A Robust Viterbi Algorithm for Symbol Recovery in the 1900MHz PCS Band

Markus Rupp, Rajeev Krishnamoorthy and Sayandev Mukherjee

Wireless Research Laboratory, Bell-Labs, Lucent Technologies

Keywords: Equalizer, Viterbi, TDMA

POC: Markus Rupp, Wireless Research Laboratory, Lucent Technologies, 791 Holmdel-Keyport Rd.,
Holmdel NJ 07733-0400, tel: (908) 888 7104, fax: (908) 888 7074, email: rupp@lucent.com

Abstract

Cellular TDMA-based handsets for the PCS 1900MHz band will soon become very popular. With the cellular band at 900MHz being congested by too many users the new band offers the service to twice as many costumers. However, the requirements on the symbol recovery become much harder since a Doppler speed twice as high as in cellular band is expected. Further impairment is given by the hardware realization that introduces timing errors and frequency offsets. This paper describes a Viterbi based equalizer-scheme that performs well under these conditions and satisfies the standards for cellular phones.

1 INTRODUCTION

In the design of an equalizer for a cellular handset various impairments have to be taken into account.

- Impairments of the channel (Doppler speed, delay spread, additive noise)

- Impairments of the hardware realization (timing error, frequency offset)

While the impairments for the channel are well defined in the standards (PN-3386, see [1]), the impairments of the phone are largely dependent on the hardware design. In our design we do not assume a perfect timing recovery (non-coherent receiver). We also expect that the PLL circuit introduces a small frequency offset according to the accuracy of the crystal in use. Furthermore, we assume the RF of the phone to introduce a noise figure of 8dB.

The PN-3386 standards define minimum requirements for the transmission over channels with Doppler frequency offset (in terms of vehicle velocity) and delay spread. At vehicle speeds of 100Km/h for flat Rayleigh fading the Bit-Error-Rate (BER) needs to be below 3% for a Signal-to-Noise-Ratio (SNR) of 19dB, and for delay spreads of up to one symbol $T = 41.2\mu s$ the 3% BER is required at SNR=22dB. The SNR values

were computed from the receiver strength values defined in the standards assuming the above noise figure of the phone.

The modulation scheme used in the TDMA cellular phone system is a $\pi/4$-offset DQPSK. Thus, the four possible symbols are taken each time alternatingly from one of the two sets $\{1, j, -1, -j\}$ and $\{e^{j\pi/4}, e^{j3\pi/4}, e^{j5\pi/4}, e^{j7\pi/4}\}$. Figure 1 depicts the possible constellation. The down-link (from base sta-
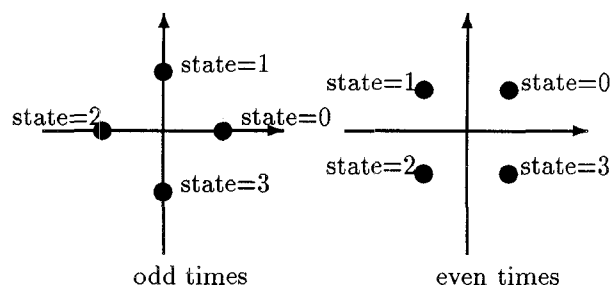

Figure 1: *The two alternating 4PSK schemes for $\pi/4$-offset-QPSK.*

tion to handset) transmission is organized in frames of 40 ms duration. Each frame consists of six slots that are identified by a sync word of 14 known phase transitions. In full rate mode one user acquires two slots of each frame. Thus, three users can be served at every frequency. In a half rate mode the user acquires only one slot each frame. This allow to double the user number by reducing the bit rates correspondingly. In the center of each slot is a Coded Digital Verification Color Code (CDVCC) that identifies the base station to the handset. It is an eight bit word protected by a four bit shortened (12,8) Hamming code. Thus, the CDVCC is defined by six known phase tran-

sitions. Each slot contains 162 symbols of which only 142 carry unknown information.

In this paper we propose a Viterbi algorithm with tentative decision that is able to satisfy the requirements defined in the standards under various impairments. We hereby utilize the structure and a-priori information defined by the standards. Simulation results show the robust performance under all conditions addressed here.

2 THE VITERBI ALGORITHM FOR PCS

Although being a standard procedure, the Viterbi algorithm for channel equalization[2] offers much freedom in its design. Parallel to the Viterbi a channel tracking algorithm has to run. The Least-Mean-Squares (LMS) algorithm has been chosen for this purpose since it is well suited to the driving sequence of (randomly) white symbols with equal modulus.

Running the algorithm straight through a slot does not give high performance since once the tracking runs into a deep fade, it can hardly recover afterwards. Therefore, it is much better to run also backwards beginning from the sync word of the adjacent slot. A further improvement is given if the known information on the CDVCC that is located in the middle of each slot, is used as well. In this case the algorithm is partitioned into four parts, each two are running forward and backward. The end of the run of each partition is precomputed by a fade-location algorithm.

The LMS algorithm is also utilized to obtain an initial channel estimate. Since the sync words at beginning and end of the slot are a-priori known, they are used as training sequence and deliver accurate results. As mentioned before the CDVCC is much shorter (only six transitions) than the sync words. Running the LMS over the CDVCC therefore delivers very poor estimates. Improvement could be made by repeating the adaptation several times.

Since we do not run a timing recovery, a T/2 spaced sampling is used. A symbol recovery unit finds the optimal sample location to start a slot. The channel can only be tracked with a certain length, i.e., the number of coefficients that are used to represent it. It is well known that the tracking capabilities of an adaptive algorithm decrease with increasing number of coefficients. On the other hand the accuracy of estimating a fixed, time-invariant channel increases with the number of coefficients. Thus, a compromise needs to be found that offers fast enough tracking combined with

sufficient accuracy[3]. We decided to use three coefficients for channel estimation. Since we work with T/2 sampling, we choose two sets of three T-spaced parameters. Both sets are used when computing the metric for the Viterbi algorithm.

Since we are using three coefficients, the space we would have to search through is $4^3 = 64$. In order to keep complexity small we make a tentative decision for the latest symbol and search only through $4^2 = 16$ possibilities. This reduces the complexity to an amount that can be handled by a standard signal processor.

3 SIMULATION RESULTS

Figure 2 depicts BER for a flat Rayleigh fading channel as well as a channel with $T/8$ and $T/4$ delay spread when using a simple differential detector at Doppler speed of 200Km/h. The figure shows plots over a wide SNR range covering the more severe conditions. Although the result is excellent for flat Rayleigh fading introducing a small delay spread leads to a big loss in performance. Moreover, if the sample timing is not perfect even the ideal flat Rayleigh fading case can result in a poor performance. This is indicated in the figure by triangle markers. For the $T/4$ delay spread case on the other hand the timing error does not cause any further distortion. The following paragraphs present the performance of the Viterbi algorithm under various impairments.

3.1 Performing the Standards

Figure 3 depicts the behavior of the Viterbi algorithm for flat fading and delay spread conditions ($T/4, T/2$, and T delay spread) when the sample timing is perfect and the location of the sync word is known. These curves thus define a reference to those discussed later on when we deal with timing error and sync word location estimation. The worst scenarios are given for flat fading and small delay spread. The value for flat fading at 19dB is 2.4% BER and for $T/4$ delay spread at 22dB is 1.6%. This offers quite a margin to the required 3%.

Also in Figure 3 it is shown how the algorithm reacts on a $T/4$ timing offset. At 20dB the markers ($\times$) show an increase in BER for all four cases. The increase for the flat Rayleigh fading is so strong that it does not satisfy the required 3% any longer. On the other hand using $T/2$ shifted samples for this case (which are available) even drops the BER for all cases
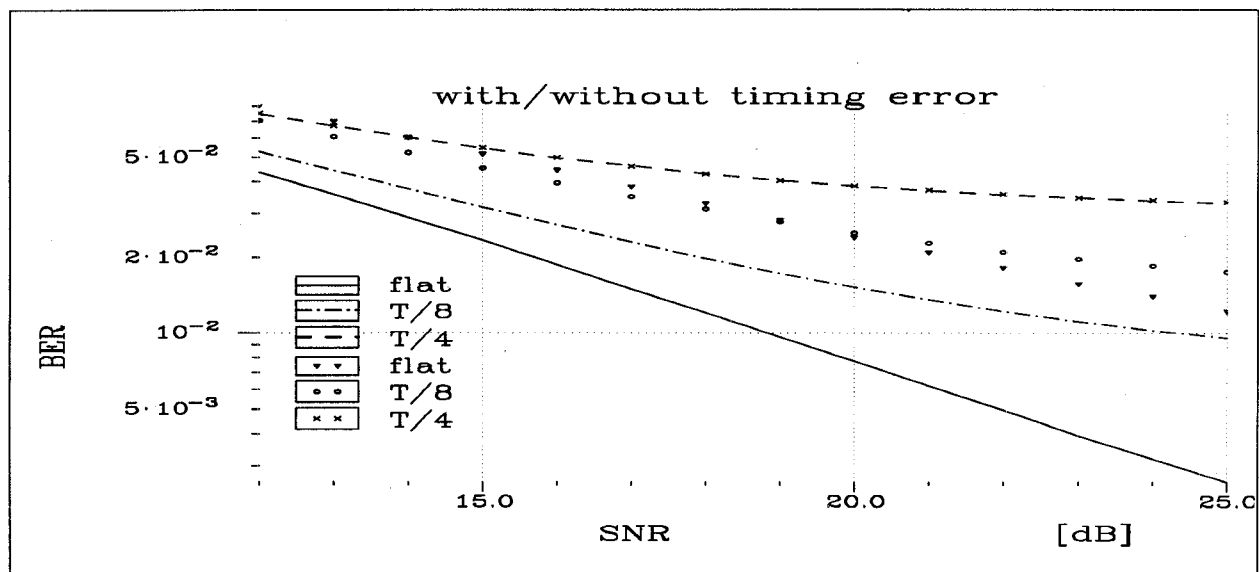
1560

Figure 2: *Differential detector on flat Rayleigh fading, T/8 and T/4 delay spread channels. Lines are for perfect timing, markers for T/4 timing error.*

except of the T delay spread (indicated by circle markers. The markers are slightly shifted away from the 20dB point to make them more discriminable). Thus, the problems caused by timing errors can be solved when a good estimate for the sync word location is available.

3.2 Sensitivity to Timing Errors

Figures 4 shows the results when applying an estimate for the sync word location. If compared the results to the static curves in Figure 3 there is hardly any loss in performance. Indicated by the markers is the performance for the four cases when a timing error of T/4 is introduced. Although a slight degradation for the T delay spread is observed, there is no performance loss for the flat and low Rayleigh fading scenarios.

3.3 Sensitivity to Frequency Offsets

Since the hardware implementation of the phone compensates frequency errors only up to 60Hz, it is important to know how this frequency error affects the algorithm. Since the case of flat fading is basically the worst case only this is considered during this section. A frequency shift of 40Hz worsens the results but they still satisfy the standards. However, 80Hz already causes a BER higher than 3% (see Figure 5). Thus, the LMS algorithm can only marginally satisfy

this impairment. Another algorithm, like Recursive-Least-Squares might perform better.

3.4 Sensitivity on CDVCC

Experiments with various CDVCC words showed that the Viterbi behavior can be very sensitive to the word. Repetitive patterns like in 0 (eight time zero), 170 (4 times pattern '10'), and 255 (eight times one) lead to poorer performance than others (up to three times worth for the all zero pattern). Figure 6 depicts the Viterbi performance for flat Rayleigh fading and 20dB SNR. Most of the CDVCC basically lead to the same performance. Only a few show poor behavior. The two worst cases occur for the CDVCC words 0 and 170. This can be explained by a rank deficiency. Since we try to identify a three tap system, we have to make sure that the excitation is persistent for this order. Given the seven symbols $s(1), ..., s(7)$ from the six transitions defined by the CDVCC, the matrix

$$\begin{pmatrix} s(1) & s(2) & s(3) & s(4) & s(5) \\ s(2) & s(3) & s(4) & s(5) & s(6) \\ s(3) & s(4) & s(5) & s(6) & s(7) \end{pmatrix}$$

can be constructed and its rank computed. It turns out that the rank is three except for CDVV=0 (rank=1) and CDVCC=170 (rank=2). Thus, even an RLS algorithm cannot deliver better results for these two words. For estimating four coefficients 22 CDVCC
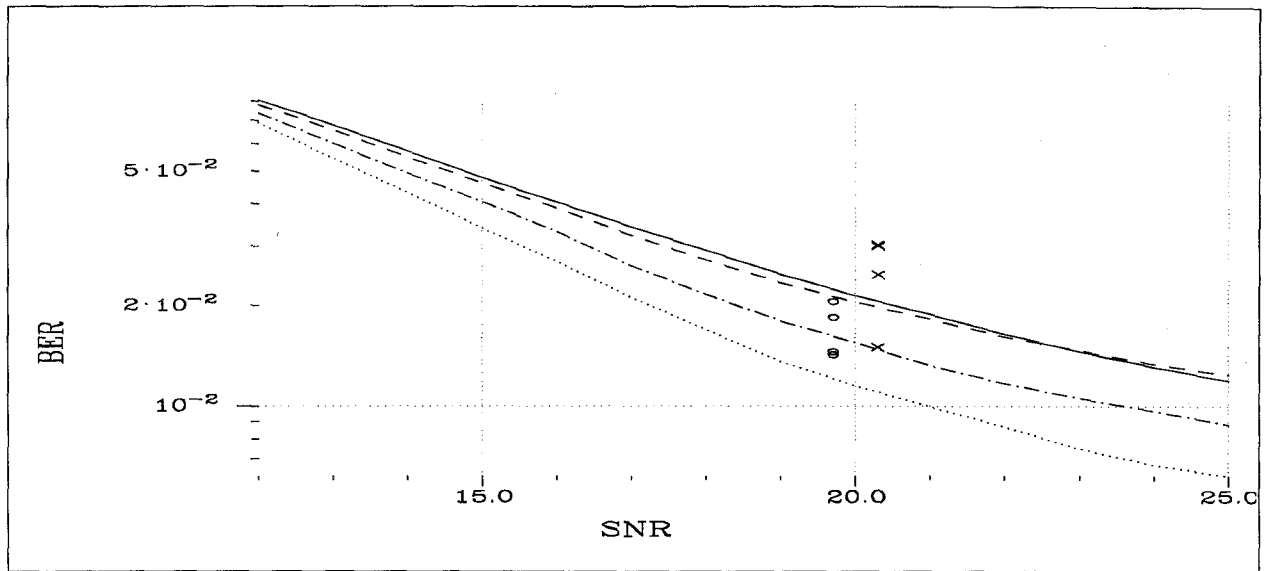
Figure 3: *Viterbi algorithm with perfect timing and knowledge of the sync word location. Lines are for perfect timing, markers for T/4 timing error.*

words would not have sufficient rank.

4 CONCLUSIONS

We presented a Viterbi based algorithm with LMS for channel tracking that proves to be robust against many kinds of impairments. It satisfies the requirements defined by the standards and offers some margin even in the worst case scenarios. The algorithm not only deals with corruptions during the transmission but can also deal with impairments introduced by the hardware realization of the handset like frequency offsets and timing errors. A DSP implementation seems possible with less than 10MIPS complexity.

References

[1] *TIA/EIA/Interim Standard PN 3386*, Telecommunication Association, Feb. 1995.

[2] John G. Proakis, *Digital Communications*, McGraw Hill 1989.

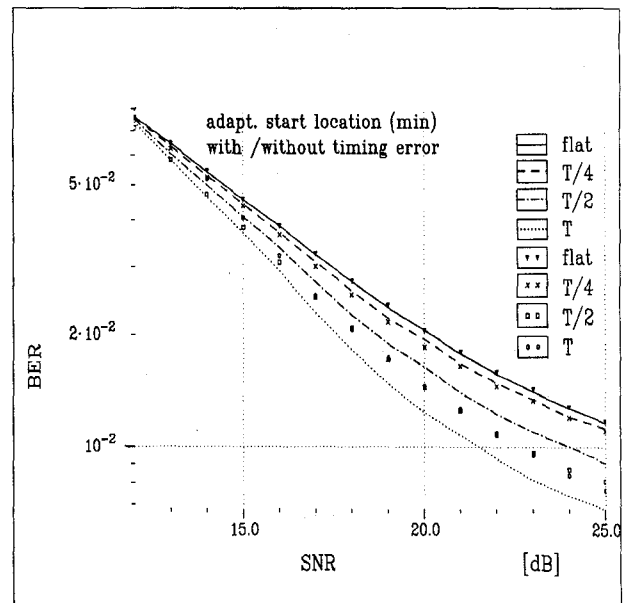[3] Simon Haykin, *Adaptive Filter Theory*, Prentice Hall 1991.

Figure 4: *Viterbi algorithm with sync location estimator. Continuous lines are for perfect timing, markers for T/4 timing error.*
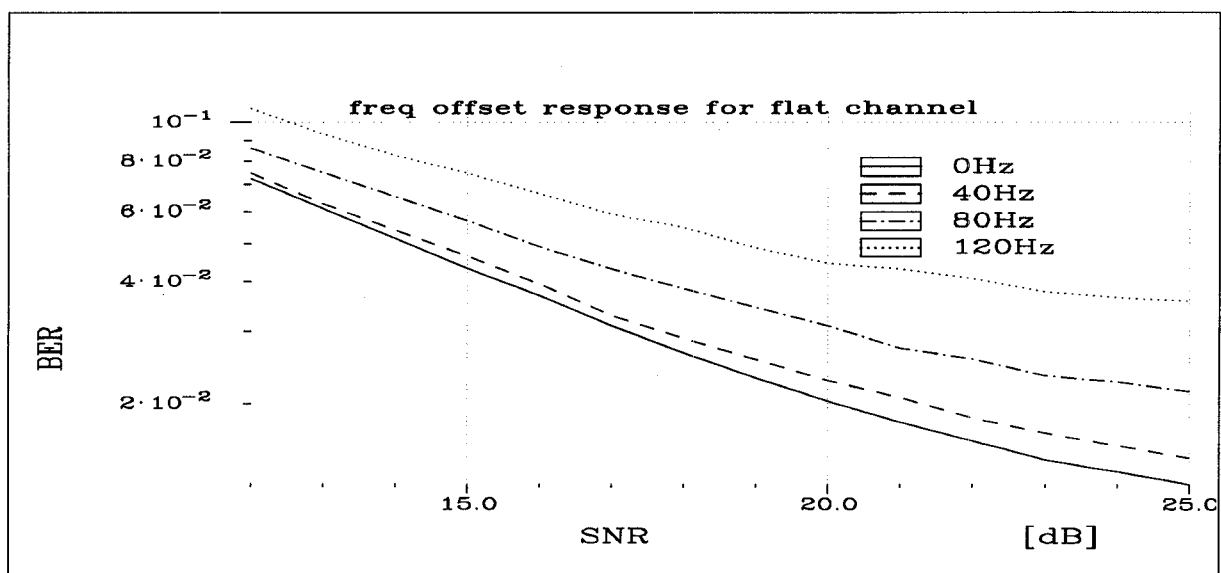
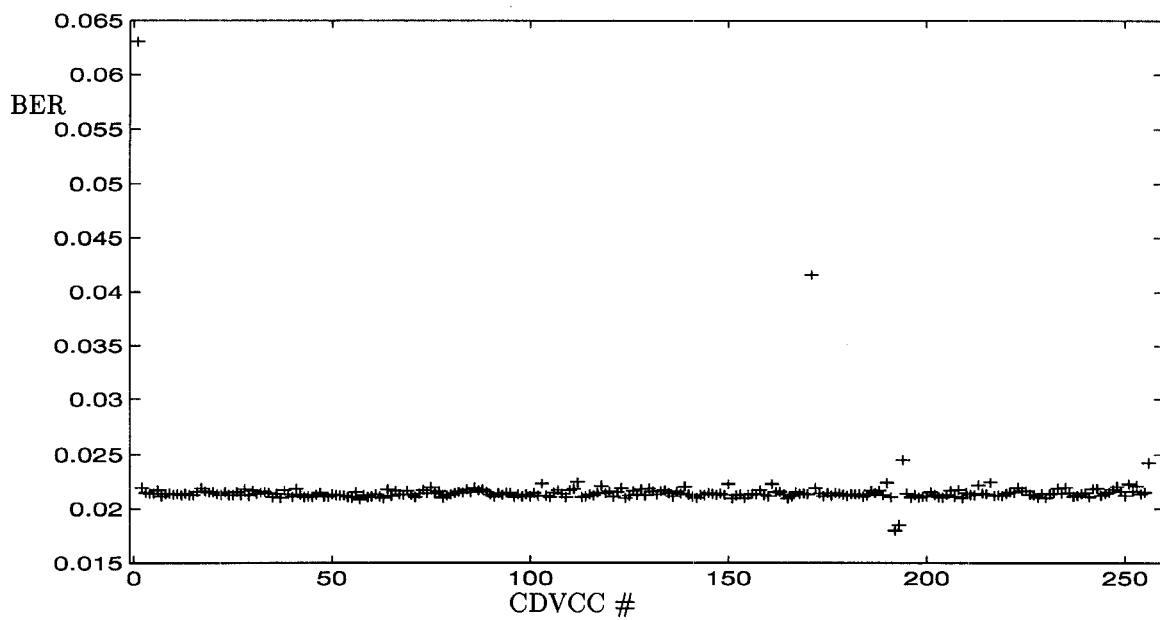Figure 5: *Viterbi algorithm under frequency offset.*

Figure 6: *Viterbi algorithm for flat Rayleigh fading and 20dB SNR as function of the CDVCC.*