Improving SC-FDMA Performance by Time Domain Equalization for UTRA LTE Uplink

Eva Peiker-Feil, Nicolas Schneckenburger, Werner G. Teich, Jürgen Lindner
Institute of Information Technology, University of Ulm, Ulm, Germany
eva.peiker@uni-ulm.de

***Abstract*— Single carrier frequency division multiple access (SC-FDMA) is used in the uplink of the next generation mobile communications standard Long Term Evolution (LTE). Based on a complete system simulation including channel coding and automatic repeat request, we compare a minimum-mean-square error block linear equalizer with an iterative equalization in time domain using a recurrent neural network structure. Based on the total throughput, iterative equalization in time domain shows a clear performance improvement for low signal-to-noise ratios (SNR). For a wide range SNRs turbo equalization leads to further throughput improvements.**

***Index Terms*— LTE, SC-FDMA, iterative equalization**

I. INTRODUCTION

MOBILE broadband access and high data rates for mobile data services are becoming more and more important. Long Term Evolution (LTE), the 4G successor of Universal Mobile Telecommunications (UMTS) 3G standard, offers both. It is the upcoming technique for mobile internet access.

The idea of this work is to improve the performance of single carrier frequency division multiple access (SC-FDMA), used in the LTE uplink [4]. SC-FDMA employs frequency domain equalization [4], which is compared with an iterative equalization in the time domain, using the recurrent neural network (RNN) equalizer [5]. In a third approach a combination of equalization and decoding for coded transmission, known as turbo equalization, is presented [6].

In a first step we considered bit error rates and packet error rates, but only minor improvement could be realized. But when implementing hybrid automatic repeat request (HARQ) [4] and having a closer look at the throughput, defined by the ratio of the number of received packets to the number of transmitted packets, the gain of time domain equalization can be easily seen.

II. SYSTEM MODEL

In order to obtain results applicable to a realistic scenario, the simulations were carried out using a simplified LTE communication system. First a short overview over the processing chain, based on the 3GPP standard is given [1], [2]. After that a brief description of the modulation method, SC-FDMA is given. The section ends with an overview over the HARQ employed in LTE.

A. LTE Uplink - A Short Overview

The simulations were carried out using the physical uplink shared channel (PUSCH). In LTE this channel carries general user traffic, e.g. voice or data. In order to keep complexity in a feasible range, some simplifications of the processing chain proposed in the standard were made. The processing of the source bits is shown in Fig. 1.

To the sequence of source bits first a cyclic redundancy check (CRC) is attached in order to allow error detection. This is followed by a segmentation into code blocks. Each block is connected to an own CRC in order to reduce the decoding complexity and improve error detection. The forward error correction uses a rate $\frac{1}{3}$ turbo code. The interleaver table of the turbo code is optimized for code block lengths up to 6114 which is therefore the maximum length of a code block in the segmentation. The rate matching has two major tasks: First to adjust the code rate of the coded bits according to the users and basestations requirements. Secondly the rate matching is of great importance to the HARQ in LTE. In the next step the different code blocks are concatenated again. This is usually followed by the multiplexing of control information to the data to be transmitted using an interleaver. This step is omitted in the simulation system, since the transmission parameters are assumed to be known at the receiver. This assumption has no influence on the performance of the system, since the control information has stronger protection

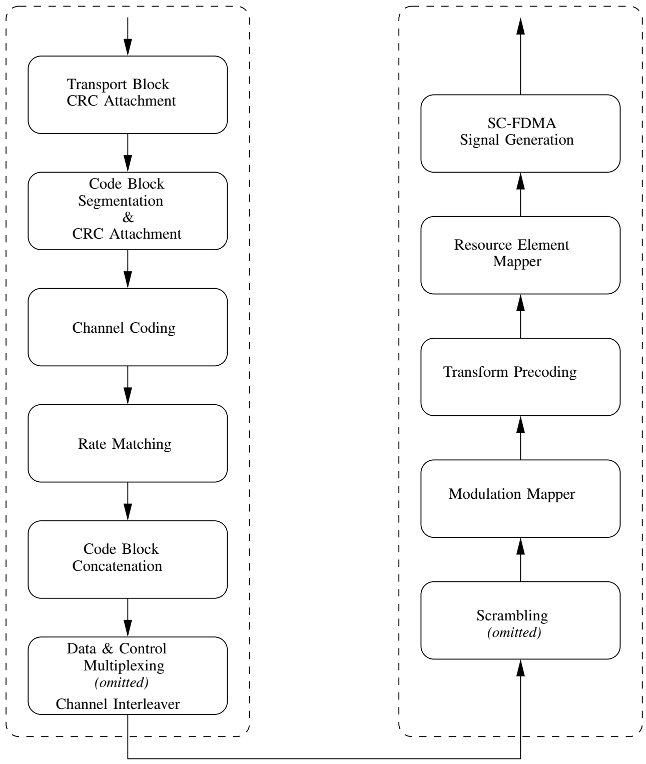

Fig. 1. Generation of the PUSCH.

against transmission errors than the actual data. Nevertheless an interleaver is used. The interleaved bits are then mapped to a modulation alphabet, either QPSK, 16QAM or 64QAM. The last step is the creation of the SC-FDMA signal, described in the following section.

B. SC-FDMA

The modulation used in the uplink of LTE is SC-FDMA, also often referred to as Fourier spread FDMA. The reason for favoring SC-FDMA in the uplink over orthogonal frequency division multiple access (OFDMA) is its reduced peak-to-average power ratio (PAPR) compared to OFDMA. The principal design of a SC-FDMA communication system is shown in Fig. 2. The incoming transmit symbols are first spread using a fast Fourier transorm (FFT). The size of the FFT depends on the number of carriers the signal is spread over. Then the spread transmit symbols are mapped on the subcarriers of the OFDMA subsystem. This can be either performed localized (i.e. to a continuous block of subcarriers) or distributed (i.e. equally spaced over the subcarriers of the OFDMA system). Although the distributed mapping exhibits a lower PAPR than the localized mapping, it is favored over the distributed mapping due to an easier scheduling between the users and smaller vulnerability against frequency offsets. The mapped and spread symbols are now transmitted over the channel using conventional OFDMA modulation: The conversion to the frequency domain is performed using an inverse FFT (IFFT), with the size of the IFFT depending on the subcarrier spacing and the total amount of bandwidth available in the system. After the transmission over the channel and the addition of white Gaussian noise, the received signal is demodulated by performing the inverse operations of the modulator.

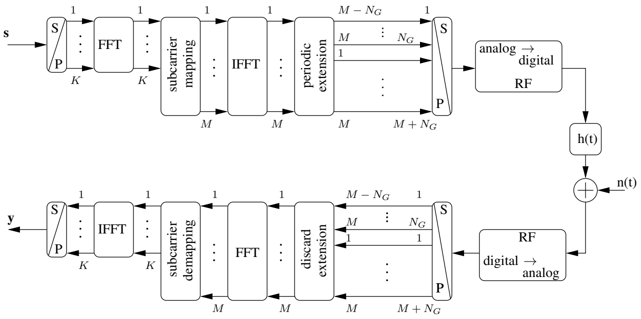

Fig. 2. Signal processing for SC-FDMA.

To keep the complexity for the simulations on a tolerable level, the maximum size of the IFFT in the transmitter is fixed to 512. Thus a maximum of 300 subcarriers are usable according to the specifications.

C. Automatic Repeat Request in LTE

The task of automatic repeat request (ARQ) in a communication system is the backward error correction. If a packet was corrupted by the transmission, it is automatically retransmitted. In a conventional ARQ the data of an erroneous transmission is discarded. In LTE a HARQ scheme is employed, in which all previous erroneous transmission attempts of a packet are combined with the current attempt. In LTE in every transmission attempt of a packet, a different representation of that packet is transmitted. The different representations, also referred to as redundancy versions, are created in the rate matching block. As stated above, the rate matching is used to adjust the code rate of the sequence output by turbo coder. The encoder has always a fixed code rate of $\frac{1}{3}$. Thus to obtain a rate of, e.g. 0.8, 60% of the coders output has to be punctured. The different representations are created by puncturing different bits of the turbo coders output in every transmission attempt. Thus a retransmission of a packets translates to an decrease of the code rate. Additionally, the multiple transmission of one bit effectively results in a rise of the SNR.

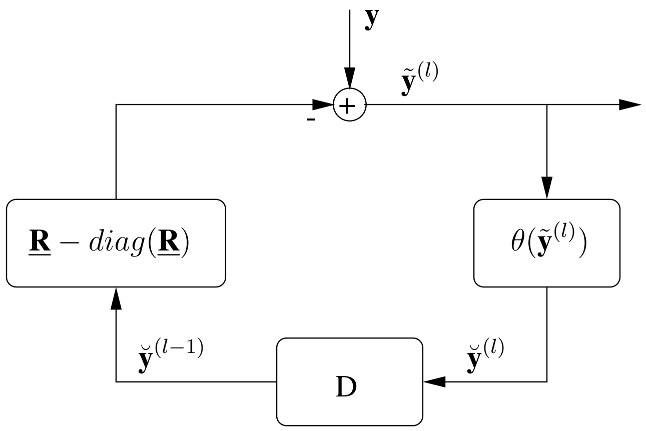

Fig. 3. Iterative interference cancellation: block diagram of the RNN equalizer.

The selection of the bits to be transmitted is performed as follows [2]: The bits output by the coder are interleaved and written into a cyclic buffer. In every transmission attempt a continuous block of bits from that buffer is transmitted, starting at the so called redundancy version starting point. Note, that due to the turbo codes sensitivity to the systematic bits, only very few systematic bits are punctured in the first transmission attempt.

D. Equalization

1) RNN: Each received symbol $\tilde{\mathbf{y}}_i$ of a block can be written as a sum of the useful part of the symbol, interference and noise (Eq. 1):

$$\tilde{y}_i = \underbrace{r_{ii}y_i}_{\text{useful part}} + \underbrace{\sum_{i=1,i\neq j}^{N} r_{ij}y_i}_{\text{interference}} + \underbrace{n_{ci}}_{\text{noise}}. \tag{1}$$

Where r_{ij} are the elements of the discrete-time channel matrix $\underline{\mathbf{R}}$ [9].

The idea of the RNN equalizer [5] is to estimate the interference and subtract it from the received symbol. Fig. 3 shows the block diagram of an RNN equalizer. The iteration equation is given by:

$$\tilde{y}_k^{(l)} = y_k - \sum_{j=1}^{K} \frac{r_{kj}}{r_{kk}} \breve{y}_k^{(l-1)} \tag{2}$$

$$\breve{y}_k^{(l)} = \theta(\tilde{y}_k^{(l)}) \tag{3}$$

where $\breve{y}_k^{(l)}$ is the soft decision of the received symbol $\tilde{y}_k^{(l)}$ at iteration step (l) and $\theta(\cdot)$ a nonlinear activation function. After the final iteration step log-likelihood ratios (LLRs) [7], [8] are calculated and passed to the turbo decoder.

The activation function $\theta(\cdot)$ builds the core of the RNN equalizer. The optimum activation function for complex-valued symbol alphabets has been derived by Sgraja et al. [10]. For binary phase shift keying the well-known hyperbolic tangent function is obtained. Based on the remaining interference power and the noise power the optimum activation function is calculated in each iteration step. For the tangent hyperbolic function the remaining degree of freedom to be adjusted is the slope at the point of inflection. Generally the steepness of the function (the slope in case of the hyperbolic tangent) will increase with increasing iteration number. This reflects the improving reliabilty of the soft symbol estimates due to the successive interference cancellation.

The updating of the subchannels is also of great influence on the RNN's performance. There are two possibilities: the serial update and the parallel update. For parallel updating all subchannels are processed together and only information from the previous iteration is used. If serial updating is used, the information of each subchannel is updated separately, i.e. the interference from one subchannel is calculated and subtracted from the received vector. This operation is done for all subchannels. The advantage of serial updating is, that in contrast to parallel updating, where only information from the last iteration is used, information, derived from the current iteration, is also processed.

2) Turbo Equalization: Turbo equalization was first introduced for intersymbol interference channels by Douillard et al. in [12]. Egle [11] proposed the combination of the RNN equalizer and decoding for turbo equalization. Fig. 4 presents the way how turbo equalization is applied. The basic idea is that the serial updated RNN for equalization and the BCJR algorithm for decoding ([13]) benefit from each other. Both offer soft information at the output. The RNN processes the received vector $\mathbf{y}$ with a fixed number of iterations. $\mathbf{L}_{e,E}^{(l)\,\prime}$, a L-value vector is calculated from the soft output information of the RNN equalizer $\tilde{\mathbf{y}}^{(l)}$. $\mathbf{L}_{e,E}^{(l)\,\prime}$ is deinterleaved, resulting in $\mathbf{L}_{e,E}^{(l)}$.The sequence is decoded and LLRs for both the information and the code bits are calculated: $\mathbf{L}_{D}^{(l)}$. To generate extrinsic LLRs, the input of the decoder is subtracted from its output: $\mathbf{L}_{e,D}^{(l)} = \mathbf{L}_{D}^{(l)} - \mathbf{L}_{e,E}^{(l)}$. This is done to guarantee, that the decoder only passes new knowledge through the iteration. The extrinsic interleaved $\mathbf{L}'_{e,D}$ LLRs are then used as a-priori input and the loop starts again.

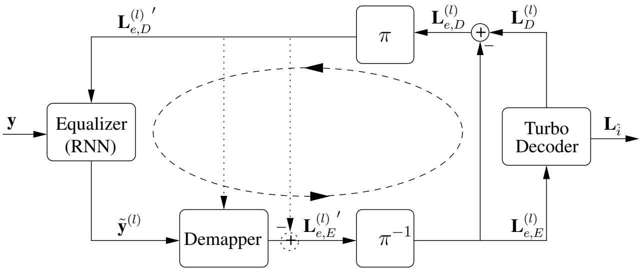

Fig. 4. Block diagram of a turbo equalizer.
π: interleaver; π^{-1}: deinterleaver.

E. MMSE

The design goal of the minimum mean-square error (MMSE) equalizer is to minimize the mean square error of the equalized and transmitted signal by taking second order statistics of the channel and the signal into account The equalization matrix for the MMSE block linear equalizer (MMSE-BLE) is given as

$$\underline{\mathbf{A}}_{\text{eq,MMSE}} = (\underline{\mathbf{R}} + \frac{\sigma^2}{\sigma_s^2}\underline{\mathbf{I}})^{-1} \qquad (4)$$

with σ_s^2 being the transmit power of the signal $\mathbf{s}$. Thus if $\underline{\mathbf{R}}$ is a diagonal matrix, the matrix inversion in Eq. 4 corresponds the reciprocal value of each diagonal element and the noise. Hence the complexity in the frequency domain is very low. However, the MMSE-BLE still suffers from noise enhancement.

III. SIMULATION RESULTS

A. Simulation Setup

The transmission bandwidth of 5 Mhz translates to a total number of $M = 512$ subcarriers available in the OFDMA subsystem. The number of subcarriers occupied for the transmission is fixed to $K = 300$. For all simulations the extended vehicular A (EVA) model from the LTE standard is used. The power delay profile [3] is assumed to be known perfectly at the receiver. The channel imupulse response is assumed to be time invariant within one slot, i.e. for seven SC-FDMA symbols. The cyclic prefix is chosen sufficiently long. The output of the turbo coder is always punctured to a code rate of $\frac{1}{2}$. The modulation alphabet used is the 16QAM alphabet from [1].

The following parameters are used in the receiver: A fixed number of eight iterations are performed in is used for the RNN, if implemented in a conventional receiver structure. For both the MMSE-BLE and conventionally implemented RNN eight iterations are performed in the turbo decoder. The turbo equalizer uses two inner iterations, each with four RNN and four turbo decoder iterations.

B. Results

Fig. 5 shows the transport block error rate (TBER) for the three equalizers. Both implementations of the RNN equalizer show a very good performance for a low signal to noise ratio and both significantly outperform the MMSE-BLE. The turbo equalization shows slight advantages compared to the conventional receiver structure. For a higher SNR the MMSE-BLE outperforms the RNN equalizer due to an error floor. This error floor is caused by the RNNs tendency to converge against a wrong solution in case of very poor representatives of the channel impulse response. Due to the statistical channel model used some channels exhibit a very high frequency selectivity and thus create a vast amount of interference. Simulations carried out for different code rates and other LTE channel models show similar results.

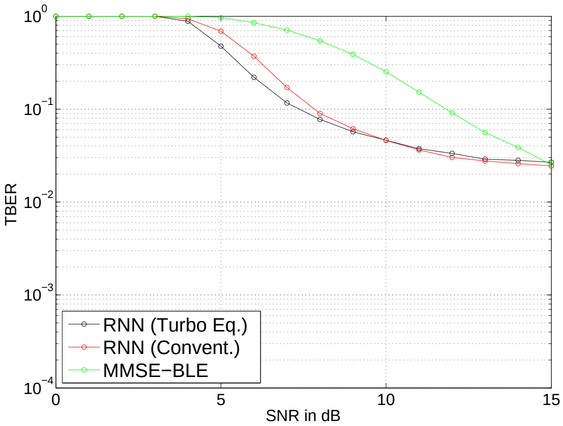

Fig. 5. Transport block error rates for the for a 16QAM modulation and a code rate of $r_c = 0.5$ for the EVA channel model for the different equalizers.

In Fig. 6 the throughput η for the time and frequency domain equalizers is shown. Considerable performance gains can be realized by the application of time domain equalization. At a throughput of $\eta = 0.9$ a gain of roughly $5dB$ is realized by the RNN equalizer. The time domain equalization shows superior performance compared to the MMSE-BLE until a SNR of about $15dB$. Then the influence of the error floor makes the MMSE-BLE outperform the RNN equalizer. When comparing the turbo equalizer with the conventional RNN equalizer, a gain of roughly $1dB$ may be realized in the SNR region between 4 to $8dB$. Overall the benefits from the time

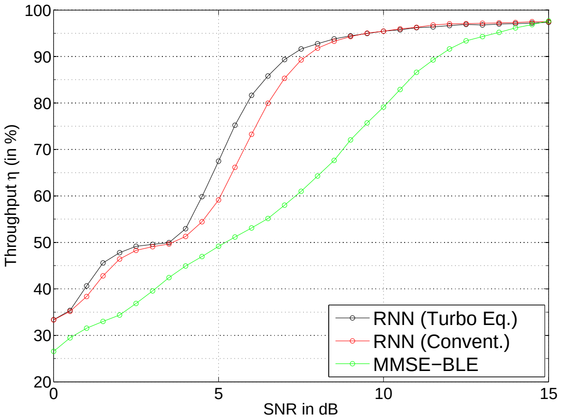

Fig. 6. Throughput for the EVA model with a code rate of $r_c = 0.5$ using a 16QAM alphabet for the different equalizers.

domain equalization may become more apparent if throughput rates are studied.

The total throughput for 16QAM on the EVA channel model are shown in Fig. 7. A larger gain in the lower SNR regions can be achieved using the RNN equalizer. Until a SNR of $15dB$, the RNN equalizer, implemented in a turbo or conventional receiver structure, can improve the throughput by up to 30%. For higher SNRs the MMSE-BLE outperforms the RNN equalizer. Especially using the turbo equalization implementation of the RNN, after $9Mbit/s$ the throughput does not improve any further. However, modifying the scheduling of the multidimensional turbo iteration process, further performance gains are possible. This can be seen form the fact that the conventionally implemented RNN equalizer achieves a throughput of up to $10Mbit/s$.

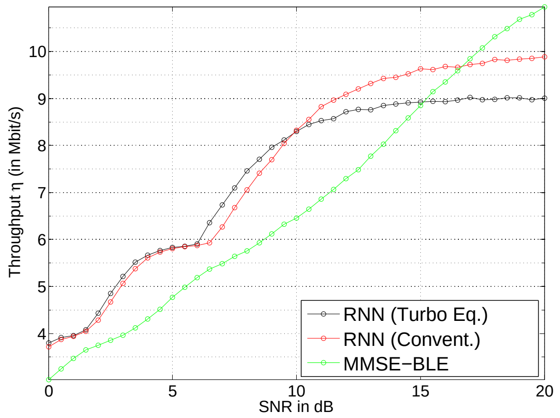

Fig. 7. Maximum throughput of the different equalizers using 16QAM modulation and the EVA channel model.

IV. CONCLUSIONS

Based on a complete system simulation for the next generation mobile communications standard LTE, we have compared three receiver concepts: frequency domain MMSE-BLE, iterative interference cancellation in time domain (RNN equalizer), and turbo equalization. Based on the total throughput, the RNN equalizer gives a substantial improvement of up to 30%. For a wide range of SNR values, turbo equalization leads to further improvements. Preliminary results show, that these results do not change, when channel estimation is included in the system model.

REFERENCES

[1] 3GPP, *Technical report 36.211 v8.7.0 (radio access networks)*, 3rd Generation Partnership Project, 2009.

[2] 3GPP, *Technical report 36.212 v8.7.0 (radio access networks)*, 3rd Generation Partnership Project, 2009.

[3] 3GPP, *Technical report 36.141 v8.5.0 (radio access networks)*, 3rd Generation Partnership Project, 2009.

[4] Farooq Khan, *LTE for 4G Mobile Broadband - Air Interface and Performance*, Cambridge University Press, 2009.

[5] W.G. Teich, M.Seidl, Code Division Multiple Access Communications: Multiuser Detection Based on a Recurrent Neural Network Structure, *IEEE ISSSTA*, vol.3, Mainz/Germany, September 1996, pp. 979-984.

[6] M. Tüchler, R. Koetter, A. C. Singer, Turbo equalization: Principles and New Results, *IEEE Transactions on Communications*, Vol. 50, No. 5, May 2002.

[7] J. Hagenauer, E. Offer, L. Papke, Iterative Decoding of Binary Block and Convolutional Codes, *IEEE Transactions on Information Theory*, vol. 42, no. 2. pp. 429-445, March 1996.

[8] B. Sklar, A Primer on Turbo Code Concepts, *IEEE Communications Magazine*, pp. 94-101, Dec. 1997.

[9] J. Lindner *Informationsübertragung: Grundlagen der Kommunikationstechnik* Springer, Berlin, 2004

[10] C. Sgraja, A. Engelhart, W.G. Teich, J. Lindner, Equalization with Recurrent Neural Networks for Complex-valued Modulation Schemes, *Proc. 3. Workshop Kommunikationstechnik* Schloss Reisensburg 1999, pp. 7-12, Günzburg, Germany, July 1999.

[11] J. Egle, *Detection Methods for Power and Bandwidth Efficient Single Carrier Block Transmissions*, Phd thesis, University of Ulm, VDI Verlag GmbH, Reihe 10, Nr. 738, 2004.

[12] C. Douillard, M. Jézeqzél, C. Berrou, A. Picart, P. Didier, A. Glavieux, Iterative correction of intersymbol interference: Turbo-equalization, *European Transactions on Telecommunications,* vol. 6, 1995, pp. 507-511.

[13] L. Bahl, J. Cocke, F. Jelinek, J. Raviv, Optimal Decoding of Linear Codes for Minimizing Symbol Error Rate, *IEEE Transactions on Information Theory,* 20(2): 284-287, 1974.