A Radio Channel Emulator for WCDMA,
Based on the Hidden Markov Model (HMM)

Anna Umbert, Pilar Díaz
Polytechnic University of Catalunya (UPC)
Signal Theory and Communications Department
Campus Nord–D4, Jordi Girona 1–3, 08034–Barcelona, Spain
E–mail: [annau, pilar]@tsc.upc.es

Abstract

One of the main development and research subjects within the telecommunications area activity is the 3G mobile systems standardisation. The radio access is, of course, the main trouble in mobile systems, so it is important to investigate its implication.

This paper describes a radio channel emulator[1] for the UTRA–FDD mode, based on the Hidden Markov Model (HMM). Since a statistical system behaviour is needed to train the HMM, off–line simulations have been made.

The results between simulated and emulated statistics are presented. The use of emulation models implies a loss of accuracy with respect to simulation models, but is adequate to operate in real time. Certainly, the main advantage of using HMM in the emulator is the huge reduction in time, resources and effort with regard to a real simulation of the system. The emulator will allow in future works, for fast testing and comparison of several higher layer protocols and error control schemes.

1. Introduction

Nowadays, the standardisation process to specify the third generation mobile system is one of the main targets in telecommunication world. In 3G mobile systems a key aspect is the packet data services integration among all the services. Although data services are already present in fixed networks, i.e. Internet, and there exist several suitable protocols, the main trouble in mobile systems is the radio access, so it is important to investigate the impact of this access in the protocols used in packet mode services.

At this aim, a radio access network emulator has been developed for the UMTS Terrestrial Radio Access – Frequency Division Duplex (UTRA–FDD) mode, which is based on a Wideband–Code Division Multiple Access (WCDMA) technique. This emulator reproduces the lower layers of the UMTS radio interface, that is: the Physical Layer, the Data Link Layer and some functionality of the Network Layer. A block diagram of the emulator is shown in Figure 1.
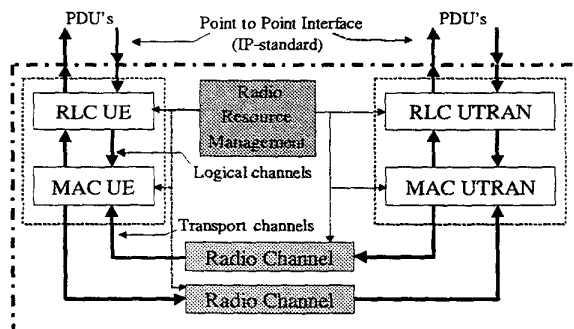

Figure 1. Block Diagram of the UTRA–FDD Radio Interface Emulator.

An important element within the emulator is the radio channel block that reproduces the behavior of the Physical Layer. To model this, Markov Chains based on the Hidden Markov Model (HMM) have been used. A statistical system behaviour is needed to train the HMM. To that end, off–line simulations have been made.

The simulation model implements each of the transmission chain elements, and can be considered quite accurate to a real system. So it has been used to derive the channel behaviour, mainly in terms of error distribution. On the other hand, the emulation model considers the system as a black box, and intends to reproduce its input/output behaviour without requiring

knowledge of the internal structure and processes that yield these results. The use of emulation models implies a loss of accuracy with respect to simulation models, but is adequate to operate in real time. Certainly, the main advantage of using HMM in the emulator is the huge reduction in time, resources and effort with regard to a real simulation of the system.

The designed emulator is a versatile tool that permits to analyse the impact of the radio access in the higher layer protocols. Moreover it permits to implement enhanced applications within the physical layer such as channel coding. In fact, the emulator has been used to analyse in real time the improvements provided by a convolutional code.

In next section the description of the emulator is presented. Then the main radio channel features that have been considered are explained. After this, the development process is commented and the validation process is shown through some examples. Some enhancement applications are also presented to show the versatility of the emulator. Finally some conclusions on the work done are presented.

2. Emulator description

The radio access network emulator shown in Figure 1 consists of different blocks according to the radio interface protocol stack shown in Figure 2.
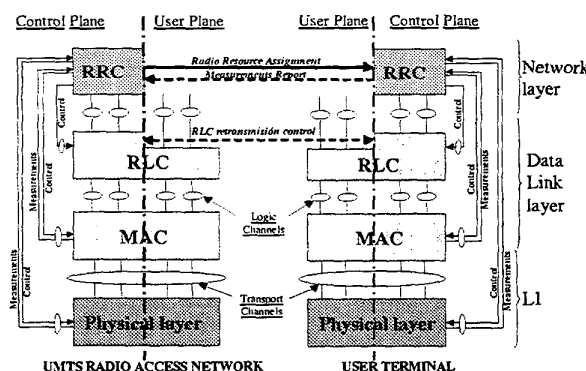

Figure 2. Radio Interface Protocol Stack.

The Radio Resource Management block belongs to the Network layer, and focuses in the Radio Resources Control (RRC). It is therefore charged to manage the available radio resources.

The RLC and the MAC blocks simulate the two sub-layers of the Data Link layer: the Radio Link Control layer (RLC) and the Medium Access Control layer (MAC). The RLC sub-layer provides segmentation and retransmission services, and can include a typical selective repeat acknowledge mechanism to provide error–free delivery. On the other hand, the MAC sub-layer maps the so–called logical channels into transport channels.

Lastly, radio channel blocks emulate the Physical layer using a Hidden Markov Chain.

The Hidden Markov Model (HMM) is a signal stochastic model [1]. The HMM principles were published in several papers by Baum and his colleagues in the late 60 and early 70. However, it did not become popular until 90. The HMM is an observable model in which the observation is a probabilistic function of the state. The resultant model is a double incrusted stochastic process with a subjacent stochastic process, which is not observable (it is hidden) and can only be seen through another set of stochastic processes that produce the observation sequence.

As a first step, an emulator for the CoDiT access scheme [2], which is based on a CDMA technique, has been developed [3]. The results are very pleasant since the emulation of the channel error characteristics is accurate enough, simple enough to implement, able to work in real time and flexible enough to cope with the requirements from different demonstration scenarios. The validation process consists in analysing the errors introduced by the HMM within the frame and comparing them with those of the real channel simulation. This analysis is done by means of the metric generated by the emulated channel and the simulation metric.

Our second step has been to obtain off–line simulation statistics for a WCDMA radio channel, in order to train a based HMM radio channel emulator for the physical layer of the uplink in UTRA–FDD. This will allow in future works, for fast testing and comparison of several high layer protocols and error control schemes.

To go into detail, a Hidden Markov Model consists of a Markov chain that has a number of states. Each state represents a range of number of errors, and they should be as close to equiprobable as possible, thus having non–uniform range each, the ranges are selected adaptively based on the simulated data. For each state the minim error, the maxim error, the error mean and the state probability are needed.

Besides state information, the HMM has another parameter, probably the most important one: the state transition matrix. This matrix contains the hop probabilities between the different states of the chain.

A diagram of a Hidden Markov Model is illustrated in Figure 3. The Markov Chain, state information, and the transition matrix appear. The transition probability between states is referenced as "a_{ij}", where "i" indicates the current state and "j" indicates the next state.

Figure 3. Hidden Markov Model diagram.
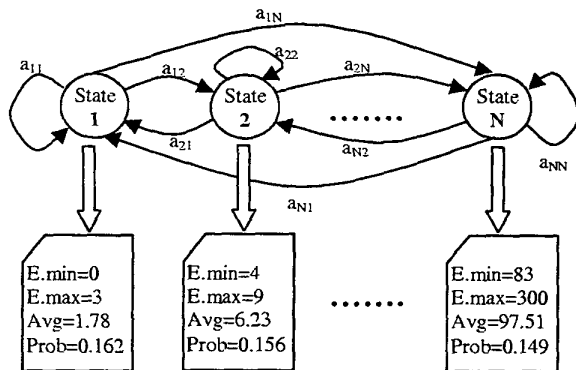
Transition matrix:

a_{11}	a_{12}	....	a_{1N}
a_{21}	a_{22}	....	a_{2N}
$\vdots$	$\vdots$	....	$\vdots$
a_{N1}	a_{N2}	....	a_{NN}

3. Radio channel features

The uplink channel simulation will consider different spreading factor (SF), environments, and simultaneous users. In table 1 the relation between the spreading factors and the data rate is shown.

SF	Data rate
4	960 Kbps
8	480 Kbps
16	240 Kbps
32	120 Kbps
64	60 Kbps
128	30 Kbps
256	15 Kbps

Table 1. Relation between SF and Uplink Data rates.

Three different demonstration scenarios have been considered with regard to the environment:

- 120 Km/h Outdoor
- 50 Km/h Outdoor
- 3 Km/h Indoor

where 120,50 and 3 are the chosen mobile speed.

The multipath fading propagation conditions proposed by 3GPP [4] have been selected. The outdoor Power Delay Profile (PDP) is shown in Figure 4, and the indoor PDP is shown in Figure 5. All taps have classical Doppler spectrum.
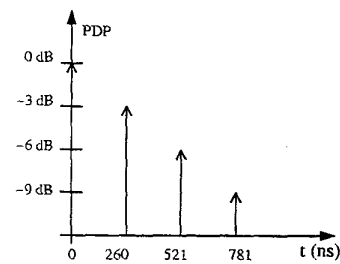

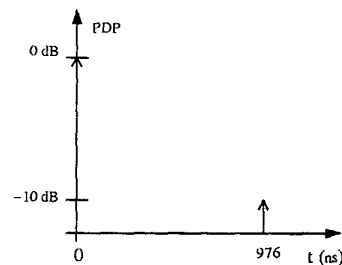Figure 4. Outdoor Power Delay Profile.

Figure 5. Indoor Power Delay Profile.

The chip rate selected in the 3GPP UTRA–FDD mode is 3.84 Mcps [5], so, in outdoor PDP, taps different from the principal are situated in one chip period (Tc), two chip periods (2 Tc) and three chip periods (3 Tc). Unlike, in indoor PDP, there is only one secondary tap situated in 3.75 Tc, and attenuated 10 dB. Note that indoor radio channel is worst than outdoor radio channel.

In the UMTS Terrestrial Radio Access (UTRA) the physical layer of the uplink makes use of Orthogonal Variable Spreading Factor (OVSF) codes to preserve the orthogonality between uplink channels of different rates and spreading factors from one user. After being spreading, the uplink channels are modulated and finally, in order to control the intra–cell and inter–cell interference, every mobile station multiplies the global uplink signal with a scrambling code which may be long (Gold code) or short (S(2) code) [6].

In the base station, and due to the multipath propagation, the original orthogonality is partially lost. This effect is more remarkable in the outdoor environment, where the delay spread of the mobile channel is equivalent to several chip intervals. A good estimation of the effects of multipath in the UTRA uplink requires detailed chip level simulations in a realistic environment, and these simulations last a lot of time.

The uplink channel impulse response is estimated from pilot bits present in each slot. This makes possible the demodulation of the physical channels using a RAKE receiver. Since each slot has 2560 chips and the channel

response change faster, a channel estimation error in the RAKE operation has been considered. The RAKE receiver is considered to have 4 arms, the delays of which are dynamically adapted to track the points of maximum energy in the instantaneous global impulse response.

The interference caused from other physical channel multiplexed and transmitted with the same spreading code (that is from the same user) is negligible, that is why to compute the intra–cell interference only active connections from other users in this cell have been considered.

With respect to the inter–cell interference it is proposed to keep its power variable and equal to a certain percentage of the total intra–cell interference power. The percentage will be around 60%.

Besides user interference, radio channels always include thermal Gaussian noise.

Another feature in the uplink is the use of a power control loop to guarantee that, regardless of the distance, the base station receives a fixed signal level. Power control has been simulated.

4. Development Process

To train the channel emulator based on the Hidden Markov Model (HMM) statistical system behaviour is needed. Off–line simulations of the radio channel have been made to obtain this statistical behaviour.

Once the emulator is trained, the statistics obtained through the simulations can be destroyed since they are no more necessaries. Only the HMM parameters are necessaries to reproduce the channel behaviour.

In future, only an extended pool of HMM parameters can be kept. So, to choose the appropriate HMM parameters is enough to emulate a given radio channel.

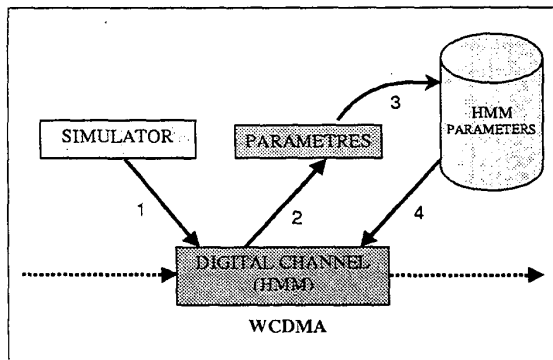In Figure 6 a diagram with the different steps of this process is shown.

Figure 6. Development process diagram.

5. Results

The modelled channel statistics have been compared with the simulated channel statistics used to train the HMM, to validate the emulator. This comparison has been done by computing the probability distribution function (p.d.f.) of:

- the number of errors in a packet or frame,
- the length of the burst error, as well as,
- the length of the error–free interval.

To make the former histogram, the number of frames with the same number of errors, k, has been computed. This value divided by the total number of frames employed in the simulation/emulation gives the probability of having k errors within a block.

For the histogram of the length of error bursts the number of error bursts with a given length, k, within a frame has been computed and divided by the total number of bursts. Error bursts of length zero are not considered. The obtained value gives the probability of having error bursts of length k within a frame.

And to make the last histogram, the number of free error intervals of a given length, k, has been computed and divided by the total number of intervals. Intervals of length zero are not considered. The obtained value gives the probability of having free error intervals of length k within a frame.

An example of validation is illustrated from Figure 7 to Figure 9. The results shown in the figures correspond to the outdoor environment with mobile speed 120 Km/h, using a spreading factor 128 and with Eb/No=6dB. Each frame has 300 bits length, and the Bit Error Rate (BER) is 1.68E–02.
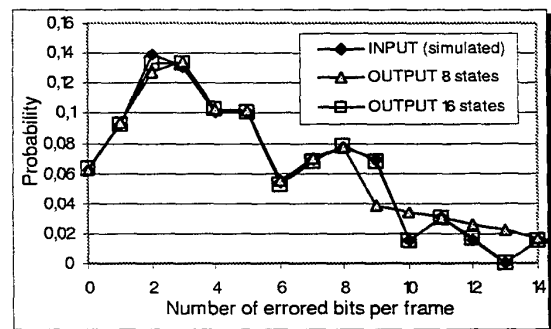

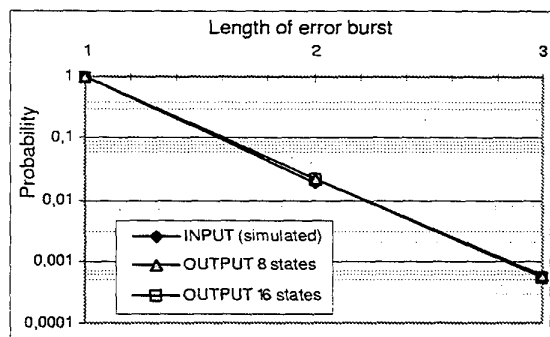Figure 7. Histogram of error number. per frame.
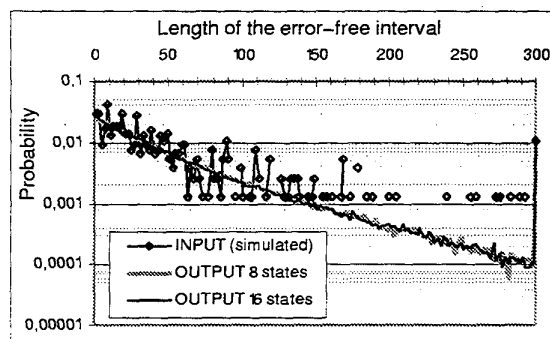
Figure 8. Histogram of burst length.

Figure 9. Histogram of the length of the error-free interval.

As you see, in this example the emulation models with 8 and with 16 states have been compared. It is important to point out that the number of states used in the HMM is a relevant parameter. If the number of states is higher, the model is more precise, but the number of parameters to save also increases. Otherwise if the number of states decreases, less parameters are needed, and the model loses precision.

In the previous example, as far as the 8 states model is considered, a discrepancy is observed in the histogram of error number per frame starting from point 9, as well as in the histogram of the length of the error-free interval. In both statistics, the HMM generates an average of the real behaviour, that is why it does not have enough information to reproduce the real case.

Increasing the number of states can partially solve this problem. In fact, the histogram of error number per frame (in figure 7) is perfectly emulated, whereas the histogram of the length of the error-free interval (in figure 9) does not improve.

Certainly, the real statistic of the length of the error-free interval change very often, and the HMM does not keep this information within its parameters, so it is only capable to reproduce the tendency of this statistic.

It is therefore important to choose the correct number of states for the emulator. To this aim, a process to choose the appropriate value has been defined.

Another example of validation is illustrated from Figure 10 to Figure 12. This time the results shown correspond to the outdoor environment with mobile speed 50 Km/h, using a spreading factor 32 and with Eb/No=4dB. Each frame has 1200 bits length, and the Bit Error Rate (BER) is 1. 87E–02.
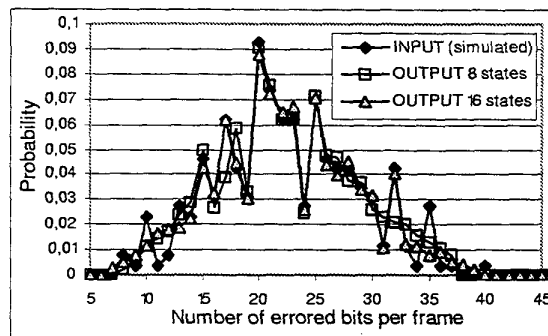

Figure 10. Histogram of error number per frame.
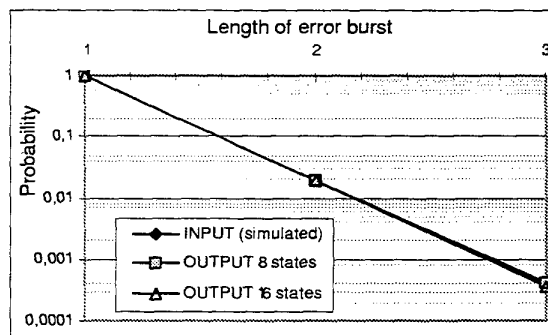

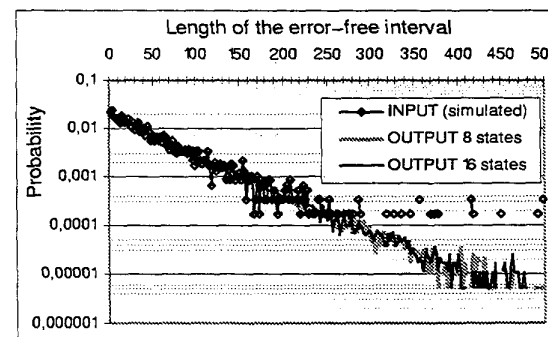**Figure 11. Histogram of burst length.**

Figure 12. Histogram of the length of the error-free interval.

Note that in this case the error–free interval statistic is less variant, so the HMM reproduces better this characteristic. Certainly, in this case the discrepancies appear near a probability of 0.0001.

These two examples show the precision of the HMM for a WCDMA technique when employed to emulate the errors introduced by the mobile channel. An important number of cases has been analysed to validate the HMM–based emulator used as error generator and the results obtained have been also good enough.

After a laborious analysis the adequate number of states for each different channel has been selected. In short, very bad channels (BER under 10^{-1}) need 32 states, bad channels (BER under 10^{-3}) need 16 states, good channels (BER around 10^{-4}) need 8 or 4 states, and very good channels must be emulated using 2 states. Only power two values have been selected in order to facilitate the computational implementation.

6. Enhancement applications

The designed emulator is a versatile tool, which can emulate an important number of radio channels with a WCDMA technique, but it can also provide enhanced applications. In this sense, the emulator has been modified to include channel coding in the HMM.

The whole process outline consists in encoding the data to transmit, send the data through the radio channel emulated (the one previously presented), and decode the data. The comparison between the former data and the latter gives the statistical behavior of the chain. Again, with this statistical the HMM can be trained to include channel coding. This process is shown in Figure 13.
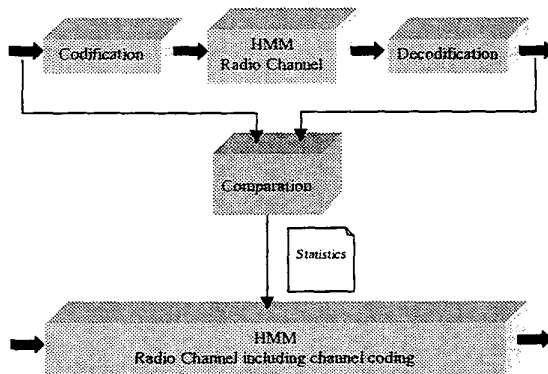

Figure 13. Process to train a HMM including codes.

Next an example using a convolutional code of rate R=1/3 and constraint length 5 is presented. In this example the HMM radio channel emulates a WCDMA channel with spreading factor 64, and Eb/No=2dB. The results shown in Figures 14 – 16 illustrate the statistics obtained when the HMM includes code and when the code is separately processed.
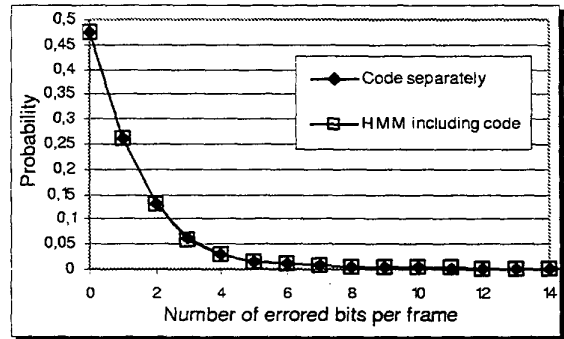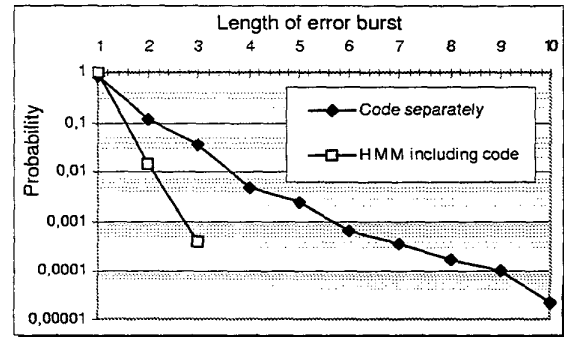

Figure 14. Histogram of error number per frame.

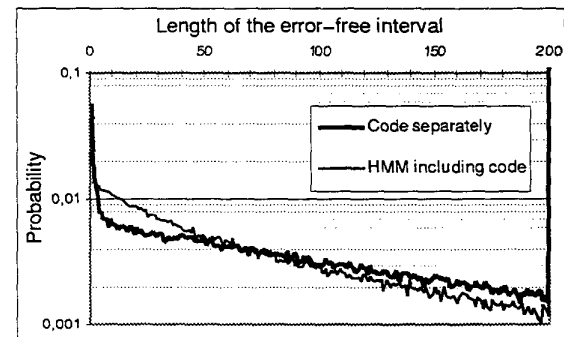Figure 15. Histogram of burst length.

Figure 16. Histogram of the length of the error–free interval.

Apparently these results are more exactly than those obtained without considering codes. That is why in this case the BER is better due to the code. In fact, the radio channel BER without code is 6.56E–02, and with code BER is 5.83E–03. Moreover in this case the lost of accuracy of the HMM radio channel is present in both curves.

Even though there is a lost of accuracy, the HMM emulator including codes is much faster than the equivalent simulated system. Thus, other enhanced applications can be made with the HMM including codes, such as to analyze the BER improvement due to a certain code as shown in Figure 17 for the mentioned convolutional code.

It is obvious that this tool allows to determine a system performance in terms of BER with an important saving of simulation time and resources.
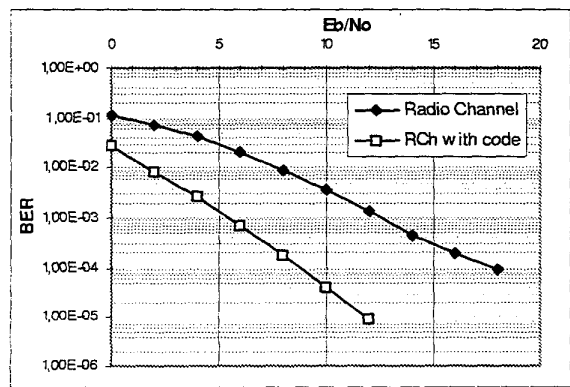
Eb/No

	0	5	10	15	20

1,00E+00
1,00E-01 ←— Radio Channel
1,00E-02 —□— RCh with code
BER 1,00E-03
1,00E-04
1,00E-05
1,00E-06

Figure17. BER improvement due to the convolutional code (R= 1/3).

7. Conclusions

In this paper a radio channel emulator based on the Hidden Markov Model has been presented. This emulator has been thought to operate within a radio access network emulator that has also been described. The considered radio channel features for the uplink have been described to show that the emulator has been made as close as possible to a real system. The development process of the channel emulator has been shortly described, and two examples of the validation process have been shown. An enhanced application including channel coding in the Hidden Markov Model has been finally presented.

The emulator has been validated against off-line simulations to show the model accuracy. In this validation, the number of states for the HMM is a key aspect, and through a hard analysis, its value has been adequately selected.

Although this emulator has been developed to be used within a radio interface emulator for a WCDMA technique, it is applicable to any analysis work, which requires sending data through a digital mobile communication channel. It is therefore a versatile and flexible tool.

Only a few parameters need to be saved to use the emulator. Thus, a database containing the parameters of different propagation channels can be made, and the emulator can easily emulate a great number of radio channels.

The huge reduction in time, resources and effort with regard to a real simulation of the system are the main advantages of the emulator. Certainly, the real time operation is one of the best emulator features. Some examples presented in this paper show that the emulator is accurate enough although there is a lost of accuracy. With these advantages the emulator becomes a powerful tool for other enhanced applications since it allows for determination of system performance in terms of BER with an important reduction in simulation resources and time.

8. References

[1] Lawrence R. Rabiner, "A tutorial on Hidden Markov Models and Selected Applications in Speech Recognition", Proceedings of the IEEE, Vol 77, No.2, February 1989.

[2] PG Andermo, "Overview of CoDiT Project". Race Mobile Telecommunications Summit, pp 33–42, Cascais, Novembre 1995.

[3] Anna Umbert, Pilar Díaz, "A Generic Channel Emulator to Evaluate Higher Layer Protocols in a CDMA System", PIMRC congress, London, September 2000.

[4] 3rd Generation Partnership Project (3GPP), "UE Radio transmission and Reception (FDD)", 3G Technical Specification 25.101 v3.1.0, December 1999.

[5] 3rd Generation Partnership Project (3GPP), "Physical layer –General description", 3G Technical Specification 25.201 v3.0.1, December 1999.

[6] 3rd Generation Partnership Project (3GPP), "Spreading and modulation (FDD)", 3G Technical Specification 25.213 v3.1.1, December 1999.