some uncertainty. On the other hand, the optimization result has proved the viability of the approach. All simulation results are given for $SNR = 3dB$. In order to be able to optimize the parameters with respect to implementation area, hardware architectures need to be selected and area models are required. However, the area models do not need to reflect the complete area but only those parts of the implementation architectures which are affected by the parameters. This simplifies the models considerably since several blocks of our design are fixed by the choice of the constraint length.

We did not include the wordlengths n_{Δ} and n_{in} in the optimization process. The wordlength of the metric difference n_{Δ} is dependent on n_{λ_i} and the properties of the code and is thus not a free parameter. Since a TMU is a cheap device in terms of area consumption, at the end of the optimization process the input wordlength n_{in} can be chosen sufficiently large. Thus, we focused attention on the quantization of the branch metrics and the metric differences provided by the inner ACSU. Table 3 summarizes the simulation results in terms of the implementation loss, given in dB .

Wordlength	Loss (n_{λ_i})	Loss (n_{ϵ})	Loss (n_{λ_o})
2	-	0.15	-
3	0.13	0.013	0.15
4	0.04	0.01	0.019
5	0.005	-	0.007

Table 3. Performance Effect of the Quantizers

The wordlength of the metric difference n_{ϵ} influences big parts of the design (deinterleaver and SOVA-SMU) and is discussed later on. The parameters n_{λ_i} and n_{λ_o} affect mainly the ACSUs, since the quantizers Q_i , Q_o and the TMUs are small devices. For the ACSUs area estimates were obtained by logic

synthesis from VHDL descriptions. All area results and formulas presented below represent accumulated cell area multiplied by factors which account for wiring. Since full-custom macros usually require less wiring overhead than standard cells, the area of the used RAM blocks was multiplied by 2.0 while the standard cell area was multiplied by 2.5. The target technology was the $1\mu\text{m}$ CMOS standard cell technology from European Silicon Structures (ES2). Table 4 gives the area results for the inner and outer ACSU. The inner ACSU is slightly bigger than the outer ACSU since the metric differences are additional outputs of the inner ACSU. The synthesized ACSUs allow clock speeds of approximately 50MHz.

$n_{\lambda_{i,o}}$	ACSU _i (mm ²)	ACSU _o (mm ²)
3	5.6	4.4
4	6.44	5.4
5	7.6	6.56

Table 4. Area of the ACSUs versus branch metric wordlengths

The SOVA-SMU is the most complex part of the design. We have chosen the two-step architecture presented in [7] as implementation architecture. The SOVA-SMU is composed of a hard deciding register exchange SMU, delay lines, path comparison unit and update unit. The area of a register exchange SMU is roughly proportional to D_i . The delay lines need to delay the decision bits as well as the quantized metric differences by D_i clock cycles. The area is thus dominated by RAMs whose size is proportional to $D_i * (n_s + 1)$. The remaining units are roughly proportional to the parameter U . Optimization was based on the following functions:

$$\begin{aligned} A_{D_i} &= K_{D_i} + 0.15\text{mm}^2 * D_i + 0.016\text{mm}^2 * D_i * (n_s + 1) \\ A_U &= K_U + 0.187\text{mm}^2 * U \end{aligned} \quad (1)$$

Note that we incorporated the fixed portion of the design in the constants K_{D_i} and K_u since they do not affect the optimization. Figures 4 and 5 give actual simulation results for the parameters D_i and U at $SNR = 3\text{dB}$.

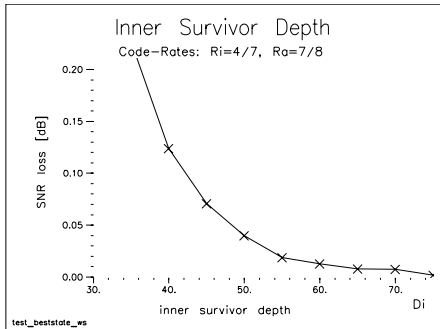


Figure 4. Effect of D_i

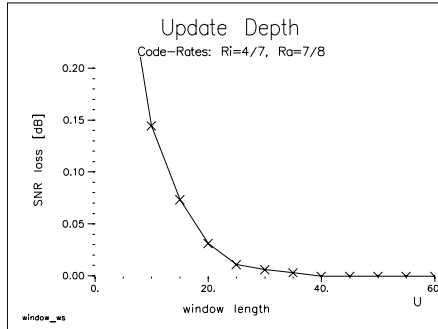


Figure 5. Effect of U

The results comply with the results from [8]. Figures 6 and 7 give the simulation results for deinterleaver parameters and outer survivor depth. Figure 6 contains graphs for variable R with C very large and vice versa.

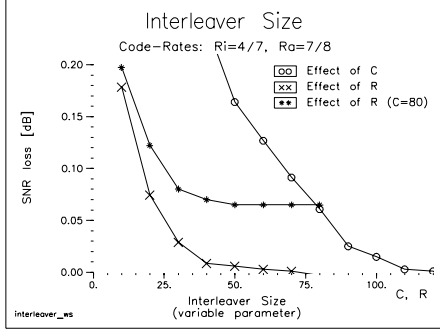


Figure 6. Effect of R and C

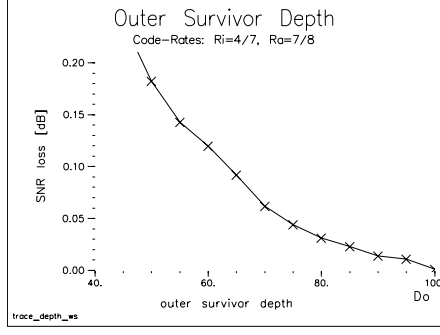


Figure 7. Effect of D_o

The figure shows clearly that an asymmetric deinterleaver should be implemented since R can be chosen much smaller than C . In addition a graph for R variable and $C = 80$ is included. The graph shows that, even in the case of the deinterleaver, the independence assumption is good, although the graphs for $C = 80$ and very large C tighten for very low R . The required memory and thus size of the RAM is obviously proportional to $R * C * (n_s + 1)$:

$$A_{il} = K_{il} + 0.002mm^2 * R * C * (n_s + 1) \quad (2)$$

For the outer SMU a block trace back architecture was chosen which is dominated by the required RAMs. The area is thus proportional to D_o :

$$A_{SMU} = K_{D_o} + 0.067mm^2 * D_o \quad (3)$$

5. Parameter Optimization

We ran an exhaustive search to find the optimum parameter set from the base data of Figures 4 – 7 in conjunction with the area models. As can be seen from the figures not all possible parameter settings were simulated. We thus ran, in a second optimization phase, a search based on linear interpolated performance data around the set of base data points to include the entire solution space into the search. Table 5 shows the obtained optimum with the consumed area portions according to the area models.

Unit(s)	Parameter	Value	Loss	Area (mm^2)
TMU _i , Q _i + ACSU _i	n_{λ_i}	5	0.005	8.6
Q _Δ , SOVA-SMU	n_s	3	0.013	17.3
	U	25	0.011	
	D_i	55	0.019	
Deinterleaver	R	15	0.122	11.9
	C	99	0.008	
TMU _o , Q _o + ACSU _o	n_{λ_o}	5	0.007	7.8
SMU _o	D_o	89	0.015	6
all	–	–	0.2	51.6

Table 5. Obtained parameter set

It becomes instantly clear from Table 5 that the parameter selection would hardly be as efficient without systematic optimization. Sixty percent of the implementation loss of $0.2dB$ was allocated to the parameter R of the deinterleaver which finally consumed 25% of the active core area of the chip. Note that the constant factors of the area models are included in the above figures to give more realistic results. Another facet of the optimization is shown in Table 6 where optimization results for various tolerable losses are given.

D_i	U	R	C	D_o	n_{λ_i}	n_s	n_{λ_o}	Loss	Area (mm^2)
74	36	21	100	100	5	3	5	0.1	63
55	25	20	90	99	5	3	5	0.15	54
55	25	15	99	89	5	3	5	0.2	51.6
53	25	10	99	100	5	3	5	0.25	48
50	24	10	100	100	4	3	5	0.3	45

Table 6. Maximum implementation loss versus area

This shows that performance can be traded versus area to a large extent which was not expected initially. Around our target loss of $0.2dB$ the parameter R is mainly affected while other parameters come into play at more relaxed or harder requirements. The effect of the parameter set of Table 5 was again verified by simulation. Figure 8 shows the result which, at an overall loss of $0.19dB$ at $SNR = 3dB$, matches the prediction very well.

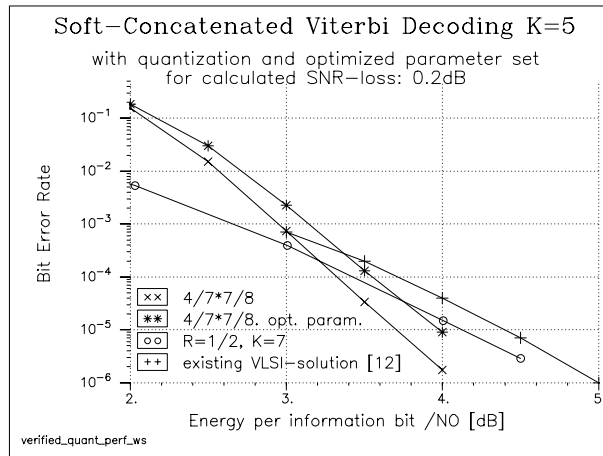


Figure 8. Verification of the overall implementation loss

But even with a less precise result we would have been able to choose from the parameter sets of Table 6 which would greatly reduce the parameter search space. To compare we included a performance graph of a commercial decoder for the reference code (taken from [12]). An honest comparison of the single coder scheme versus the concatenated scheme requires an actual area result. Although a detailed discussion of the implementation is beyond the scope of this paper and no place and route was carried out for the design, we believe that the estimates allow for realistic comparisons. This is because they include

actual results of fairly large building blocks and the global factors for wiring were chosen pessimistically. This should suffice as margin for the uncertainty of the design completion. Including a pad ring in the estimate leads to a die size of 64mm^2 ($1\mu\text{m}$ technology), whereas the decoder for the reference code [12] required 67mm^2 in $0.7\mu\text{m}$ technology. This shows that the proposed scheme is indeed competitive.

6. Conclusion

In this paper design considerations for a soft-concatenated Viterbi decoding scheme have been presented. We have shown that the proposed scheme exhibits a better coding gain compared to the well known standard code with 64 states for bit error rates better than $2 * 10^{-4}$. The scheme is thus well suited for high performance applications like HDTV. Furthermore, we have presented an optimization method which allows the systematical optimization of the implementation parameters of digital signal processing hardware. We tackled the problem of the large parameter space by separating the influence of individual parameters. Since during optimization the performance effects of parameter sets are determined by superposition of the individual effects, a final performance verification and possibly further iterations are required to find the optimum. However, in our case the independence of the parameter effects was found to be sufficient and iterative optimization was not required.

References

- [1] A. Brine, P. G. Farrell, and R. A. Harris, "Low complexity concatenated coding schemes for digital satellite communications," *Int. Journal of Sat. Comm.*, vol. 7, pp. 209–217, 1989.
- [2] W. W. Wu, D. Haccoun, R. Peile, and Y. Hirata, "Coding for satellite communication," *IEEE Journal on Sel. Areas in Comm.*, pp. 724–748, May 1987.
- [3] J. Hagenauer and P. Höher, "Concatenated Viterbi-decoding," in *Fourth Swedish-Soviet Int. Workshop on Inf. Theory*, (Gotland, Sweden), pp. 29–33, 1989.
- [4] S. Honda, S. Kubota, and S. Kato, "DSD (Double Soft Decision) concatenated FEC scheme in mobile satellite communication systems," *IEEE Journal on Sel. Areas in Comm.*, vol. 10, pp. 1271–1277, October 1992.
- [5] J. Hagenauer, E. Offer, and L. Papke, "Improving the standard coding system for deep space missions," in *Proc. of the IEEE Int. Conf. on Comm.*, pp. 1092–1097, May 1993.
- [6] J. Hagenauer and P. Höher, "A Viterbi Algorithm with Soft Outputs and It's Application," in *Proc. of the IEEE GLOBECOM*, pp. 47.1.1–47.1.7, Nov. 1989.
- [7] O. J. Joeressen, M. Vaupel, and H. Meyr, "VLSI Architectures for Soft-Output Viterbi Decoding," in *Proc. of the Int. Conf. on Appl. Specific Array Proc.*, pp. 373–384, Aug. 1992.
- [8] O. J. Joeressen, M. Vaupel, and H. Meyr, "Soft-Output Viterbi Decoding: VLSI Implementation Issues," in *Proc. of IEEE Vehicular Technology Conf.*, pp. 941–944, May 1993.
- [9] O. J. Joeressen, M. Oerder, R. Serra, and H. Meyr, "DIRECS: System design of a 100Mbit/s digital receiver," *IEE Proceedings-G*, pp. 222–230, April 1992.
- [10] Y. Yasuda, K. Kashiki, and Y. Hirata, "High-rate punctured convolutional codes for soft decision," *IEEE Trans. Comm.*, pp. 315–319, March 1984.
- [11] L. Erup and R. A. Harris, "On numerical optimization of communications system design," *IEEE Journal on Sel. Areas in Comm.*, pp. 106–125, Jan. 1988.
- [12] R. Kerr, H. Dehesh, A. Bar-David, and D. Werner, "A 25 MHz Viterbi FEC Codec," in *IEEE Custom Integr. Circuit Conf.*, pp. 13.6.1–13.6.5, May 1990.

Acknowledgement

The support we were given by the Deutsche Forschungsgemeinschaft (DFG) under grant Me 651/12-1 is gratefully appreciated.