Energy-Efficient Wireless Communication for Mobile Multimedia Terminals

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 presents a control system that adapts a WCDMA receiver at run-time to minimize the energy consumption while providing an adequate Quality of Service (QoS). The adaptation is done at run-time, because of the dynamic environment of a mobile receiver. Simulations show that run-time adaptation to the environment decreases the energy consumption of a receiver and also improves other QoS parameters, such as a higher throughput and a lower frame error rate.

Keywords *energy-efficient, wireless communication*

1 Introduction

Due to Internet, the world is getting more and more connected. At the same time, the world is getting less and less wired. People are becoming more mobile, carrying mobile devices such as laptops and mobile phones around. The combination of both trends results in a huge demand for wireless communication.

When we observe current mobile devices such as laptops, mobile phones, PDAs, and other kinds of mobile equipment people are carrying around, some painful shortcomings become clear.

First, a mobile device is by definition too big and heavy. The source of the problem is mainly the battery. About 60% of the size and the weight of a mobile device is due to the battery.

This brings us to the second problem: the battery is exhausted too quickly. A laptop can only be used for a few hours without a connection to the main outlet. A bigger battery can be employed, but this is counter productive to the first problem. Although the capacity of the battery is increasing every year (more energy for same weight/size), the improvement is marginal (about 10% per year) compared to the additional energy requirements for new functionalities. Therefore, things become worse in the coming years.

The third problem is that the environment of a mobile device may change dramatically in the short term as well as in the long term. For example, when a mobile caller rounds the corner, the quality of the connection may become considerably different.

The fourth problem is more an indirect problem for the consumer. More and more standards for wireless communication are coming up, new standards become increasingly complex, have more options and require more tuning. Therefore, the lifetime of devices becomes shorter and the time to market has to be faster. These conflicting issues makes the life of a manufacturer very difficult. He has to produce more complex devices and services with more features in a shorter time. And last but not least, it should also be cheaper! Ultimately, at the end of the chain, the consumer will pay for the disasters in one way or other.

To summarize, there is a need for energy efficient, flexible architectures that provide an adequate quality of service. In this work, we will not focus on improving the capacity of batteries and we do not search for 'more' energy capacity, but we search for ways to use the existing energy budget in a more efficient way. Or stated better, in a more effective way, i.e. do more with the same amount of energy. The Chameleon project at the University of Twente addresses these problems. The approach of the Chameleon project is to use reconfiguration at different levels of the architecture to adapt — like a Chameleon — the mobile device to the current situation. So, given the QoS requirement of the user and the current environment we adapt *the entire system* — from hardware to application software — to fit exactly the user's needs.

Although the most common focus of low-power design is the development of low-power hardware, having low-power hardware only does not solve the problem. Also the other parts of the architecture of a mobile should be energy aware. This article focuses on energy efficiency in the protocol stack used for wireless communications by adapting the settings of the receiver to the external environment.

2 Wireless Communication

Wireless communication has two quite different characteristics compared to communication via fixed wires. First, the scarce energy resources of a battery-powered mobile require an energy-efficient operation. Second, the dynamic environment of a mobile causes large quality variations of the transport medium and requires a continuously monitoring of the Quality of Service (QoS) of the wireless link. A wireless link is far less stable than a fixed link, because the environment of a wireless link changes over time. For example, a user with a mobile terminal can walk around the corner and the transmission path between mobile and base station may become much worse or just better. This means that the environment may be very dynamic within a short time. A lot of different factors influence the quality of the wireless link. Some factors are technology dependent or implementation dependent, while other factors are determined by the environment. For example, factors that influence the quality of a Wideband Code Division Multiple Access (WCDMA) link are (among others):

- signal-to-noise ratio
- spreading factor
- oversampling rate at receiver
- modulation type
- number of paths (reflections) with per path: phase shift, amplitude, and delay
- chip rate
- speed and direction (Doppler effect)
- number of fingers of the receiver
- interference between:
  - symbols of different paths of the same user
  - symbols of other simultaneously transmitting users
  - symbols of different channels
- accuracy of: channel estimation and tracking
- quantization

This list is incomplete and the factors are not independent of each other. What we want to make clear is that a lot of different parameters influence the perceived quality of a wireless link and that it is not trivial to determine the quality of a wireless link. A lot of different channel models exist, but they might deviate significantly from reality at a certain moment in time. The main problem is that on beforehand it is unknown what the influence of all effects is on the channel. Beside this, also the dependencies between the effects makes it difficult to model the channel behavior correctly.

The QoS of a wireless link may be expressed in different ways, e.g.:

- the perceived quality, e.g.: resolution of an image, frame loss of video, delay of voice,
- the cost, e.g.: energy cost, cost in money,
- the security.

Most definitions of QoS consider the perceived quality only. However, there is a direct relation between the perceived quality and for example the costs. A mobile that runs twice as long with the same perceived quality has a better QoS. Especially for a mobile, the costs are not negligible. Therefore, we will consider the costs as an essential part of the total QoS.

3 Application

To demonstrate our approach, we will we consider a RAKE receiver in combination with a turbo decoder that is used in a mobile UMTS terminal. We consider only the reception of down-link traffic (transmission from base station to the mobile) because energy-efficiency is more important for the mobile than for the base station. We assume that the reader is familiar with UMTS receiver principles.

3.1 Rake Receiver

A RAKE receiver (see Figure 1) is used for Wideband Code Division Multiple Access systems (WCDMA) [4]. In a WCDMA system all the users transmit in the same band simultaneously. Each transmitted bit is spread by the transmitter by means of a multiplication with a pseudo random code. The length of this code is called the spreading factor. A larger spreading factor gives a better resistance against interference (interference of multiple users, multiple connections, multiple paths). The receiver de-spreads the received signal by a correlation process.

A RAKE receiver has multiple fingers to correlate the received signals from different paths with different delays, and combines the results of the different paths to construct one output signal. For a more detailed description of a RAKE receiver, see [4, 7].

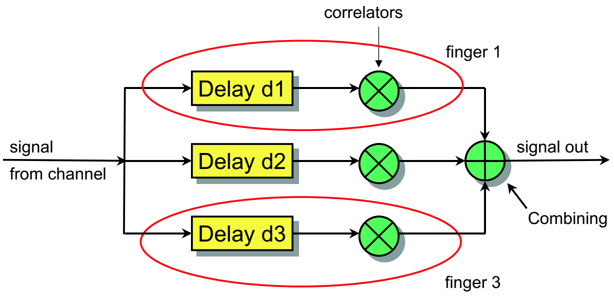

Figure 1: Basic Principle of a Rake Receiver

3.2 Turbo Decoder

A turbo decoder is a forward error correcting decoder. It uses extra redundant information generated by a turbo encoder. It uses the soft values of a RAKE receiver as input and produces hard bits (0 or 1) on the output. The turbo decoding algorithm can run multiple times on the same input data to improve the output of a turbo decoder. This iterative principle gives a turbo decoder a better performance than a conventional decoder. More details about turbo decoding can be found in e.g. [8].

4 Approach

Our main goal is to operate as cost-effective as possible while still providing an adequate QoS. We distinguish between effective and efficient. Effective means to minimize the amount of effort to perform a certain task. So, this means a functional optimization. When the control system has determined what the minimum amount of effort is, then this has to be performed as efficient as possible. This second step is implementation and architecture dependent. We will focus on the first optimization step, how to operate as effective as possible. The most effective way of operation depends on the current environment. So, when the environment changes, another way of operation may become more effective. Therefore, this optimization has to be performed at run-time. In our approach, the status of the wireless link is monitored continuously and the receiver is adapted to operate in the most effective way by a control system as shown in Figure 2. This control system has to be simple, because the time to make a decision is limited (due to the changing environment) and the computational costs have to be low. The effort to run the control system should be lower than the achieved saving. Hence, computational intensive optimization algorithms, such as generic algorithms or simulated annealing are not usable.

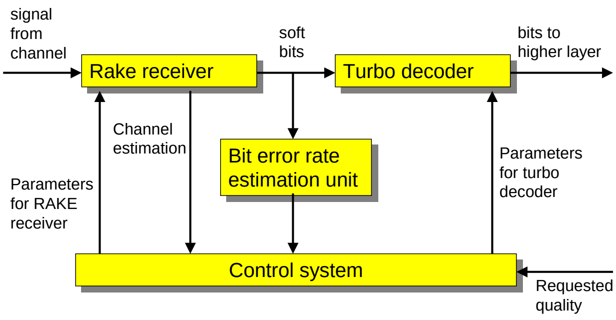

Figure 2: The Control System for a Wireless Receiver

Summarizing, we propose a control system, which adapts a WCDMA receiver to the dynamically changing external environment at run-time. The goal is to operate with minimum effort, while still delivering a satisfying QoS level that has been specified by the user (or its application).

4.1 Perceived Quality

In our approach the end-user specifies a certain QoS. This may be specified in different dimensions, e.g. the desired throughput (bits/s), the maximum delay (e.g. are re-transmits acceptable?) and the reliability (does the application of the end-user tolerate that received frames might still contain errors?). In general, a frame has to be received error free, except for some specific data types such as used for video or voice. So, even if a frame contains only one bit error, it is useless and has to be re-transmitted if this is useful anyhow. For streaming data (e.g. video) the re-transmitted frame may arrive too late to be useful. In this case, it is better to save the effort of re-transmission. We use at application level the frame error rate (FER) as quality measure, which may be considered as a good measure.

4.2 FEC Decoder Characteristics

A forward error correction (FEC) decoder processes usually on a block of bits. An important question is what the error correction capacity of a FEC decoder is, i.e. how many bits in a block may be received incorrectly such that the FEC decoder is still able to decode this block correctly? Unfortunately, from theory no nice functions are known that give the error correction capacity of a turbo decoder. To estimate

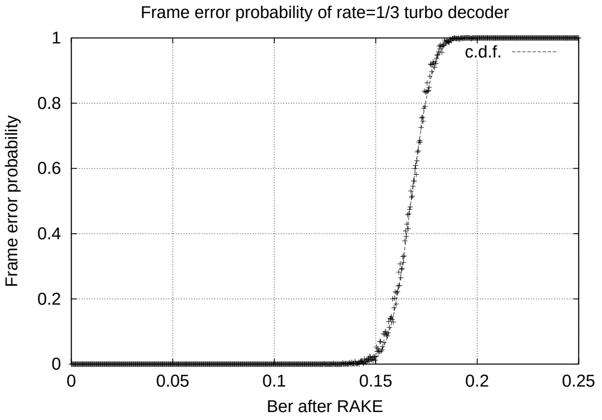

Figure 3: Frame Error Probability of Turbo Decoder with rate=1/3 - 646464 blocks

this capacity, we simulated the transmission of blocks under many different circumstances and investigated how often the FEC decoder was able to decode a block correctly as a function of the bit error rate (BER) of the input block. Figure 3 shows the result for a specific turbo decoder. So, if the incoming block has a BER of 10%, the turbo decoder is (almost) always able to decode the block correctly, whereas for a BER of 20%, the turbo decoder can almost never decode the block correctly. For a BER of around 16.5%, the turbo decoder can decode the block correctly in about 50% of the cases. Similar graphs can be obtained for a Viterbi decoder and FECs with different parameter settings.

It is important to notice that irrespective whether the BER of the input block is 1% or 10%, in both cases the turbo decoder is able to decode the block correctly. Therefore, it is useful to tune the RAKE receiver in such a way that the output of the RAKE receiver is just good enough, such that the turbo decoder is able to correct these blocks correctly. Spending more effort in the RAKE receiver to produce better output is a waste of effort.

4.3 BER Estimation

To use the results of the previous subsection, we need to estimate the BER of the output of the RAKE receiver so that we can predict the probability that the FEC decoder can correct a frame.

A common method is to estimate the BER with pilot symbols. Pilot symbols are a predefined sequence of symbols that are known at both transmitter and receiver side. Therefore, the receiver knows which sequence is transmitted and can determine the BER.

Another approach is to model the channel with all the known effects (see e.g. [3]). A state of the art article on this area is [2]. 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. In practice, it is not possible to model all the different effects that cause the distortion of the wireless channel. Therefore using this method, an exact estimation of the quality of the signal of the wireless channel is impossible.

In [6] we propose another method to obtain an estimation of the BER of a wireless link, which uses BPSK/QPSK modulation direct after the RAKE receiver, before a forward error correction (FEC) decoder. The method is based on statistical analysis of the received symbols before de-modulation. Our method to estimate the BER differs significantly from the mentioned approaches. We only use the soft output from the receiver, and require no additional information about the channel. In our opinion, it does not 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. The estimation error of our method for BER estimation is below 2% [6]. Furthermore, the proposed method uses no pilot symbols, avoiding the transmission of pilot symbols for the purpose of BER estimation. Instead it uses all received data for an estimation of the BER. In addition to that, our method provides the possibility to estimate the quality resulting from a planned adaptation of a parameter. E.g. in [6] we show that the BER is equal to $\Phi(-\mu/\sigma)$ where Φ is the cumulative distribution function of the standard normal distribution and μ and σ are calculated from the received soft values (symbols). When we plan to double the spreading factor, we know that the μ will double too and that σ will become $\sigma/\sqrt{2}$. Similarly, when we plan to halve the spreading factor, we know that μ will halve too and σ will become $\sigma * \sqrt{2}$. Using this, we can make an accurate estimation for the BER after the planned change.

The proposed method to make an estimation of the BER for a wireless channel using a BPSK/QPSK modulation scheme, can be easily extended to other modulation schemes, such as 16-QAM or 64-QAM.

5 Control System

To test our approach, we built a control system that adapts the spreading factor and the number of fingers of the RAKE receiver. Although we control only two parameters, the same approach can be used for controlling other parameters. Both parameters have a substantial influence on both the quality and the energy consumption. Doubling the spreading factor will about double the energy consumption of the RAKE receiver. Also, adding more fingers causes an about linear increase in the energy consumption. The effects of such changes on the QoS can be predicted with the results from Section 4 and [6].

The control system adapts the spreading factor for the long-term and the number of fingers for the short-term. Changing the spreading factor requires negotiation with the base station, which introduces delay and additional costs. The number of fingers of a RAKE receiver can be adapted quickly within the receiver and can be used for the fine-tuning. If reconfigurable hardware is superfluous after adapting the system, it can be switched off to save energy or can be reconfigured to perform another function (e.g. to perform turbo decoding). Next, we outline how the control system adapts the parameters.

When the BER drops below a certain BER threshold, the control system directly takes actions by adding fingers or increasing the spreading factor because it expects that the turbo decoder is no longer able to decode the received frames correctly. Because the expected quality as a result of the adaptation is only an estimation, it maintains a certain margin to be sure that the QoS after adaptation is good enough. In a second phase, fine-tuning can be done to decrease the quality to minimize the the energy consumption for the new situation. The control system will first evaluate whether maximization of the number of fingers is sufficient to get the required quality. Adaptation of the number of fingers is preferred above adaptation of the spreading factor, because this can be done quickly and cheap. Only when this does not raise the QoS to the desired level, the spreading factor is adapted.

When the QoS is adequate, there may be another mode of operation (i.e. another set of the parameters) that gives a lower QoS that still satisfies the requested QoS of the user while the amount of effort to deliver this QoS is lower. In this situation, it is maybe possible to save the energy consumption by lowering the QoS. To recognize such a situation, the control system makes a prediction of the expected quality after lowering the spreading factor or decreasing the number of fingers. Also increasing the number of fingers to improve the quality so that the spreading factor may be decreased may lead to a decrease of effort. The spreading factor is only adapted when the expected QoS is considerable better then the desired QoS threshold given by the user. Otherwise, it might be necessary to reduce the spreading factor again when the quality of the wireless link changes only slightly. This avoids oscillations of the control system with regard to the spreading factor.

When the quality of the wireless channel has not been changed noticeably, it is not useful to adapt parameters. Therefore, an optimization is only investigated when the control system detects that the quality of the wireless channel has changed noticeably. This implies that the overhead is low for a stable situation. In a dynamic situation, the overhead becomes slightly higher due to computations that are necessary to estimate the quality, but this is compensated by the potential savings after adaptation of the parameters.

A simple evaluation of the number of executed operations shows that the total overhead is below 1% of the total computation costs that is necessary in the RAKE receiver.

6 Results

We implemented the proposed control system in our simulation environment. Figure 4 shows an example scenario of a dynamic environment and the adaptation of the receiver by the control system. After 3.84 Mchip (which is equivalent to 1 second in UMTS) one of the settings of the external environment is changed. Although the changes in the environment are relatively fast, the results do not change substantial when the period between the changes is longer. Settings that are changed in the external environment are the number of simultaneously transmitting users, the number of received paths (reflections) and the signal-to-noise ratio.

The graphs in Figure 4 show respectively: (1) the number of users, (2) the signal to noise ratio, (3) the number of paths, (4) the BER after the RAKE receiver (but before the turbo decoder), (5) an indication whether the frame could be decoded correctly or not by the turbo decoder, (6) the costs expressed in number of operations required to process one bit, (7) the spreading factor of the RAKE receiver and (8) the number of fingers of the RAKE receiver. The hori-

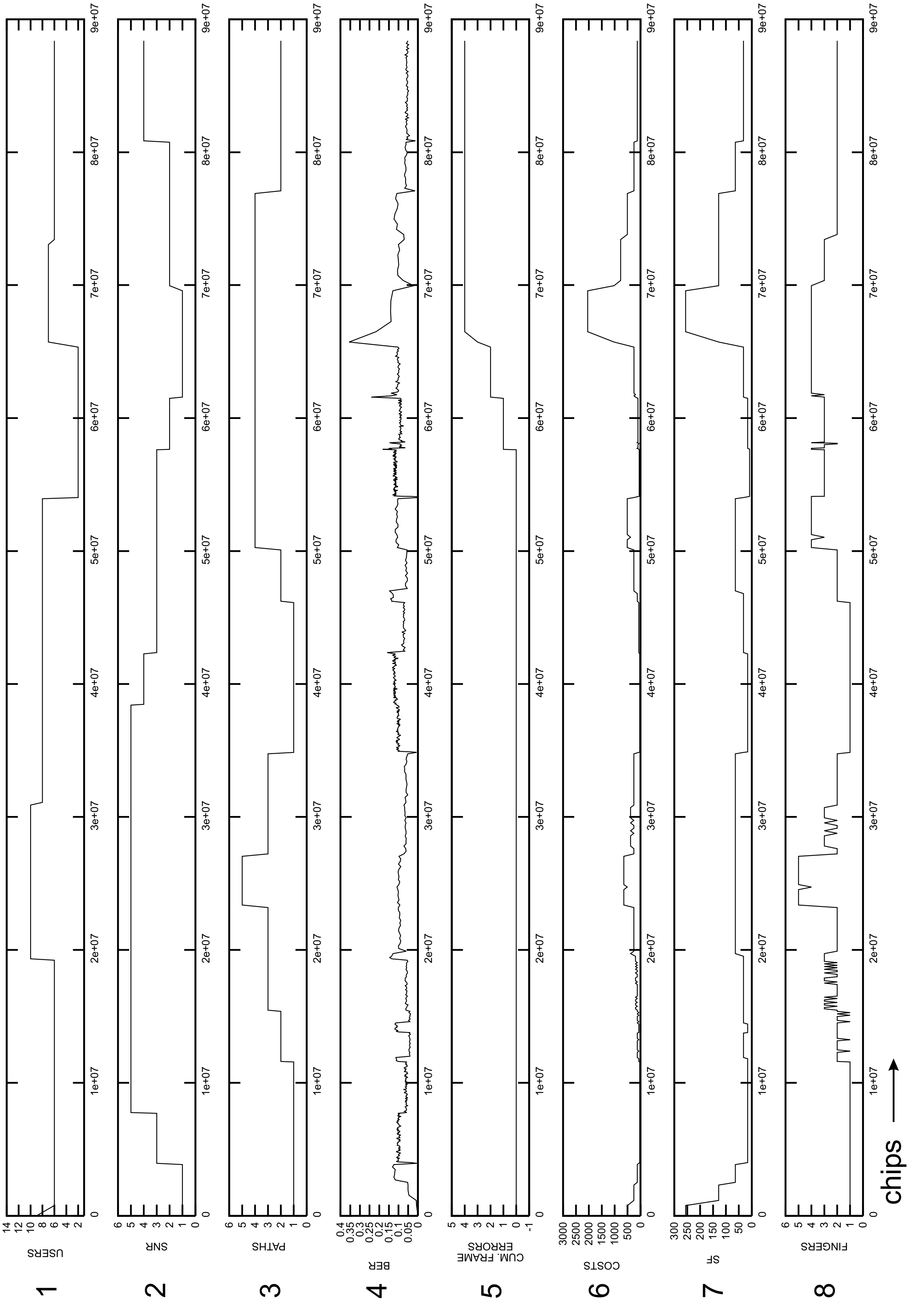

Figure 4: Sample Scenario 1

	configuration	throughput (Kbits/s)	FER (%)	costs (op./bit)	total frames	total costs (Gops)	delay (s)
s	control	42	0.4%	511	5020	2.6	120
c	sf=256, f=3	5	1.7%	4312	5086	21.9	1017
e	sf=128, f=3	10	4.3%	2223	5225	11.6	523
n	sf=64, f=3	18	8.6%	1166	5470	6.3	304
a	sf=32, f=3	29	26%	718	6757	4.9	233
r	sf=16, f=3	38	52%	556	10417	5.8	274
i	sf=8, f=3	15	90%	1315	50000	65.8	3333
o	sf=256, f=1	5	4.2%	2414	5219	12.6	1044
	sf=128, f=1	8	21%	1470	6329	9.3	791
1	sf=64, f=1	14	29.5%	822	7092	5.8	507
	sf=32, f=1	19	52.4%	607	10504	6.4	553
	sf=16, f=1	22	72%	517	17857	9.2	812
	sf=8, f=1	7	95%	1535	100000	154	14286
s	control	11	2.1%	2172	5107	11.1	464
c	sf=256, f=3	4	11%	4615	5618	25.9	1404
e	sf=128, f=3	7	33%	3085	7463	23.0	1066
n	sf=64, f=3	9	55%	2319	11111	25.8	1235
.	sf=32, f=3	6	86%	3651	35714	130	5952
	sf=256, f=1	9	42%	4007	8612	34.5	958
	sf=128, f=1	5	50%	2328	10000	23.2	2000
2	sf=32, f=1	5	86%	2110	35714	75.4	7143

Table 1: Two Example Scenarios with Different Settings of RAKE Receiver

zontal axis shows always the number of transmitted chips (so linear with the time).

The first graphs (1)–(3) show the status of the external environment, graphs (4)–(6) show the consequences on quality and costs and the graphs (7) and (8) show the resulting settings of the parameters of the RAKE receiver. Graph 7 shows that the control system does not oscillate with regard to the spreading factor. Once a decision is made by the control system, the spreading factor is not adapted again before the external environment has been changed considerably. Graph 8 shows sometimes some oscillations with regard to the number of fingers, but this is no problem because the number of fingers can be changed local on the RAKE receiver, which does not introduces additional costs.

Although it gives some insight in the working of the control system, the end-user is in the total costs and the total QoS. Therefore, we summarize the results that are relevant to the end-user in Table 1. The first row of Table 1 represents the QoS results of Figure 4 (with the adapted receiver) for the end user. The next rows show the result for the same scenario for the external environment, while the RAKE receiver was not adapted. In such a static case, the spreading factor and the maximum number of fingers are fixed during the whole scenario. Different rows show different combinations of spreading factors and number of fingers for the same scenario. The second part of Table 1 shows an additional example for another scenario with more simultaneously transmitting users, i.e. a scenario where the quality of the wireless link is worse. The first column of Table 1 denotes the used scenario for the changes in the external environment. The second column denotes the used configuration. The third column denotes the throughput in Kbit/s. The throughput denotes how many Kbits per second are received correctly. Note that when the FER is above zero, the actual number of bits that were transmitted is higher. Furthermore, the throughput of the coded bits is about three times higher because the transmitted bits are turbo encoded with rate 1/3. The fourth column denotes the percentage of frames that are received incorrectly. This means that the turbo decoder is not able to correct all bits in the frame. For most applications, an incorrect frame has to be re-transmitted. This introduces additional costs and it introduces also additional delay. These additional costs are not included in Table 1. When the frame error rate increases, the probability that more re-transmits are necessary to receive a certain frame increases a lot. Therefore, a setting with a high throughput but a high frame error rate may be unacceptable. The fifth column denotes the costs required to execute the algorithms of the receiver, expressed in number of operations per bit.

The last three columns of Table 1 describes the costs and the delay for each configuration for a case where 5000 frames of 1000 bits are transmitted (e.g. for the transmission of a mp3 file). Column six denotes the number of frames that has to be send with re-transmission of incorrectly received frames included. Column seven denotes the costs in number of operations that in total is needed to receive the number of frames denoted in column six. Additional costs for re-transmits (such as costs to request for re-transmits) are not included. The last column denotes the total delay for the reception of the 5000 frames calculated In reality the time is higher because it also takes time to request for a re-transmit. However, a large part of this delay may be hidden by windowing. Table 1 shows different results:

- Scenario 1

  The low frame error rate for the adapted receiver shows that the receiver adapts well to the quality of the channel. The low delay compared to the other configurations with static settings (column eight) is also the result of an adequate adaptation of the receiver. Also the throughput as well as the costs of the adapted receiver outperforms the other configurations with static settings.

  The configurations with a static setting with a spreading factor of 16 have a good cost/bit ratio but the throughput is lower and the frame error rate (FER) is much higher. However, when we consider the total costs (column 7), a higher spreading factor is better, e.g. the configuration sf=32 and 3 fingers. Especially the higher FER is a serious problem. First, it introduces extra costs due to re-transmits and additional delay. Furthermore, the reception of data is bursty. When the channel becomes too bad to receive the frames correctly, the reception of correct frames might stall for a long time.

  A low FER for scenario 1 is also achievable with a static configuration with a high spreading factor (e.g. sf=256). However, the costs/bit increases immense because this spreading factor is also used in case of a relative good channel state. Also column seven (total costs) shows the advance of the adapted configuration. Due to the same cause, the throughput is limited.

- Scenario 2

  Scenario 2 is a scenario with even more simultaneously transmitting users resulting in a worse channel compared to scenario 1. Also in this scenario, the FER for the adapted receiver is much lower compared to the static settings. Although the static situation with spreading factor 32 and 1 finger just outperforms the adapted receiver in terms of cost/bit, the FER is unacceptable high.

7 Conclusions

This paper presents a control system that adapts the settings of a WCDMA receiver at run-time to the environment to achieve energy-efficient communication while guaranteeing an adequate Quality of Service for the end-user. Through adaptation to the environment, the required QoS is delivered with minimum effort. The key issue in our approach is cross layer optimization, instead of optimization of each individual OSI layer. With this global optimization, substantial gains can be achieved. The gain is largely dependent on the dynamics of the external environment. Two different scenarios for a wireless channel are presented to compare the adapted receiver with receivers with static settings for the spreading factor and the number of fingers. The adapted configuration maintains a remarkable low frame error rate compared to the other static configurations. This shows that the adapted receiver adapts very well to the external environment. Furthermore, the throughput, the delay as well as the costs of the adapted receiver outperforms the configurations with static settings. The presented control system has little overhead (<1%) and includes margins to prevent against oscillations.

8 Related Work

8.1 Wireless Image Transmission System

In [1] a total system energy minimization for wireless image transmission is discussed. The system uses a "set partitioning in hierarchical trees" (SPIHT) encoder [5] to compress an image. Next, a Reed

Solomon encoder adds redundancy for forward error correction. The images are transmitted over an IMT-2000 channel, which is based on CDMA technology. The system optimizes five parameters to achieve minimal energy consumption within the given quality of service constraint. The peak signal noise ratio (PSNR) of the image is used as a metric for the quality of service. The PSNR is the mean of the summed squared difference between the pixels of the original and the resulting image. The five adaptable parameters are:

- On the transmitter side:
  - rate of the encoder, which expresses the number of bits per pixel.
  - Reed Solomon encoder: maximum number of correctable symbols per block.
  - power of the amplifier of the transmitter
- On the receiver side:
  - number of RAKE fingers of the receiver
  - Reed Solomon decoder: maximum number of correctable symbols per block.

Note that the parameters for the Reed-Solomon should be the same for the encoder and the decoder. Therefore, the configuration space limits to four parameters.

A real-delay gate-level simulator is used to obtain the absolute energy costs for the different components. The performance is expressed as a function of the signal to noise ratio. Due to the large number of possibilities, the optimization algorithm that selects the optimal set of parameters updates with one possibility at a time instead of a global optimization.

At first glance, our approach looks very similar to the work in [1], but it differs significantly. We mention a number of striking differences.

First, the method from [1] assumes that the performance of the channel is known. However, in reality the performance of a wireless channel is unknown. In our work, one of the significant contributions is that we can measure the performance of an unknown channel.

Second, a slow fading channel is assumed. However, in reality the channel may also be fast fading. The system should be able to adapt to a fast fading channel as well.

Third, one of the most important parameters for a CDMA based system, the spreading factor is not taken into account. The spreading factor has a major impact on quality as well as costs. In our system, this parameter is studied thoroughly.

Fourth, the system presented in [1] focuses on image transmission. More specifically, it assumes that transmitted block may contain errors. This assumption is only valid for a very limited amount of applications. In our approach, we also support different kind of data.

Fifth, the system presented in [1] does not support re-transmission of frames. However, it may be much better in terms of energy consumption to lower the performance of the receiver and to re-transmit some packets every now and then when necessary. Our system supports this trade-off.

Sixth, in [1] it is assumes that the transmitter can optimize the transmit power for the point to point link. However, in a CDMA system power control should be optimized global over the cell. The reason for this is that if one of the users changes the transmit power, this results also in a different channel quality for the other users. Therefore, we do not involve the transmit power into our control system.

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.

References

[1] S. Appadwedula, M. Goel, N. R. Shanbhag, D. L. Jones, and K. Ramchandran. Total system energy minimization for wireless image transmission. *Journal of VLSI Signal Processing*, 27:99–117, 2001.

[2] J. Cheng and N. C. Beaulieu. Accurate DS-CDMA bit-error probability calculation in Rayleigh fading. *IEEE transactions on wireless communications*, 1(1):3–15, 2002.

[3] R. K. Morrow. Accurate CDMA BER calculations with low computational complexity. *IEEE Transactions on Communications*, pages 1413–1417, Nov. 1998.

[4] T. Ojanperä. *Wideband CDMA for Third Generation Mobile Communications.* The Artech House Universal Personal Communications Series. Artech House, 1998. ISBN: 0-89006-735-X.

[5] A. Said and W. A. Pearlsman. A new fast and efficient image codec based on set partitioning in

hierarchical trees. *IEEE Transactions on Circuit and Systems for Video Technology*, 6:243–250, June 1996.

[6] L. T. Smit, G. J. M. Smit, J. L. Hurink, and A. B. J. Kokkeler. Soft output bit error rate estimation for WCDMA. In *Proceedings of Personal Wireless Conference 2003*, pages 115–124, Sept. 2003. ISBN: 3-540-20123-8; ISSN: 0302-9743.

[7] G. L. Turin. Introduction to spread-spectrum antimultipath techniques and their application to urban digital radio. *Proc. of the IEEE*, 68(3):328–353, Mar. 1980.

[8] M. C. Valenti. *Iterative Detection and Decoding for Wireless Communications.* PhD thesis, Virginia Polytechnic Institute and State University, July 1999.

Lodewijk Smit received the M.S. degree and the Ph.D. degree in computer science from the University of Twente (UT), The Netherland, in 1996 and 2004, respectively.
He worked for three years at Akzo Nobel in Arnhem, The Netherlands. Since 2000 he works in the Chameleon project, which investigates new hardware and software architectures for battery-powered hand-held computers. Recently he joined the European Union project Smart Chips for Smart Surroundings that defines and develops efficient (ultra low-power), flexible, reconfigurable core building blocks, including the supporting tools, for future ambient systems. His current interest is in the scheduling of applications to heterogeneous multiprocessor architectures.

Gerard Smit received his M.S. degree in electrical engineering from the University of Twente and received his Ph.D. degree in 1994. He worked for four years at Océ R&D in Venlo the Netherlands. Since 1996 he works in the Chameleon project, which investigates new hardware and software architectures for battery-powered hand-held computers. Currently his interests are: low-power communication, wireless multimedia communication, and reconfigurable architectures for energy reduction. Dr. Smit has been and still is responsible for a number of research projects sponsored by the EC, industry and Dutch government in the field of multimedia and reconfigurable systems.

Johann Hurink received a Ph.D. degree at the University of Osnabrueck (Germany) in 1992 for a thesis on a scheduling problem occurring in the area of public transport. From 1992 until 1998 he has been an assistant professor at the same university working on local search methods and complex scheduling problems. This work resulted in a Habilitation thesis at the University of Osnabrueck in 1999. Since 1998 he is an assistant professor in the group 'Discrete Mathematics and Mathematical Programming' at the department of Mathematical Sciences at the University of Twente. Current work includes the application of optimization techniques and scheduling models to problems from logistics and telecommunication.