TSG-RAN Working Group 1 meeting #6 ***TSGR1#6(99)819***
Espoo, Finland
July 13-16, 1999

Agenda Item:

Source: InterDigital Comm. Corp.

Title: Simulation of Forward Error Correction for TDD RACH

Document for: Information and Discussion

Introduction

TSGR1(99)707 presented analysis of two Forward Error Correction (FEC) techniques for the TDD RACH preamble.

- A block encoded technique based on an 84,56 code
- A half rate convolutionally encoded block.

That analysis showed that the half rate convolutional encoder provided better performance on the assumption of AWGN and an ideal channel.

This document provides simulation results for the same two approaches, and this simulation will show that the basic conclusions are unchanged; i.e. the convolutional encoded RACH burst has better performance than the block coded technique by nearly 3dB.

The document also shows the performance for a TDD RACH format which supports 20 octets of payload, as required by TSG RAN WG2.

Comparison of half rate convolutional coding and 84,56 block code

Simulation Conditions

- Spreading sequence: OVSF (SF = 8 for ½-rate convolutional coding, SF = 16 for block coding)
- Data modulation: QPSK modulation (1 QPSK symbol = 2 coded bits)
- Pulse shaping: Raised cosine pulse with alpha=0.22
- Channel model: ITU channel with Vehicular type A (Time invariant channel for a single time slot)
- Data Demodulation: DMF (discrete chip-matched filter) with perfect channel estimation
- Normalization is performed to have the QPSK symbol energy after demodulating, equal to the transmitted QPSK symbol energy.

56 bits payload case (Half-slot RACH format)

½-rate Convolutional code

- Input to convolutional encoder: 56 bits payload + 16 CRC bits + 8 tail bits
- Output from convolutional encoder: 2*80 coded bits
- QPSK output: 80 QPSK symbols + 4 QPSK symbols extra: **84 QPSK symbols**
- Burst type (SF = 8):
- Decoder: Viterbi decoding
- Frame error rate (FER) calculation: It is assumed that a burst (frame) error occurs when there is at least one bit error.

Energy per bit (Eb) for ½-rate convolutional code

$$E_b \quad (dB) = E_s + 10\log 10\left(\frac{84}{56}\right),$$ where Es: QPSK symbol energy = 2*E(coded)

Block code case

- Input to Block encoder: 56 bits payload
- Output from Block encoder: 84 coded bits (including 28 CRC bits)
- QPSK output: 42 QPSK symbols

- Burst type (SF = 16):

- Frame error rate (FER) calculation: It is assumed that based on the demodulator output bits, a burst (frame) error occurs when there are more than 2 bit errors among 84 bits (56 bits payload+28 CRC bits).

Energy per bit (Eb) for block code

$$E_b \quad (dB) = E_s + 10\log 10\left(\frac{42}{56}\right),$$ where Es: QPSK symbol energy = 2*E(coded)

<u>Simulation results</u>

As shown in Figure 1, in the presence of multipath channel based on ITU channel mode and AWGN channel, the half rate convolutional encoding performs better than the block encoding with about 3 dB gain over the range of FER considered here.

These results reinforce the conclusion that half rate convolutional encoding is superior to the 84,56 block coding technique. A 3 dB advantage is shown even while including the overhead for the additional error detection field (16 bits) and the Viterbi Decoder Tail (8 bits).

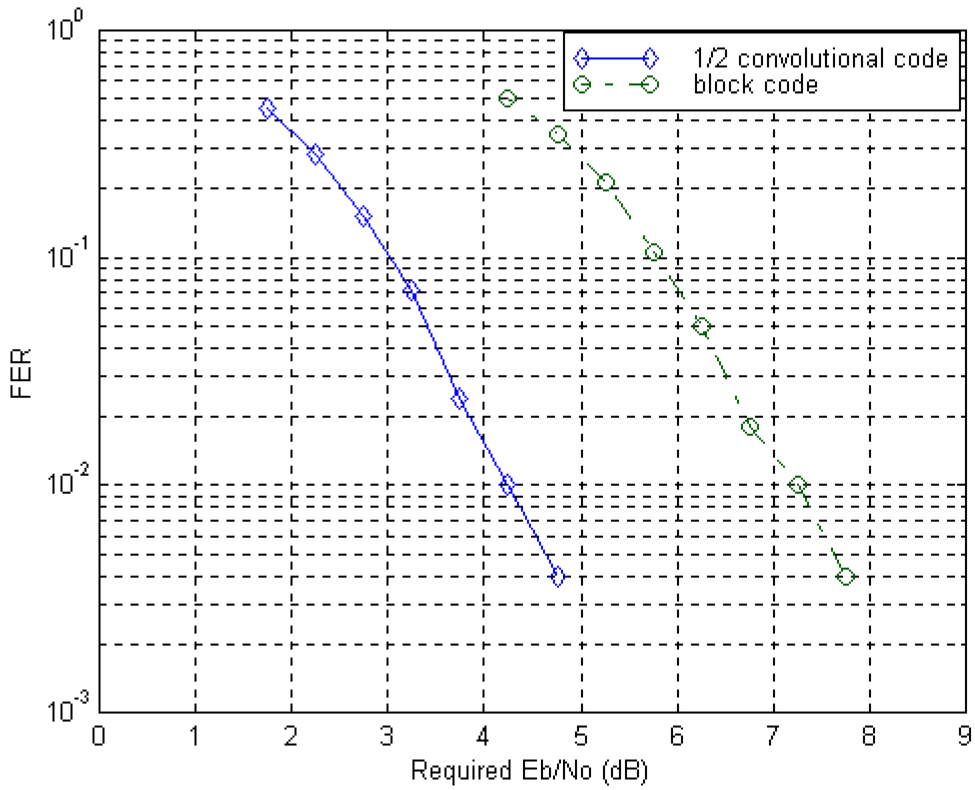

Figure 1: Frame error rate vs. Eb/No for ½-rate convolutional code and block code

2) 20 Octs (160 bits) payload case (Full- time slot RACH format)

This section presents simulation results for a proposed TDD RACH format. Half rate convolutional coded, full slot RACH, using the standard rate matching algorithm.

Specifications

- Input to convolutional encoder: 160 bits payload + 16 CRC bits + 8 tail bits
- Output from convolutional encoder: 2*184 coded bits
- In order to fit the TDD RACH bursts, the rate matching algorithm is used, which is specified as in TS 25.222.
- QPSK output: 184 QPSK symbols + 40 QPSK symbols (by applying rate matching (repetition)): 224 QPSK symbols
- Burst type (SF = 8):

112 symbols	Midamble (512 chips)	112 symbols	GP
625*m* sec			

- Demodulating repetition bits : To implement the (inverse) rate matching, the process combined the original QPSK symbol and its repeated version.
- Decoder: Viterbi decoding

- Frame error rate (FER) calculation: It is assumed that a burst (frame) error occurs when there is at least one bit error.

Energy per bit (Eb) for ½-rate convolutional code

$$E_b \quad (dB) = E_s + 10\log10\left(224/56\right),$$ where Es: QPSK symbol energy = 2*E(coded)

Note that as compared with 56 bits payload, given the QPSK symbol energy (Es), for the 20 Octets payload case we can take advantage of about 0.3 dB gain in Eb due to less overhead bits, as follows:

$$G = 10\log10\left(84/56\right) - 10\log10\left(\frac{224}{160}\right) = 0.2996(dB)$$

Simulation results

Figure 2. Shows that the TDD RACH burst with 20 Octs payload has about 0.5 dB better Frame Error Rate performance than the case with the 56 bits payload. There are two reasons for the performance gain.

- The overhead for CRC and Tail is a smaller fraction of the total burst energy(0.3 dB gain) and
- The repetition gain (around 1dB) because of the symbol repetition introduced by the rate matching process.

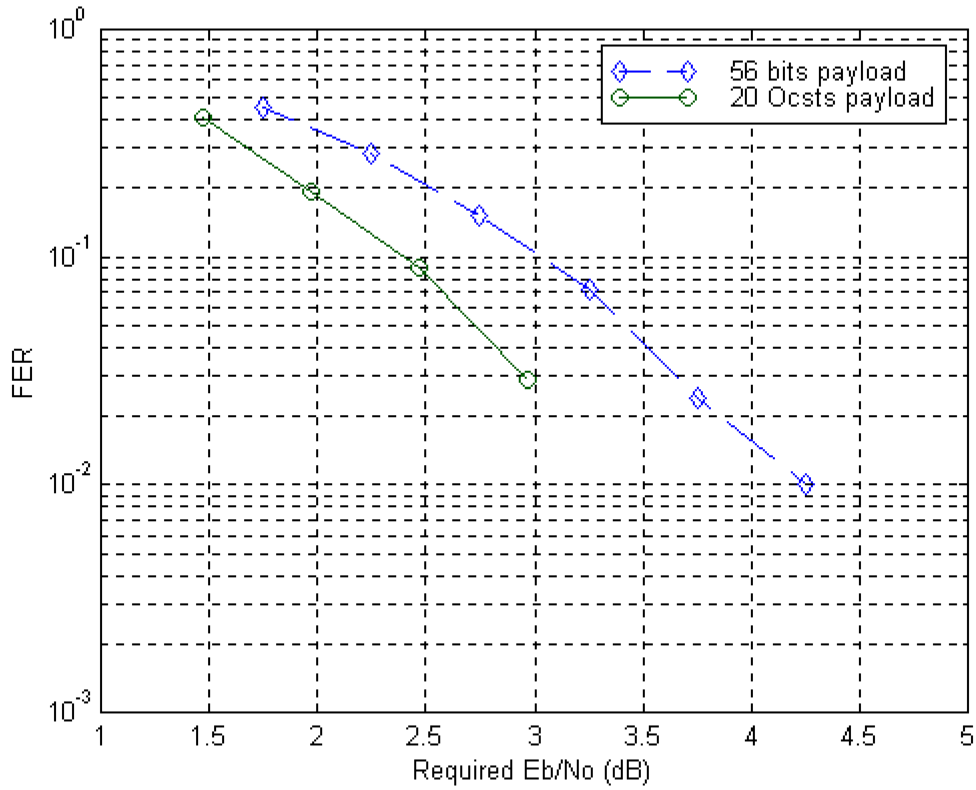

Figure 2: FER vs. Be/No for 20 Cots (160 bits) payload and 56 bits payload

Recommendations and Conclusions

- Convolution encoding and SF=8 are recommended for the TDD RACH.
- Half rate convolution encoding is nominally 3 dB superior to the 84,56 block coding technique, and should be used, independent of whether the half slot or full slot RACH is employed.
- A full slot RACH with SF=8 and half rate coding will provide a solution with 20 octets payload and performance as shown in figure 2.