

BER Estimation for wireless links using BPSK/QPSK modulation

Lodewijk T. Smit, Gerard J.M. Smit and Johann L. Hurink
Department of Electrical Engineering, Mathematics & Computer Science,
University of Twente, Enschede, the Netherlands
email:L.T.Smit@utwente.nl

Abstract—This paper introduces a method that computes an estimation of the bit error rate (BER) based on the RAKE receiver soft output only. For this method no knowledge is needed about the channel characteristics nor the precise external conditions. Simulations show that the mean error of the estimation is below 2%, with only a small variance. Also an estimation of the BER for a different spreading factor or a different number of RAKE finger can be made. Implementation issues for a practical use of the method are discussed.

Keywords: BER estimation, WCDMA

I. INTRODUCTION

In this paper we introduce a method to compute an estimation of the bit error rate (BER) of a wireless channel. The presented method is used to estimate the current quality of the wireless channel using the data received by a RAKE receiver [1]. This information can be used to adapt the receiver to obtain the desired Quality of Service (QoS) for a given application or protocol with minimal computational effort. This reduction in computational effort can be translated to a reduction in energy consumption for a mobile terminal or to a reduction in the amount of resources for a base station.

Applications or protocols demand a certain QoS that translates to a certain quality (BER) of the wireless link. In general, two principles are used to determine the quality of the output of the RAKE receiver. First, known sequences of (pilot) symbols are transmitted in parallel with the data, so the BER can be determined. Second, models are used which require the current status of the environment as input to compute the BER. In this paper, we use an alternative (third) method. We apply statistical methods on the soft output of the RAKE receiver, to compute the BER without additional knowledge of the current environment or transmission of extra pilot symbols.

To illustrate our approach we give an example of the application of our BER estimation algorithm in a wide-band code division multiple access (WCDMA) system [2]. The output of the RAKE receiver is used as

input to an (adaptable) forward error correction (FEC) turbo decoder as depicted in Figure 1. The used turbo FEC operates on a sequence of bits, grouped in a block. Given the number of errors per block, we can predict whether the used forward error decoder is able to correct the received block [3] or not. In our adaptive system [4], the spreading factor used by the WCDMA transmission is decreased until the limit of the error decoding capacity of the FEC decoder is reached. Decreasing the spreading factor leads to a higher bitrate, which has two main advantages. First, a certain amount of data is transmitted as fast as possible, providing a good QoS for the user. Second, the transmitter and receiver can be switched off earlier, saving power, which is especially useful for a mobile handheld terminal.

The frame error rate (FER) after the FEC decoder also gives an indication of the quality of the received signal. However, the BER estimation after the RAKE receiver gives much more information about the quality. There are two reasons for that. First, we not only know whether the turbo decoder is able to correct the received frame or not, but we also know whether the quality of the received frame is near the turbo decoder error correcting capacity or there is room for improvement. In this way we can predict how much the quality of the output of the RAKE receiver should be improved or decreased (e.g. by changing the spreading factor) so that the turbo decoder is just able to correctly decode the received frames. Second, we can predict what will happen when we change parameters. For example, if we plan to change the spreading factor of the RAKE receiver,

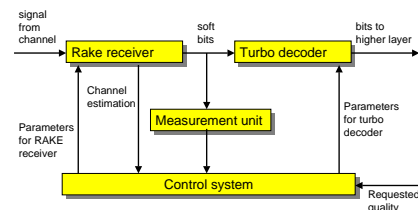


Fig. 1. The Control System Of The Terminal

we are able to predict the resulting BER and FER. Thus, we can predict whether the quality of the output of the RAKE receiver after reduction of the spreading factor is still good enough such that the turbo decoder can still correct most of the frames and what the consequences are for the QoS (e.g. latency, throughput, BER).

Our BER estimation algorithm gives detailed information about the quality of the signal and as we know the characteristics of the forward error decoder, we can make a careful trade-off between the different parameter settings that are possible at physical layer (RAKE receiver), link layer (FEC decoder) and transport layer (e.g. retransmissions) of the network protocol stack. This cross-layer approach ensures a global optimization, with potential higher savings compared to optimizations performed per individual layer of the network protocol stack.

In this way, we can minimize the energy consumption and/or the use of resources at run-time, while satisfying an adequate QoS, which is requested by the end user of the system.

Section two describes related work. Section three derives and explains the method for BER estimation. Section four presents simulation results giving the difference between the real BER and the estimated BER. Section five discusses how to implement the presented method in hardware, followed by the conclusions in the last section.

II. RELATED WORK

In general, the BER is not known at the receiver side, because the original transmitted data is unknown. A commonly used method to compute the BER is to use pilot symbols. Pilot symbols represent a predefined sequence of symbols, which are known at the transmitter and the receiver side. Therefore, the BER can be computed for these pilot symbols. Third generation telephony uses for instance pilot symbols [5]. This approach has several disadvantages. First, the transmission of the pilot symbols introduces overhead. Second, the BER is only computed over a small amount of the total bits that are transmitted. Third, the BER of the pilot symbols may differ from the BER of the data.

Another approach is to model the channel with all the known effects, e.g. [6]. A state of the art article on this area is [7]. Using this method it is possible to achieve accurate BER estimations for the modeled channel. However, the actual properties of the channel and the modeled effects can differ significantly from the constructed model. Also, effects that are not modeled can happen in real situations. In practice, it is not possible to model all the different effects that cause the disturbance of the wireless channel. Estimation of the exact quality of the signal of the wireless channel is therefore impossible.

Our approach differs significantly from the two mentioned approaches. We only use the soft output from the rake receiver, and require no additional information about the channel. In our opinion, it doesn't matter which physical effect is responsible for the degradation of the signal to determine the BER. The advantage is that an accurate estimation can be made independent of the unpredictable dynamic changing external environment.

III. BER ESTIMATION

In an ideal situation, without disturbance of the channel, the output of the soft value of the rake receiver is equal to the used sf (spreading factor) for a transmitted bit with value one. Similarly, for a bit with value zero (represented by minus one), the soft output of the rake receiver is $-sf$. This perfect situation is shown in Figure 2.

In case of disturbance of the channel, the sampled chip values are no longer exactly equal to one or minus one, but can be higher or lower. A lot of external causes may be responsible for this disturbance. Most effects that change the signal can be modeled with a normal distribution. For example, AWGN behavior and the fact that the spreading codes of other channels are not perfectly orthogonal, can be modeled with a normal distribution. A few effects, e.g. fading, do not behave like a normal distribution. However, the central limit theorem [8] states that regardless of the type of distribution, the distribution will approximate a normal distribution, if the number of samples is large (>30). Therefore, we can approximate the values of the soft values of the output of the RAKE receiver with a normal distribution. One soft output value is composed of different chip values. If the number of chips per bit is higher, a better approximation of the normal distribution is made.

Figure 3 shows the expected normal distribution behavior for the soft output values of the RAKE receiver for a pretty good channel. When the channel becomes worse, the mean will not change (significantly), but the standard deviation will increase, as shown in Figure 4. Some bits are received incorrect in this figure. All soft values > 0 are considered to be transmitted ones and all soft values < 0 are considered to be transmitted zeros. Figure 5 shows the effect for an extremely bad channel. As can be seen from the figure, the two distributions are heavily mixed up. Every bit with value one that is received with a negative soft output is received incorrectly and also the positive soft output for a transmitted bit with value zero is received incorrectly. The marked area in Figure 5 is the probability that a bit is received incorrectly.

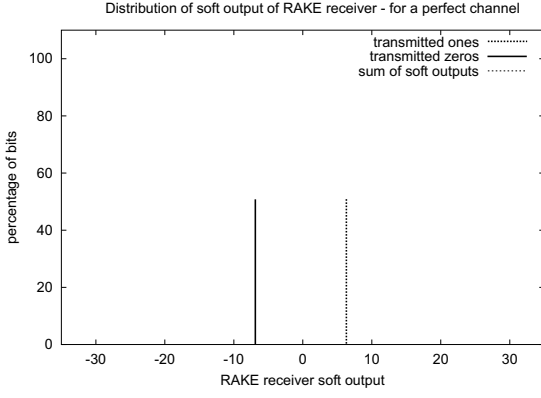


Fig. 2. Perfect channel

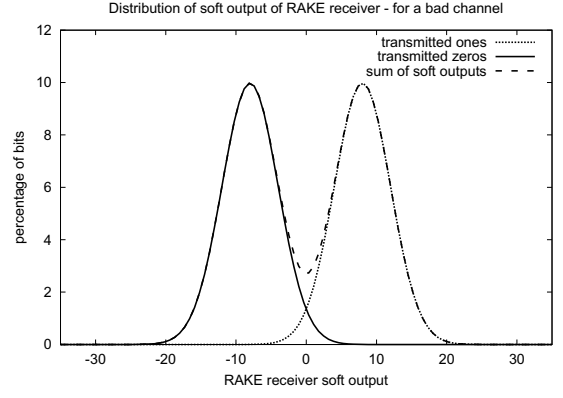


Fig. 4. Bad channel

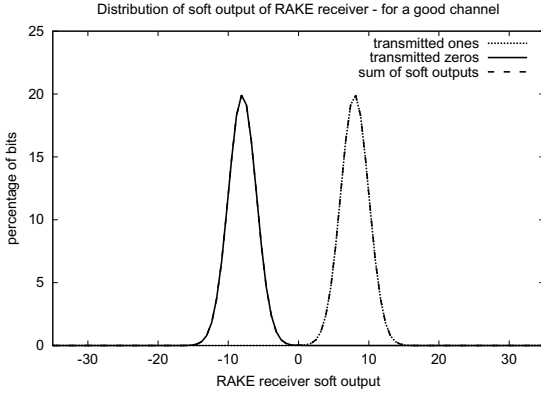


Fig. 3. Good channel

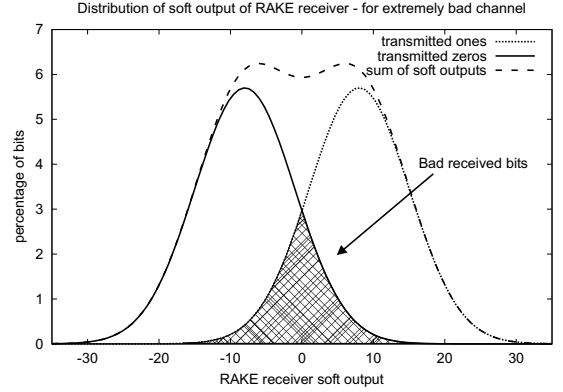


Fig. 5. Very bad channel

In reality, the distribution is not as ideal as the distribution shown in Figures 2 to 5. Figure 6 shows the soft output values of one transmitted block (3012) bits. To plot the distribution, all the soft values are rounded to the nearest integral number to make classes. Figure 6 shows the distributions for the transmitted ones and zeros.

Unfortunately, the receiver can not determine whether a soft value belongs to the 1-distribution or to the 0-distribution. The soft output of the RAKE receiver is the addition of the 1-distribution and the 0-distribution, which is also plotted in Figures 2 to 6 as a dotted line.

Our goal is to predict the bit error rate (BER), i.e. the size of the marked area in Figure 5. Let $X(Y)$ denote the distribution of the soft output values of the transmitted zeros (ones). Using these distributions, the BER can be expressed by:

$$BER = pP(X \geq 0) + (1 - p)P(Y \leq 0). \quad (1)$$

where p denotes the probability that a zero is transmitted.

Since both distributions are mirrored to the zero axis and due to the mentioned assumption, X and Y can be

expressed in terms of a standard normal distribution:

$$X = \sigma Z - \mu. \quad (2)$$

$$Y = \sigma Z + \mu. \quad (3)$$

where Z denotes the standard normal distribution, μ the mean and σ the standard deviation. Using this, the BER reduces to:

$$BER = P(X \geq 0) = P(Z < \frac{\mu}{\sigma}) = \Phi(-\frac{\mu}{\sigma}). \quad (4)$$

where $\Phi(z)$ is the function that gives the area of the standard normal distribution to the right of z , i.e. the probability that a value is smaller than z . The function $\Phi(z)$ is widely available in tabular form.

We want to get a prediction of μ and σ based on the soft output values of the RAKE receiver. Using the soft output values, we derive estimates $\hat{\mu}$ and $\hat{\sigma}$ for μ and σ respectively.

Note that if there are only effects with a normal distribution (like disturbance of other users, AWGN, etc), $\hat{\mu}$ will be equal to the spreading factor. However, for other effects (e.g. fading effects like Doppler), the $\hat{\mu}$ can differ significantly from the spreading factor.

As mentioned before, the received soft output values of

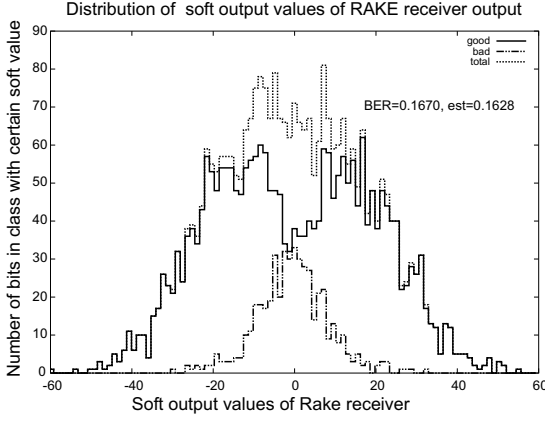


Fig. 6. Output of RAKE receiver for bad channel

the RAKE receiver do not correspond to the distribution X and Y , but to a distribution W , which results from the combination of the distributions X and Y (with probability p we get distribution X and with probability $(1 - p)$ distribution Y). For W we have:

$$P(W \leq w) = pP(X \leq w) + (1 - p)P(Y \leq w). \quad (5)$$

Based on measured results for W and using moments of distributions, it is possible to estimate the characteristic values μ and σ of the distributions X and Y , which together form distribution W (see [9]). If r is a positive integer, and if X is a random variable, the r th moment of X is defined to be $m_r(X) \equiv E(X^r)$, provided the expectation exists, see [10]. For a standard normal distribution, the first, second, third and fourth moments are respectively zero, one, zero and three. The first and third moment of Z are zero and can not be used to compute the two unknown variables $\hat{\mu}$ and $\hat{\sigma}$. Therefore the second and fourth moment of W are used.

The second moment of W is:

$$m_2(W) = p(E(X^2)) + (1 - p)(E(Y^2)). \quad (6)$$

Scrambling (used in almost every wireless communication system) ensures that approximately an equal number of ones and zeros are transmitted. This means that $p \approx \frac{1}{2}$. Setting $p = \frac{1}{2}$, and using equations (2), (3) and the moments of the standard normal distribution, equation (6) becomes:

$$m_2(W) = \mu^2 + \sigma^2. \quad (7)$$

therefore,

$$\sigma^2 = m_2(W) - \mu^2. \quad (8)$$

The fourth moment of W is:

$$m_4(W) = p(E(X^4)) + (1 - p)(E(Y^4)). \quad (9)$$

With $p = \frac{1}{2}$, this equation becomes:

$$m_4(W) = \mu^4 + \binom{4}{2} \mu^2 \sigma^2 + \sigma^4 E(Z^4). \quad (10)$$

Substituting the moments of Z gives:

$$m_4(W) = \mu^4 + 6\mu^2 \sigma^2 + 3\sigma^4. \quad (11)$$

Replacing σ^2 with (7) and simplifying yields:

$$\mu^4 = \frac{3}{2}(m_2(W))^2 - \frac{1}{2}m_4(W). \quad (12)$$

So,

$$\mu = \sqrt[4]{\frac{3}{2}(m_2(W))^2 - \frac{1}{2}m_4(W)}. \quad (13)$$

Using Equations (8) and (13) in combination with the estimation of the second and the fourth moments of W based on the individual samples $V_1..V_n$ of the output of the rake receiver, the Formulas (14) and (15) can be derived for the estimators $\hat{\mu}$ for the mean and $\hat{\sigma}$ for the standard deviation:

$$\hat{\mu} = \sqrt[4]{\frac{3}{2} \left(\frac{\sum_{i=1}^n V_i^2}{n} \right)^2 - \frac{1}{2} \frac{\sum_{i=1}^n V_i^4}{n}} \quad (14)$$

$$\hat{\sigma} = \sqrt{\frac{\sum_{i=1}^n V_i^2}{n} - \hat{\mu}^2} \quad (15)$$

Finally, the BER estimation can be computed with:

$$\widehat{BER} = \Phi \left(-\frac{\hat{\mu}}{\hat{\sigma}} \right) \quad (16)$$

A. Bias in the Estimators

In the previous section we computed the r th moment for the stochastic variable X from the samples $W_1 \dots W_n$ with $E(X^r) = \frac{1}{n} \sum_{i=1}^n W_i^r$. However, the r th moment is slightly higher due to bias. The correct estimates without bias for the r th moment M_r are [11]:

$$M_2 = \frac{n}{n-1} m_2 \quad (17)$$

$$M_4 = \frac{n(n^2 - 2n + 3)}{(n-1)(n-2)(n-3)} m_4 - \frac{3n(2n-3)}{(n-1)(n-2)(n-3)} m_2^2. \quad (18)$$

where m_r is equal to $\frac{1}{n} \sum_{i=1}^n W_i^r$, with n the number of samples. When the number of samples is large, the

difference between m_r (r th moment with bias) and M_r (r th moment without bias) becomes negligible small. E.g. for $n=1000$ samples, the difference is about 0.1 percent for the second moment. Since, we keep n large, we may neglect the correction terms in order to obtain a substantial easier computation of the moments. When n is small, one should consider to include the correction terms mentioned above. Note that the correction terms have to be computed only once per frame.

IV. RESULTS

In our simulation environment we performed several simulations with a realistic time-variant channel. In successively simulations, the number of simultaneously transmitting users, the number of paths and the amount of added white Gaussian noise (AWGN) is changed. All simulations uses blocks with 1000 randomly generated turbo encoded bits, making a block size of 3012 bits. For each received block, the real BER is determined and compared with the estimated BER. The reported estimation error is the absolute difference between the estimated BER and the real BER (expressed in %); i.e.,

$$est\ error = |BER_{EST} - BER_{REAL}| * 100\%. \quad (19)$$

The estimation error is reported as the absolute difference, because the relative difference can be very high with a low BER. For example, if 2 errors (BER=0.002) are estimated for a block with 1000 bits and the block contains 1 error (BER=0.001) the relative difference is large, while the absolute difference is only 0.1%. For our application, we are interested in the absolute difference.

In Figure 7 the mean estimation error is depicted, as function of classes with a width of 0.01 of the real BER of the received block (e.g., the estimation error of all blocks with a real BER in the range [0.15, 0.16) are summed up and divided through the number of blocks in the class to get a mean estimation error). For a specific case, the presented results show that the estimation is better for a lower spreading factor. Having a specific BER, a lower spreading factor means a better channel than the same BER for a higher spreading factor. Therefore, the estimation works better for a better channel. In addition to the average estimation error, information about the variance in the estimation is relevant, because the estimation will be worthless if the variance is too high. In Figure 8, the estimation error for $sf = 8$ is depicted. Beside the mean of the estimated BER also the variance is given. For each BER class, the interval $[\mu - \sigma, \mu + \sigma]$ is given. Given this figure, we can conclude that, for a real BER below 0.2, a good prediction is possible with an error of at most 2%. We are not interested in $BER > 0.2$, because blocks with a $BER > 0.2$ can not be corrected by a FEC decoder (e.g.

a turbo decoder). The same kind of simulations have been performed for different scenarios, e.g. Rayleigh fading channels, different amount of users, different amount of paths, etc. The achieved results were similar to the ones given in the Figures 7 and 8.

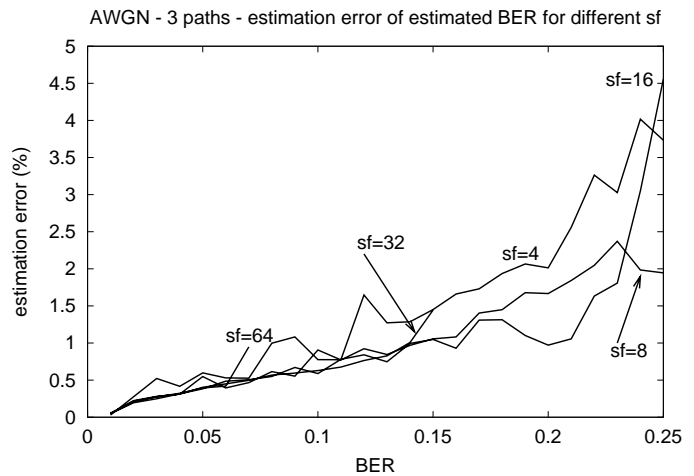


Fig. 7. Estimation Error, for different Spreading Factors

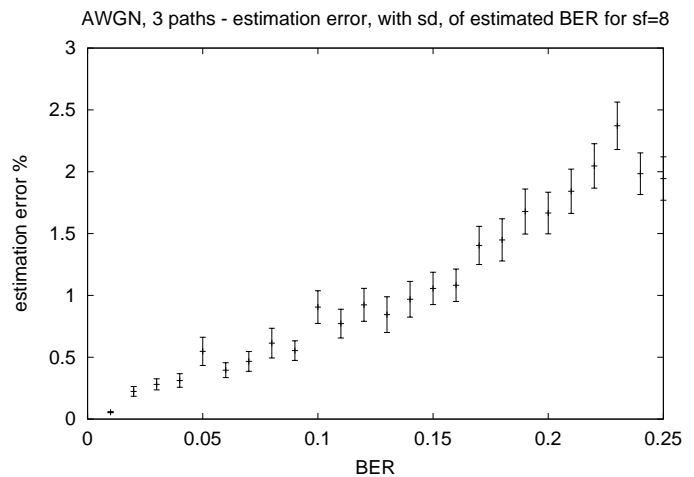


Fig. 8. Estimation Error and its Variance displayed for $sf = 8$ ($[\mu - \sigma, \mu + \sigma]$ for each class)

A. External Validation

To validate the results of our simulation and to verify the proper working of our algorithm, Ericsson Eurolab in Enschede did an additional set of simulations. They got only Formulas (14) – (16) without additional information. Ericsson used their UMTS simulator and tried to estimate the BER with the Formulas (14) – (16) and the soft output of their simulator. Two different channels have been simulated: AWGN and an Ericsson proprietary channel that is very realistic with multiple users, multiple paths, power variations, etc. Figure 9 shows the soft output values of

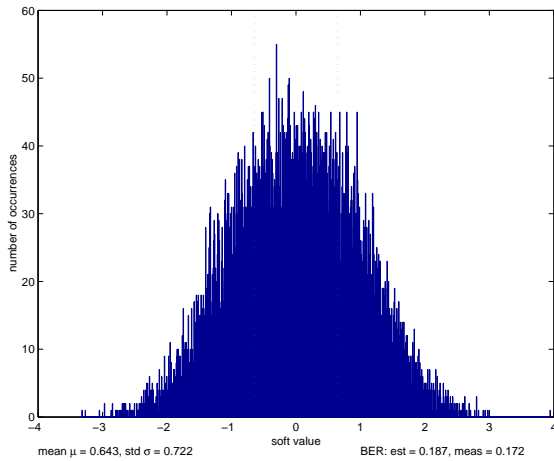


Fig. 9. DCH, real. channel, 120k samples

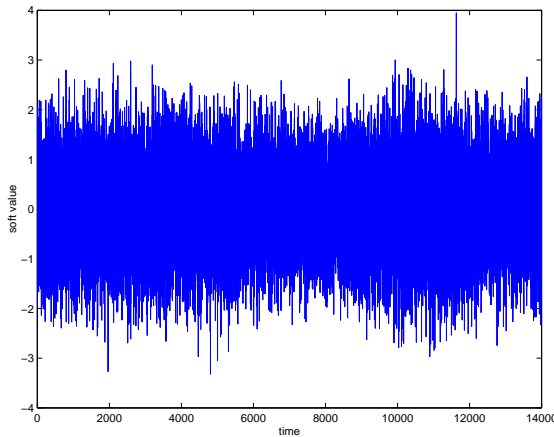


Fig. 10. DCH, real. channel, data

the RAKE receiver. The estimated BER is 0.187 and the real BER is 0.172. Figure 10 shows the accompanying diagram of the received power. The estimation error for the Ericsson proprietary channel was about 1.7% and the estimation error for the AWGN channel was even lower. As expected, the BER estimation algorithm gives less accurate results when power control is disabled. However, the whole performance of WCDMA depends on a good power control.

V. IMPLEMENTATION

The proposed method is simple and the involved Formulas (14) and (15) can be implemented easily on an ALU (e.g. an ARM). In this section we give some considerations how a real implementation can be made on an ASIC or reconfigurable architecture.

Figure 11 shows a very simple hardware support, which can be used to compute the terms $\sum_{i=1}^n W_i^2$ and

$\sum_{i=1}^n W_i^4$, that must be done at a speed that is equal

to the incoming bit rate (maximal 2Mbit/s in case of UMTS). The structure consists of two look-up tables (LUT), two adders and two registers and is meant to do the computation streaming, while the samples are coming from the RAKE receiver. The LUTs are used to look up the power of two and the power of four of the incoming sample. The result from the LUT is added to the subtotal of the previous additions that is stored in the register. At begin of the reception of a new block, the register is initialized to zero, and at the end of a block, the content of the register is passed to the output. In real implementations, the soft output from the RAKE receiver is quantized with a limited number of bits. A quantization with more of 8 bits is not useful, because there is no observable SNR degradation. Supposing 6 bits that represent a signed soft value, the LUTs can be limited to $2^5 = 32$ entries because the sign bit can be ignored. This proposed structure can be implemented in dedicated hardware or an FPGA.

The remainder of the computation of the formula can be done after finishing the computation of the summation. Note that the speed of this computation can be much lower, because this has to be done only once per received block. Therefore, this computation can be done by a general purpose processor, like an ARM. If everything has to be done in dedicated hardware or FPGA, division by n and multiplication by $\frac{1}{2}$ and $\frac{3}{2}$ can be done by shifting, if only the first 2^k samples of all samples of a block are used, where k is as large as possible. The square roots can be stored in a LUT.

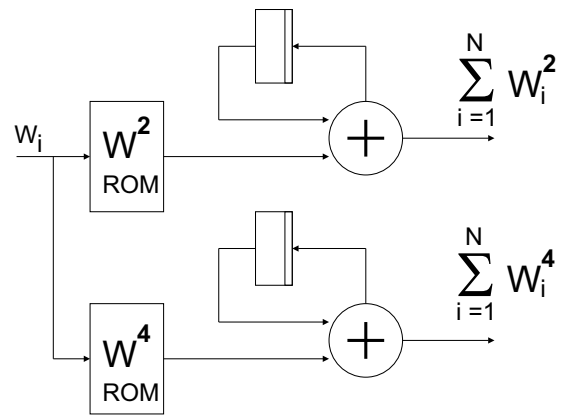


Fig. 11. Hardware support for BER estimation

VI. WCDMA SPECIFIC ESTIMATIONS

Section III showed how a BER estimation can be made for a given situation. It would be interesting to know, whether we can predict what will happen with the quality when one of the parameters of the receiver is changed.

Using this predictions, the RAKE receiver can be adapted to the current environment by a control system at run-time.

In this section we explain how to make such a prediction for two important parameters of a RAKE receiver: the spreading factor and the number of fingers. The spreading factor has a substantial influence on the quality of the output of the RAKE receiver, as well as the costs. For a control system, it would be useful to be able to estimate the effect of doubling or halving the spreading factor on the BER (resp. $(BER_{SFdouble})$ and (BER_{SFhalf})). Similarly, it would be useful for the control system to be able to estimate the effect of changing the number of fingers. The number of used fingers of the RAKE receiver can be changed quickly by an easy local change on the receiver. When the quality of the output of the RAKE receiver is too bad, we would like to estimate whether it is possible to achieve the desired quality by adding fingers. Furthermore, when the quality of output of the RAKE receiver is sufficient, we would like to know whether it is possible to decrease the number of fingers to save on computation costs while still providing an accurate QoS.

A. Spreading Factor

The BER_{SFhalf} can be computed easily from the current BER distribution. When the spreading factor is halved, the mean (that represents the average soft value) of the distribution is halved as well. The central limit theorem states that the formula for the standard deviation of the mean is [13]: $\sigma_M = \sigma/\sqrt{n}$, where σ is the standard deviation of the original distribution and n is the number of samples. The standard deviation of the BER_{SFhalf} distribution is equal to the standard deviation of the current BER distribution multiplied by a factor $\sqrt{2}$ because the number of samples is halved. The increase of the standard deviation is caused by calculating the soft values using only half of the amount of chips. Therefore, the reliability is less, and the standard deviation is higher.

The $BER_{SFdouble}$ can be found using the same method. When the spreading factor is doubled, the mean is doubled and the standard deviation should be divided by a factor $\sqrt{2}$.

Table I summarizes the consequences for μ and σ resulting from a change of the spreading factor. The estimated BER of the new situation can be calculated using the new derived μ and σ from Table I.

B. Number of RAKE Fingers

The soft output is a combination of the correlation of the n fingers of the RAKE receiver, which we assume to have a normal distribution. When we also assume that:

double SF	$\mu_{SF_double} = \mu_{current} * 2$
	$\sigma_{SF_double} = \sigma_{current} / \sqrt{2}$
half SF	$\mu_{SF_half} = \mu_{current} / 2$
	$\sigma_{SF_double} = \sigma_{current} * \sqrt{2}$

TABLE I
CONSEQUENCES OF CHANGING THE SPREADING FACTOR ON μ
AND σ

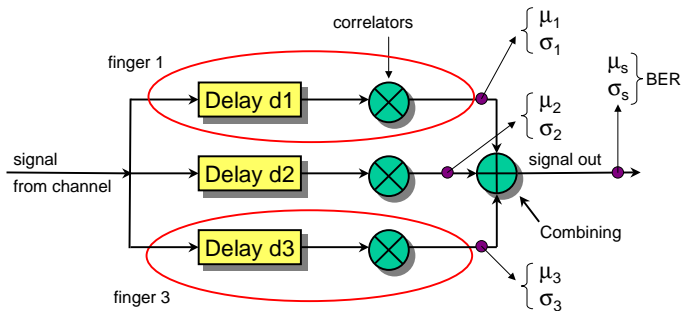


Fig. 12. Estimation of μ and σ per Finger of the RAKE Receiver

- the output of an individual finger has a normal distribution,
- the distributions of the different fingers are independent of each other,
- the RAKE receiver uses equal ratio combining

then the following relation exists:

$$\mu_s = w_1\mu_1 + \dots + w_n\mu_n \quad (20)$$

$$\sigma_s = \sqrt{(w_1\sigma_1)^2 + \dots + (w_n\sigma_n)^2} \quad (21)$$

where n is the number of RAKE fingers, w_i is the weight of a finger for the combining and μ_i and σ_i are respectively the mean and the standard deviation of the output of finger i ($i = 1, \dots, n$) and μ_s and σ_s are respectively the mean and the standard deviation of the output of the RAKE receiver as illustrated in Figure 12. The values μ_i and σ_i for each finger can be calculated with the Formulas (14) and (15) based on the output of the involved finger in a similar way as the computation of μ_s and σ_s for the soft output values.

If the RAKE receiver uses equal ratio combining (ERC) with an equal weight for each finger in the combination as shown in Figure 12, the weights w_1, \dots, w_n are equal to one. If the RAKE receiver uses maximum ratio combining (MRC) instead of equal ratio combining (ERC), then — of course — the weights for each individual finger are known and can be used for the weights w_1, \dots, w_n .

time:	t_0	t_1	t_2	t_3	t_4	t_5	t_6	t_7	t_8	t_9	t_{10}	t_{11}	t_{12}	t_{13}	t_{14}	t_{15}
path1:	1 ₁	1 ₂	1 ₃	1 ₄	1 ₅	1 ₆	1 ₇	1 ₈	2 ₁	2 ₂	2 ₃	2 ₄	2 ₅	2 ₆	2 ₇	2 ₈
path2:	1 ₃	1 ₄	1 ₅	1 ₆	1 ₇	1 ₈	2 ₁	2 ₂	2 ₃	2 ₄	2 ₅	2 ₆	2 ₇	2 ₈	3 ₁	3 ₂
path3:	1 ₆	1 ₇	1 ₈	2 ₁	2 ₂	2 ₃	2 ₄	2 ₅	2 ₆	2 ₇	2 ₈	3 ₁	3 ₂	3 ₃	3 ₄	3 ₅
path4:	2 ₁	2 ₂	2 ₃	2 ₄	2 ₅	2 ₆	2 ₇	2 ₈	3 ₁	3 ₂	3 ₃	3 ₄	3 ₅	3 ₆	3 ₇	3 ₈
path5:	2 ₅	2 ₆	2 ₇	2 ₈	3 ₁	3 ₂	3 ₃	3 ₄	3 ₅	3 ₆	3 ₇	3 ₈	4 ₁	4 ₂	4 ₃	4 ₄

TABLE II

FIVE PATHS WITH DIFFERENT DELAYS

Unfortunately, the output of the fingers of the RAKE receiver are not independent of each other. A substantial dependency comes from the shift in time between the paths, which causes non-orthogonality between the used spreading codes. Table II shows this phenomenon by giving the received chips using a spreading factor 8 and a delay of 2, 5, 8 and 12 chips for respectively the second, third, fourth and fifth path. When the delays do not change, the spreading codes are correlated continuously on the same position. For example, the delay between the first and the second path is two. So, when the spreading code of path two is shifted two positions to the right and is not orthogonal with regard to the unshifted spreading code of path one, then these two paths are not independent of each other.

Simulations show that the dependencies between the paths depends on the actual delay between the paths (using the same spreading code for the different simulations). Fortunately, in reality, the delays of the paths changes constantly, so the effect is limited. However, the effect is not negligible. To make a rough compensation, we correct the estimation with the current deviation. For example, when the RAKE receiver operates with four fingers and we would like to know the quality for three fingers we use the following procedure. First, a BER estimation is made based on an estimation of the μ and σ of the individual fingers and combining them using Formulas (20) and (21). Next, a BER estimation is made based on the soft output values of the RAKE receiver and a ratio is computed of the two estimated BERs. Finally, a BER estimation is made using the three individual fingers, compensated with the computed ratio.

VII. CONCLUSIONS

A method is introduced that makes an estimation of the bit error rate based on statistical analysis of the output of the RAKE receiver soft output only, without

prior knowledge about the channel model and all external influences. Simulations show that the mean error of the estimation is below 2%, having only a small variation. Implementation issues for use of the method in practice are discussed.

Acknowledgements

This research is conducted within the Chameleon project (TES.5004) supported by the PROGRAM for Research on Embedded Systems & Software (PROGRESS) of the Dutch organization for Scientific Research NWO, the Dutch Ministry of Economic Affairs and the technology foundation STW. We would like to thank dr. W.C.M. Kallenberg for his mathematical support and André Kokkeler for simulations performed at Ericsson Eurolab in Enschede.

REFERENCES

- [1] Price, R., Green, P.: A communication technique for multipath channels. In: Proceedings of the IRE. Volume 46. (1958) 555–570
- [2] Milstein, L.B.: Wideband code division multiple access. IEEE Journal on Selected Areas in Communications **18** (2000) 1344–1353
- [3] Smit, L.T., Smit, G.J.M., Havinga, P.J.M., Hurink, J.L., Broersma, H.J.: Influences of rake receiver/turbo decoder parameters on energy consumption and quality. In: Proc. of 2002 International Conference On Third Generation Wireless and Beyond. (2002) 227–235
- [4] Smit, L.T., Smit, G.J.M., Havinga, P.J., Hurink, J.L., Broersma, H.J.: Run-time control for software defined radio. In: proceedings PROGRESS 2002 workshop. (2002)
- [5] <http://www.3gpp.org>.
- [6] Morrow, R.K.: Accurate CDMA BER calculations with low computational complexity. IEEE Transactions on Communications (1998) 1413–1417
- [7] Cheng, J., Beaulieu, N.C.: Accurate DS-CDMA bit-error probability calculation in rayleigh fading. IEEE transactions on wireless communications **1** (2002) 3–15
- [8] Mann, P.S.: Introductory Statistics. 2 edn. John Wiley & Sons (1995) ISBN: 0-471-31009-3.
- [9] Tan, W., Chang, W.: Some comparisons of the method of moments and the maximum likelihood in estimating parameters of a mixture of two normal densities. Journal of the American Statistical Association **67** (1972) 702–708
- [10] Dudewicz, E.J., Mishra, S.N.: Modern Mathematical Statistics. John Wiley & Sons, Inc. (1988) ISSN: 0271-6232.
- [11] Cramér, H.: Mathematical Methods Of Statistics. 10 edn. Princeton University Press (1963)
- [12] Becker, J., Pionteck, T., Glesner, M.: Simulation, prototyping and reconfigurable hardware realization of CDMA RAKE-Receiver algorithms for flexible mobile transceivers. In: Proc. of ERSA'01. (2001) 119–126
- [13] Hald, A.: Statistical Theory with Engineering Applications. John Wiley & Sons, Inc. (1952)