Hybrid type-II ARQ/AMS supported by Channel Predictive Scheduling in a Multi-User Scenario

Nilo Casimiro Ericsson and Anders Ahlén
Signals and Systems
Uppsala University, PO Box 528
SE-751 20 Uppsala, Sweden
{nce, Anders.Ahlen}@signal.uu.se

Sorour Falahati and Arne Svensson
Communication Systems Group
Department of Signals and Systems
Chalmers University of Technology
SE-412 96 Göteborg, Sweden
{sorour, Arne.Svensson}@s2.chalmers.se

Abstract

A combination of radio channel predictive time slot scheduling and link layer FEC/ARQ (Forward Error Correction / Automatic Repeat reQuest) with adaptive modulation, for IP (Internet Protocol) packet data, is presented and evaluated through simulations. The main idea is to make use of the fast fading characteristics of the radio channel through time slot scheduling, instead of alleviating the effects with over-pessimistic channel coding. By using information about the individual data streams, together with information about future wireless channel characteristics for the different mobile hosts, it is possible to plan the transmission to achieve a good quality of service while utilizing the scarce spectrum efficiently. The simulated results confirm that our proposed system can provide a high throughput while maintaining the BER and tthe delay low for the majority of the users.

1 Introduction

In future packet based wireless communication systems, the downlink will be excessively used for data transmission to mobile terminals. An obstacle in this context is the time-variability of the channel. To achieve a high system throughput also over fading channels, adaptive methods for the adjustment of the modulation alphabet and the coding complexity, together with time-slot scheduling, can be used [5–7, 10]. In our previous publications [3, 4], we describe two substantially different approaches to packet data transmission over fading channels, using adaptive methods for modulation and coding. In this paper, we combine the two approaches.

One approach is the Hybrid type-II ARQ scheme combined with an Adaptive Modulation System (AMS) [4]. The other approach is based on prediction of the channel quality, thereby allowing for scheduling of the transmission among multiple users [3].

The idea behind the proposed HARQ-II/AMS is to reduce the signaling constellation size and/or the coding rate at each (re)transmission attempt. Therefore, more immunity against channel impairments can be provided in the latter transmission attempts while still having the prospect of a high transmission rate in the first attempt. The drawback of this system is the delay imposed by the method: The adaptation of coding rate/modulation is only controlled by the ACK/NACK (Acknowledgement / Negative Acknowledgement) feedback signal which increases the system delay in poor channel conditions. The performance of this system was previously studied and evaluated only for the single user case, where no scheduling algorithm was implemented.

In the scheduling approach, prediction of different user channels provide a basis for detailed scheduling of the transmission, by combining time-slot allocation and adaptive modulation. This approach can also take into account the desired error-probability and the priority associated with different users, as well as the current traffic situation. Moreover, the frequency band can be used efficiently, since the different users are allocated time-slots when their transmission conditions are predicted to be favorable, allowing them to use a high modulation level. The resulting constant and low (user-specified) error-rate provides the error correcting codes with manageable data, avoiding bandwidth consuming re-transmissions. The main drawbacks are the computational complexity, and the sensitivity to channel prediction errors.

In this paper, the two methods are combined and the performance of the combined approach is evaluated and compared to the two individual methods for IP traffic over fading channels. The combined method successfully reduces the drawbacks of the individual methods described above: Knowledge of the channel conditions gives a hint on the initial coding rate/modulation to use, reducing the delay due to numerous NACKs, and, the ARQ-scheme provides robustness against channel prediction errors.

In Section 2 the problem addressed within this work is explained and discussed. The system is described thoroughly in Section 3. The simulation results are presented in Section 4. Finally some conclusions are drawn in Section 5.

2 Problem formulation

We address two problems: The *spectrum efficiency* of data transmission over wireless links, and the *performance* of IP over unreliable channels. The solution we propose gracefully combines the two problems into one: The problem of allocating time-slots on the wireless link to different data streams, based on the high level protocol requirements, and the actual physical channel constraints.

2.1 IP over Fading Channels

The problem of transmitting TCP/IP flows over wireless links has been addressed in several publications [1, 11, 12]. TCP was designed for high bandwidth, wired networks, where retransmission timeouts most probably are a cause of traffic congestion at some node along the path, not due to erroneous reception because of bad channel conditions. The action taken by TCP at the source, apart from retransmitting the lost packets, is to reduce the rate at which the packets are sent, to help alleviate the assumed congestion. Suggestions exist to introduce more explicit TCP signaling [12] (Explicit Congestion Notification) to accommodate lossy links into IP networks without degrading TCP performance.

Questions arise whether we should implement split connections (indirect TCP) [11], or allow the TCP to maintain an end-to-end connection. The split connection idea disrupts the TCP philosophy of end-to-end connections, but might be necessary to counteract the effects of a lossy wireless link. In a split connection, a proxy server at the base-station handles the TCP connection with the source host at one end, and the destination host at the other. In the latter case, end-to-end TCP, a snoop agent could be used to improve TCP performance [1]. The snoop agent, hosted by the base station, would have a shorter retransmission time-out, expecting ACKs to come quicker from the mobile host, and thus handling some of the retransmissions that would have otherwise been done by the original TCP source. The snoop agent also selectively passes ACKs from the mobile host to the source host. By doing this, the snoop agent also prevents the source from going into congestion mode, unless real congestion occurs in the wired network.

In our approach, we maintain a buffer with separate queues for the different IP flows, allowing us to extract some header information necessary for the scheduler to make a good decision on which queues to choose for future transmissions. Both alternatives (split connection, and snoop agent) then remain open for usage, so we do not need to make that choice.

2.2 Spectrum efficiency

To obtain high data throughput over wireless channels, we have to act clever. The channel quality varies substantially over time, due to radio interference and the mobility of the radio stations. Negative effects, such as different types of fading, impose a high probability for a bad radio connection. *Slow fading* can be counteracted by controlling radio transmitter power, or performing handover to another base station. The remedy against *fast fading*, however, is traditionally different types of channel coding and interleaving. In a simplified explanation, the channel coding adds extra information, or controlled redundancy, to the transmitted data, whereas the interleaver spreads the information over time, to make it more robust against error bursts which occur in the fading dips. The channel coding, which often is over-pessimistic, generates much overhead to the wireless system, which in turn wastes precious bandwidth. We have a substantially different approach.

It has been demonstrated in [2] that it is possible to predict the channel SINR variations quite accurately several milliseconds into the future. That particular method involves channel measurements and non-linear, model-based, time-series predictions. Having these predictions, one per radio link, they can be used together with a target error rate to assign initial coding and modulation rates (Section 3.3) for the HARQ-II/AMS, but also for planning of the transmissions to the different mobile hosts, giving access to the users that have good predicted channel quality. Doing this, we increase our chances of getting the data across the wireless link without error and at a high rate, thus increasing the system throughput, and the spectrum efficiency.

2.3 Spectrum efficient IP

So, by matching the higher protocol layer requirements with the physical constraints of the radio channel, we hope to find a time-slot allocation, along with a coding strategy, that efficiently utilizes the available spectrum for the running applications.

3 System description

Traffic originating from a variety of applications are likely to co-exist in future wireless networks. They all have their preferred services from the lower layers, ranging from end-to-end transmission control through TCP, to latency sensitive real-time services, provided by e.g. RTP (Real Time Protocol). We would like to meet as many of the requirements as possible from these applications. Therefore, we need to differentiate the incoming traffic, and serve the packets with the objective of maintaining a satisfying service level for as many users as possible.

We need some kind of quantity which can be specified by the application, to dictate requirements on the actual service. In our approach we have chosen to use three parameters, namely the maximum error rate, the minimum throughput and the maximum latency. We translate them to the following service specifications, which are easily implemented:

Target error rate gives a specification of the kind of error protection and channel quality that is needed to serve an application.

Packet stream priority is required when different packet streams have different requirements on latency. High priority packets are processed before lower priority ones.

Number of transmission attempts at the link layer reflects the latency requirements as well, but also the kind of transport protocol used. For TCP it would be preferable if the link layer service actually gets the packet through the wireless link, before reaching a retransmission timeout. We can allow ourselves to make many link layer transmission attempts, since they are cheaper than retransmitting a whole TCP segment. For UDP, on the other hand, applications don't require guaranteed delivery of every packet, but they want it fast, so fewer attempts should be made for UDP applications before giving up and discarding the frame.

3.1 Traffic

It has been stated that it is impossible to predict how the traffic in a future Internet will behave [8]. Not only is the number of nodes and connected servers growing exponentially, but also the applications running on clients will be abundant and impossible to predict today. It is therefore of vital importance that the intelligence built into the network can adapt to any new type of traffic presented to it.

To evaluate our approach we have chosen to mix several types of traffic. They all constitute sessions that exist for a predefined amount of time. Each session can have either deterministic, fixed values for packet sizes and packet inter-arrivals, or, they can be partly or completely drawn from some random distribution. For our purposes we have defined three traffic classes: VOICE, DATA, and MEDIA. These traffic classes are presented next.

VOICE requires a low latency but is on the other hand not sensitive to occasional errors or packet losses. It is also recognized by its small but frequent packets. To model this we have fixed-size and regularly arriving packets throughout the duration of a session.

DATA which is the opposite of the VOICE class, is recognized by its high sensitivity to errors and its high variability in both inter-arrival and size. This is modeled with a Pareto distribution for the size, and an inter-arrival time drawn from an exponential distribution.

MEDIA can be viewed as a mixture of the previous two classes; It has a relatively fixed rate, but the size of the packets can vary substantially, depending on the varying nature of e.g. streaming video encoding. It is more important to maintain the timing of the packets, than to achieve a completely error-free transmission of them.

The parameters related to the different classes are described in Table 1, and their respective random distribution functions, the Pareto distribution (1) where P is the packet size and S is the minimum packet size for the distribution, and the Exponential distribution (2) where T is the expected value of the inter-arrival time t, are given below.

$$F(P) = 1 - \left(\frac{S}{P}\right)^{\alpha}, \quad P \geq S \tag{1}$$

$$F(t) = 1 - e^{-\frac{t}{T}}, \quad t \geq 0 \tag{2}$$

3.2 Buffering and queueing

At the incoming side from the wired network we maintain a buffer with separate queues for the different traffic flows, also distinguishing the packets with respect to the destination, so that each source-destination pair has a separate incoming queue. All incoming packets are scanned for size, priority, session identification number, and required service level from the link layer.

In the case that the wireless link is unable to meet the requirements with regard to delay and throughput, we assume to have a selective packet discard policy [13] from the buffer, so that the impact on the applications is as small as possible. This information is stored in the buffer and can be accessed by the scheduler: The buffer controller is assumed able to submit a status report to the scheduling subsystem, described in Section 3.3, so that the scheduler can make an appropriate decision on which queues to choose for the next transmission frame.

The queues are emptied in a bit-by-bit manner, independently of the individual packet boundaries. The bit-stream is passed to the link-layer FEC/ARQ subsystem, along with information about the service requirements. The incoming buffer is described in Figure 1. At the receiving side of the wireless link, the packets have to be re-assembled, before

Table 1. The traffic source parameters and their service requirement parameters. FT stands for Fixed inter-arrival Time, and RS stands for Random Size. For the random size, we have a Pareto distribution, see equation (1), and for the random time, we have an Exponential distribution, see equation (2).

Class	Source parameters	Service parameters
VOICE	FTFS T=0.001 s S=256 bits	Prio=6 BER=10^{-3} Ntrans=3
DATA	RTRS T=0.01 s S=8000 bits, α=1.03	Prio=2 BER=10^{-5} Ntrans=8
MEDIA	FTRS T=0.005 s S=1280 bits, α=1.03	Prio=4 BER=10^{-4} Ntrans=3

passing them up to the network layer. This can be done since the scheduling decision is transmitted (broadcasted) to the receiving side, and it totally determines which byte belongs to which flow.

3.3 The Scheduling subsystem

The system we propose makes use of a channel predictor (Figure 2) and a multi-user time-slot scheduler that are mounted on top of the Hybrid type-II ARQ/AMS scheme. The scheduler creates a signaling pipe [12] between the network layer buffer and the link layer service, making them mutually aware of one-another. For instance, the network layer does not ask for a link service whenever there is data to transmit. Instead it notifies the scheduler of the incoming traffic by passing queueing information (A in Figure 1) about the amount of data and type of service that would be preferred by the packets. The scheduler then asks the link layer for a report (B in Figure 1) about how the channel conditions would meet the required service. This can be done since the link layer has access to channel prediction data of all the established connections: The link layer builds up an $M \times N$ matrix of channel quality predictions, where M is the number of time-slots, and N is the number of mobile hosts with ongoing sessions. These predictions cover the next following time-frame of $5ms$. The predictor has a prediction horizon of $10ms$. Figure 2 shows the channel predictor for one link. Based on the predicted values of the SINR for each of the N user's channels, and the target error rates , the initial code rate and signaling constellation are chosen in advance by the link layer, for a set of $M = 48$ future time-slots. It is then the task for the scheduler to, along with the queue sizes and priorities, distribute these time slots among the different queues in a fashion that maximizes some criterion, such as throughput or another measure of user satisfaction.

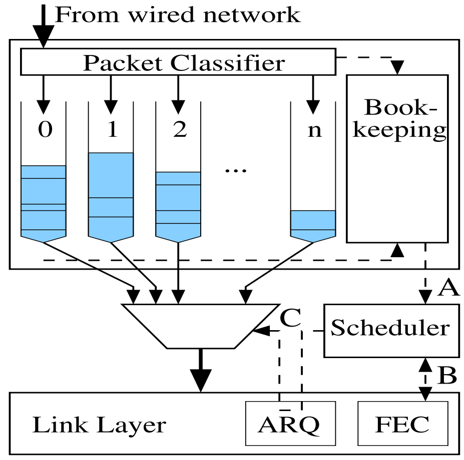

Figure 1. Schematic of the buffer and its queues, and how they interconnect to the scheduler and link layer. The packets arrive at the top and are appended to their respective queues. The buffer regularly submits a status report (A) to the scheduler, containing info about the priorities, the size of the queues, and the required link service, some of which is also passed to the link layer (B). The scheduling decision (C) is updated by the link layer ARQ, and is then used to drain the queues.

The scheduler performs the scheduling in two rounds. In the first round, each time slot is simply allocated to the user that can transmit at the highest rate in that time slot. If the buffered data in the queues had infinite size, this approach would actually maximize system throughput. However, since the buffers are not infinite, the scheduler runs a second round, where time-slots are redistributed from users that have been over-supplied (rich), to users that have been under-supplied (poor). We call this equalization to user satisfaction the Robin Hood principle: To take from the rich, and give to the poor. This algorithm is simple and it works as follows:

1. Find the rich and poor users by comparing their allocations to their amount of data in the queues
2. Loop until either no more rich or no more poor users exist:
   (a) For the richest user, find its worst time-slot, in the meaning of lowest transmission rate
   (b) Among the poor users, find the best user in that time-slot, and give the time-slot to him. In case two poor users have the same transmission rate, choose the one with the higher priority
   (c) Update the rich and poor variables
3. The resulting decision is a 48-element vector with one user ID per time-slot

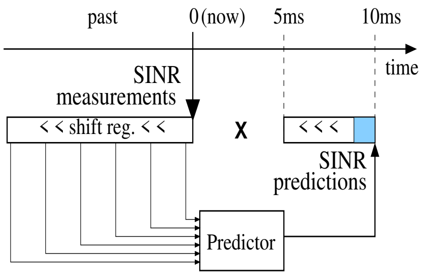

Figure 2. The input to the predictor are previous channel measurements, which are fed through a shift-register. The predictions are buffered for 5 ms in another shift-register. When it is full, it is passed to the scheduler, that makes the coding/modulation decision for each time-slot. At the point marked with an "X", the decision is broadcasted to the mobile terminals, and it will be executed in the next transmission frame.

The resulting decision is then distributed to the mobile hosts, via the control channel, described in Section 3.6. It is also passed to the link layer, that after filling some of the time-slots with occasional retransmission data, accepts new data from the queues through a modified decision (C in Figure 1).

3.4 The FEC/ARQ subsystem

Even though the intention is that the scheduling subsystem shall make decisions and select appropriate coding and modulation complexities, to avoid retransmissions, we cannot exclude the possibility of an erroneous reception. In the case of an erroneous reception, a NACK is sent back to the base station, telling it to take appropriate measures for the event. The coding rate/modulation is then chosen to be less than the one used in the previously unsuccessfully transmitted packet, according to the Hybrid type-II ARQ/AMS (HARQ-II/AMS) scheme. This scheme improves the robustness against the channel distortion while simultaneously utilizes the channel capacity efficiently due to the adaptive error correction and modulation facilities. Since the FEC/ARQ is in the link layer, there are no fixed requirements on a completely error free reception. Different services can be given to the network layer, but we can never guarantee an error free transmission of the data over the wireless link. One of the options we provide for the service is the number of attempts the link layer should try to transmit the data free of errors. For real-time applications such as telephony, where UDP might be the transport protocol, say, only three attempts should be made (Ntrans = 3 in Table 1) before giving up, and discarding the frame. For applications where TCP is used, the only constraint is the TCP retransmission timeout that, when it is reached, will invoke a new segment to be sent from the source. In this case, up to ten transmission attempts (Ntrans = 10) could be made before giving up.

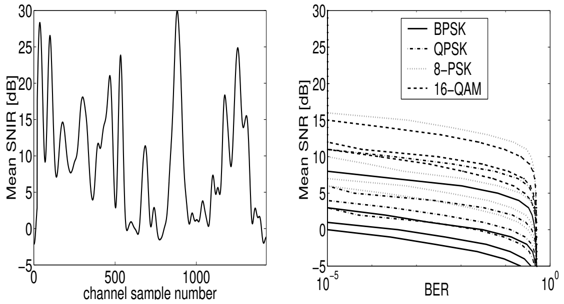

Figure 3. SNIR trend and the bit error probability for different combination of modulation and coding rates. For a fixed modulation, the BER corresponding to the different codes is decreasing by decreasing the coding rate.

The HARQ-II/AMS is based on the Rate Compatible Convolutional (RCC) codes at the parent rate $1/3$ with constraint length 7 where the higher and lower rate codes are obtained by optimum puncturing or repetition of the parent rate code according to a puncturing or repetition pattern with period 2, respectively [4]. In this scheme, the transmission of a lower rate code in response to the retransmission request, is performed by sending only the *incremental* redundancy bits due to the rate compatibility property of the RCC codes. Moreover, due to the AMS, the transmitter has the possibility to choose one of the 16-QAM, 8-PSK, QPSK and BPSK modulation schemes which satisfies the user demand the most in the time varying channel.

As explained in Section 3.3, the link layer specifies the most appropriate selection of the modulation/coding rate based on the target BER and the predicted values of the SNIR for each of the user's channel with ongoing sessions. The selected values together with the channel condition predictions are reported to the scheduler. For this purpose, the link layer uses the predetermined minimum values of the SNIR for a given target BER which is needed on the channel to meet the BER requirements. Since these thresholds vary for different combination of modulation/coding rates, the one with the least margin is selected among the schemes which the predicted channel SNIR value exceeds their threshold. Therefore, the evaluation of the BER performance for all the possible combination of modulation and coding rate is essential, however very complicated for the following reasons.

In general large signalling constellation such as 16-QAM and 8-PSK are difficult to analyze in a time varying channels, particularly, when convolutional coding is involved where the BER analysis is relied on the union bounds or other approximations which are known to be inaccurate at low SNIR. Therefore, instead of the analytical solution, a numerical approach is suggested. For each new transmission any modulation and coding rate can be used. However, taking into account the retransmission, increases the amount of combinations substantially. To avoid that we have evaluated the BER numerically without considering the history (the retransmission) i.e. for fixed modulation and coding rate. Moreover, the AWGN channel is assumed during one time-slot due to the very fast fading channel. The simulated BER is therefore evaluated for all the combinations of 16-QAM, 8-PSK, QPSK and BPSK modulations and the code rates 1, 2/3, 1/2, 2/5, 1/3, 1/4, 2/9, 1/5, 2/11 and 1/6. However, in order to have a clear and understandable picture, some of the simulated BER curves are illustrated in Figure 3 along with a realization of SNIR of a channel. We would like to mention that in this figure, 16-QAM at rate 1 is not shown due to its very poor performance at this rate. As shown later in Section 4, in spite of this simplification, very good results are obtained.

After reporting the predicted channel conditions, the appropriate modulation/coding rate for each user's channel who demands service, and the temporary priorities (i.e. the users with retransmission request), the scheduler allocates the time slots among the users and passes the decision to the link layer which constructs the frame as described in the following.

For each user with allocated time-slots, the priority is given to the erroneous packets with the retransmission request if they exist. With the constraint on having *fixed modulation* within a time-slot while variable coding rate is allowed for different packets, the appropriate modulation/coding rates is selected for each retransmitted packet. In this case, a lower rate code than the one in previous transmission should be chosen. This constraint stems from the fact that when the modulation is changed the receiver needs some time to be adjusted which may not be tolerable within a time-slot. But changing the coding rate, does not require different encoder/decoder and only changes the puncturing or repetition patterns [4] which is much less time consuming and therefore, tolerable.

The corresponding incremental code words are then generated, modulated and placed in the corresponding time-slots. Afterwards, the size of unused time-slots (for each users) is calculated and the modulation/coding rate is decided for new data packets based on the channel predictions and the target BER as described before. However, if there is retransmission within that slot, the modulation is predetermined due to the constraint on having fixed modulation within a slot. In this case, the code rate should be adjusted accordingly in order to satisfy the BER as much as possible.

When it is done, the link layer determines how many new data bits can be transmitted with this frame and asks for them from the buffer. The received data bits from the queue are partitioned into the data packets of 216 bits at the link layer, each containing 12 CRC bits for error detection and 6 zero tail bits corresponding to the memory of the convolutional encoder. In case of having data bits which are left but do not fill a link data packet, enough dummy zero bits are appended in order to form a data packet. After this stage, it is checked if there is any unused space within the time-slot. In case there is, it is investigated if a lower rate code can be used instead in order to use the vacancy in a time-slot for increasing the power of the code for error correction. Then the CRC encoding for error detection, convolutional encoding at the decided rate for error correction and modulation are performed for each new data packet. When it is accomplished for all the users which are time-slot allocated to, the frame is constructed and transmitted over the time varying channel. At each mobile host, the receiver performs optimum soft decoding by a Viterbi decoder at the parent rate $1/3$ which is aided by the perfect CSI. The decoded bits are fed into the CRC decoder determining if any error is detected or not. In case of error detection, if more retransmission is permitted, a NACK feed-back signal is sent to the link layer transmitter. Otherwise an ACK is fed back to the transmitter while the decoded data bits are delivered to the network layer to be prepared for the actual usage.

For the performance assessment of our proposed system which from now on is referred to as the predictive HARQ-II/AMS, we have considered two other systems for comparison. One is the so-called blind or non-predictive HARQ-II/AMS scheme where the link layer does not have access to any channel measurements and the selection of the modulation/coding rate is based on the predetermined values stored in a look-up table, similar to the one presented in [4]. In this system, the transmission starts using a modulation with large signalling constellation and a high rate code. During the retransmission, the size of constellation as well as the coding rate are reduced. The transmission is initialized with 8-PSK at rate 1. The next transmission attempts are performed by 8-PSK at rate $2/3$, followed by QPSK at rates $1/2$, $2/5$ and $1/3$ and BPSK at rates $1/4$ and lower. Moreover, only the information according to temporary priorities due to the users with retransmission request is provided for the scheduler. The second system is the so-called predictive AMS scheme where no coding and retransmission protocols are allowed but the link layer is provided with the channel predictions and an adaptive modulation system [3]. However, in this work we have used only 8-PSK, QPSK and BPSK where the corresponding BER performances are evaluated analytically as described in [9].

3.5 Time-slot format

Each time-slot comprises of user data, and a mid-amble of training data to aid the channel estimation process. Before and after the user data there are tails serving as guard intervals against bad timing in the reception or transmission. A descriptive image of the time-slot format is given in Figure 4. The figure also shows the location of the schedul-

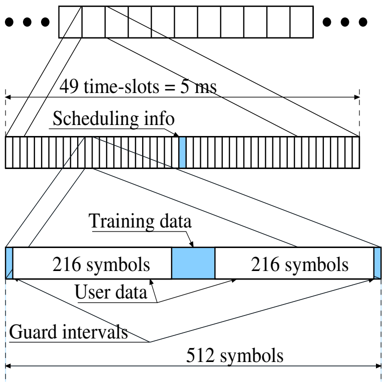

Figure 4. A down-link frame comprises of 49 time-slots, one of which is used for control (scheduling) information. Each time-slot begins and ends with guard intervals to ease slot timing. In the middle of the time-slot there is a sequence of training data, for the channel estimation.

ing information, within the frame of 49 time-slots. This scheduling info is further described in Section 3.6.

3.6 Access control information

A frame consists of 49 time-slots, each of which can be dynamically assigned to one user, except for one of the time-slots in each frame, which is dedicated to broadcast scheduling information about the next frame. A compromise between the base-station and the mobile terminal leads to the suggestion that the scheduling information is transmitted in one of the time-slots in the middle of the frame. In this way the scheduler has time to perform necessary calculations, and the mobile stations have time to adjust to the new schedule.

Since the scheduling information that is transmitted in the downlink is crucial for the efficiency of the multiple access scheme, it has to be well protected against errors. For this time-slot, only BPSK modulation with a low rate code should be used. However, the protocol must not fail in case of an erroneous reception of the scheduling information. There has to be a recovery possibility woven into the algorithm. At this early stage, we assume perfect signaling channel, so no such errors are modeled in our simulations.

4 Simulation

The simulator is written in C++, under the Linux operating system. Large parts of the code consists of a library of routines for mathematics, communications, and signal processing, developed at the Department of Signals and Systems, Chalmers University of Technology. It is our intention to provide our system as a module under NS (Network Simulator, see http://www-mash.cs.berkeley.edu/ns/), so that it is accessible to the communications community.

The simulations are carried out for the proposed system, referred to as the predictive HARQ-II/AMS together with

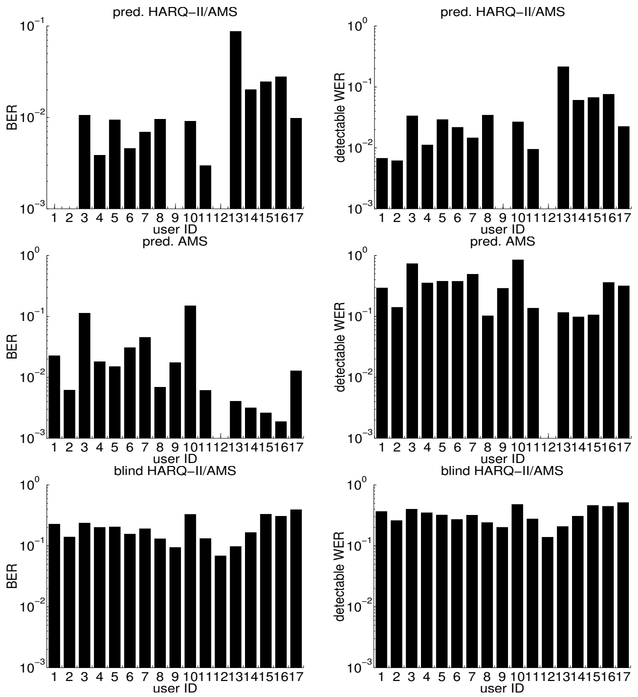

Figure 5. The bit error rate and detectable word error rate for the predictive HARQ-II/AMS, predictive AMS and blind HARQ-II/AMS for 17 users. The users belong to the classes VOICE, VOICE, DATA, DATA, VOICE, VOICE, VOICE, VOICE, MEDIA, DATA, . . . , DATA from left to right.

the predictive AMS and the blind HARQ-II/AMS described in Section 3.4. In general the BER, the detectable Word Error Rate (WER) and the throughput performance together with the average number of transmissions for each user is evaluated where the corresponding results are presented in Figures 5 and 6. The results show that in average for all the users, the least BER and WER are provided by the proposed system while the blind HARQ-II/AMS performs the worst as expected. This stems from the fact that having access to the prior information of the channel conditions results in a more appropriate choice of modulation/coding rate which satisfies the user demand. Furthermore, employing adaptive error correction capabilities increases the robustness of the system against the channel distortion.

Additionally, the predictive HARQ-II/AMS provides a high throughput almost for all the users which makes this system more favorable than the predictive AMS where the throughput for different users varies alot. A good throughput performance is also obtained by the blind HARQ-II/AMS at the expense of increasing the amount of retransmission requests. Moreover, the results have shown (but not shown here) that the blind HARQ-II/AMS occasionally delivers the erroneous data to the users due to the limitation in number of transmissions and considers them as correct

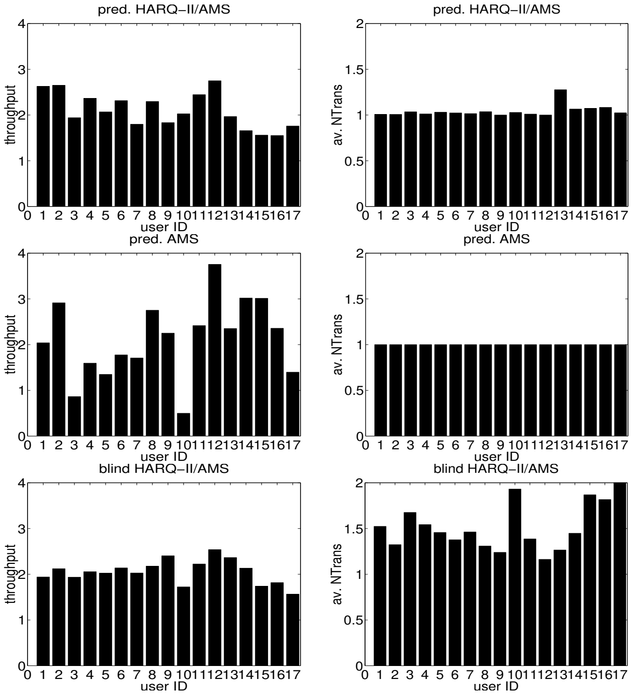

Figure 6. The throughput and average number of transmissions for the predictive HARQ-II/AMS, predictive AMS and blind HARQ-II/AMS for 17 users. The users belong to the classes VOICE, VOICE, DATA, DATA, VOICE, VOICE, VOICE, VOICE, MEDIA, DATA, . . . , DATA from left to right.

ones which are counted in throughput evaluation. However, this is not the case for the predictive HARQ-II/AMS and error free reception is accomplished before the number of transmissions reaches to its maximum value.

5 Conclusions

A combination of radio channel predictive time slot scheduling and link layer FEC/ARQ with adaptive modulation and coding, for IP (Internet Protocol) packet data, is presented and evaluated through simulations. The performance of the proposed system referred to as the predictive HARQ-II/AMS is evaluated and comparison is made when no channel state information is available at the transmitter (the blind HARQ-II/AMS) or a retransmission protocol and channel coding is not provided for the system (the predictive AMS). The results show that our proposed system reduces the system delay due to the prior information of the channel which is available at the transmitter. Additionally, applying channel coding improves the error correction capability of the system and provides robustness against the channel prediction errors, especially in situations where there are considerable changes in the channel conditions during the prediction time interval. Moreover, the error correction is done in an adaptive way, suitable for the time varying channel. Furthermore, employing the adaptive modulation system enhances the throughput by utilizing the channel capacity more efficiently. Therefore, the proposed system can provide a high throughput while keeping the BER and number of transmissions low for the users in the system.

References

[1] H. Balakrishnan, S. Seshan, E. Amir, and R. H. Katz. Improving TCP/IP Performance over Wireless Networks. In *Proc. MOBICOM*, pages 2–11, Berkeley, CA, 1995.

[2] T. Ekman and G. Kubin. Nonlinear prediction of mobile radio channels: Measurements and MARS model designs. In *ICASSP*, Phoenix, Arizona, March 1999.

[3] N. C. Ericsson. Adaptive Modulation and Scheduling of IP traffic over Fading Channels. In *Proc. IEEE Vehicular Technology Conference*, volume 2, pages 849–853, Amsterdam, the Netherlands, Sept. 1999.

[4] S. Falahati and A. Svensson. Hybrid type-II ARQ Schemes with Adaptive Modulation System for Wireless Channels. In *Proc. IEEE Vehicular Technology Conference*, volume 5, pages 2691–2695, Amsterdam, the Netherlands, Sept. 1999.

[5] D. J. Goodman, R. A. Valenzuela, K. T. Gayliard, and B. Ramamurthi. Packet reservation multiple access for local wireless communications. *IEEE Transactions on Communications*, 37(8):885–890, Aug. 1989.

[6] M. Kawagishi, S. Sampei, and N. Morinaga. A Novel reservation TDMA Based Multiple Access Scheme using Adaptive Modulation for multimedia Wireless. In *Proc. IEEE Vehicular Technology Conference*, pages 112–116, May 1998.

[7] M. Naijoh, S. Sampei, N. Morinaga, and Y. Kamio. ARQ Schemes with Adaptive Modulation/TDMA/TDD Systems for Wireless Multimedia Communication Services. In *Proc. IEEE International Symposium on Personal, Indoor and Mobile Radio Communications*, volume 2, pages 709–713, Helsinki, Finland, Sept. 1997.

[8] V. Paxson and S. Floyd. Why We Don't Know How To Simulate The Internet. In *Proc. Winter Simulation Conference*, Atlanta, Georgia, 1997.

[9] J. G. Proakis. *Digital Communications*. McGraw-Hill, New York, 3rd edition, 1995.

[10] C. Roobol, P. Beming, J. Lundsjö, and M. Johansson. A proposal for an RLC/MAC protocol for Wideband CDMA Capable of Handling Real time and non Real Services. In *Proc. IEEE Vehicular Technology Conference*, pages 107–111, May 1998.

[11] A. S. Tanenbaum. *Computer Networks*. Prentice Hall International, Upper Saddle River, New Jersey, 1996.

[12] G. Wu, Y. Bai, J. Lai, and A. Ogielski. Interactions between TCP and RLP in Wireless Internet. In *GLOBECOM*, pages 661–666, Rio de Janeiro, Brazil, December 1999.

[13] Z.-L. Zhang, S. Nelakuditi, R. Aggarwal, and R. P. Tsang. Efficient Selective Frame Discard Algorithms for Stored Video Delivery across Resource Constrained Networks. In *INFOCOM*, pages 472–479, New York, NY, March 1999.