DOWNLOADLINK WCDMA RECEIVERS BASED ON COMBINED CHIP AND SYMBOL LEVEL EQUALIZATION

Ahmet Baştuğ, Dirk T.M. Slock [0]

Eurecom Institute
2229 Route des Crêtes, B.P. 193, 06904, Sophia Antipolis, Cedex, FRANCE
ahmet.bastug@@eurecom.fr, dirk.slock@@eurecom.fr

ABSTRACT

In this paper we consider iterative techniques for multi-user detection. In the overdetermined case (in which signal and noise subspaces exist), polynomial expansion can perhaps most conveniently be applied to a (global) MMSE ZF linear receiver. In the resulting iteration sections, we then propose to replace the various receiver components by a (possibly non-linear) MMSE variant, based on the signal estimation functionality of each component. The resulting receiver sections, which can also be applied in the underdetermined case, have interpretations in terms of interference cancellation. We shall apply the proposed receiver strategy to the downlink in the FDD mode of UMTS, in which we consider a mobile terminal that also combats explicitly intercell interference.

1. BASEBAND DOWNLINK TRANSMISSION MODEL

The baseband downlink transmission model of a CDMA-based cellular system is given in Fig. 1. We consider only one interfering base station, BS 2, which, in practice, can be generalized to multiple base stations. For each base station $j, j = 1, 2$, the K^j linearly modulated user signals are transmitted over the same linear multipath channel $h^j(t)$, for we assume that downlink user signals are chip-synchronous and there is no beamforming. The symbol and chip periods T and T_c are related through the spreading factor L as T=LT_c, which here is assumed to be common for all the users and for the two base stations. The total chip sequences b^j_l are the sums of the chip sequences of all the users for each BS. Every user chip sequence is given by the product between the n^{th} symbol of the k^{th} user $a^j_{k,n}$ and an aperiodic spreading sequence $w^j_{k,l}$ which is itself the product of a periodic, unit energy Walsh-Hadamard spreading sequence $\boldsymbol{c}^j_k = \left[c^j_{k,0} c^j_{k,1} \cdots c^j_{k,L-1}\right]$, and a base-station specific unit magnitude complex scrambling sequence s^j_l with variance 1 as $w^j_{k,l} = c^j_{k,l \; mod \; L} s^j_l$:

$$b^j_l = \sum_{k=1}^{K^j} b^j_{k,l} = \sum_{k=1}^{K^j} a^j_{k,\lfloor \frac{l}{L} \rfloor} w^j_{k,l} \qquad j = 1, 2 \tag{1}$$

where $\lfloor x \rfloor$ is the largest integer which is less than or equal to x. The chip sequences $b^{1,2}_l$ get transformed into continuous-time signals by filtering them with the pulse shape $p(t)$ and passing them through the multipath propagation channels $\boldsymbol{h}^1(t)$ and $\boldsymbol{h}^2(t)$. These channels become multi-channels if there are more than one sensors at the mobile terminal or oversampling is applied. We consider that there are q sensors and the receiver samples the low-pass filtered received signal m times per chip period. As a result, the total number of samples per chip becomes mq. Stacking these mq samples in vectors, we get the sampled received vector signal

$$\boldsymbol{y}_l = \boldsymbol{y}^1_l + \boldsymbol{y}^2_l + \boldsymbol{v}_l, \quad \boldsymbol{y}^j_l = \sum_{k=1}^{K^j} \sum_{i=0}^{N-1} \boldsymbol{h}^j_i b^j_{k,l-i} \quad j = 1, 2 \tag{2}$$

where

$$\boldsymbol{y}^j_l = \begin{bmatrix} y^j_{1,l} \\ \vdots \\ y^j_{mq,l} \end{bmatrix}, \boldsymbol{h}^j_l = \begin{bmatrix} h^j_{1,l} \\ \vdots \\ h^j_{mq,l} \end{bmatrix}, \boldsymbol{v}_l = \begin{bmatrix} v_{1,l} \\ \vdots \\ v_{mq,l} \end{bmatrix}. \tag{3}$$

Here $\boldsymbol{h}^j_l$ $(mq \times 1)$ represents the vectorized samples (represented at chip rate) of the overall channel $h^j(t)$, including pulse shape, propagation channel and receiver filter. The overall channels $h^j(t)$ are assumed to have the same delay spread of N chips.

When the scrambling sequences are aperiodic and when we model the symbol sequences as i.i.d., stationary, white sequences, then since the chip sequences $b^{1,2}_l$ are sums of these sequences, they are also stationary and white. The intracell contribution to $\boldsymbol{y}_l$ then is a stationary vector process the continuous-time counterpart of which is cyclostationary with chip period. The intercell interference is a sum of contributions that are of the same form as the intracell contribution and therefore its contribution to $\boldsymbol{y}_l$ is also a stationary

[0] Contact author

vector process. The remaining noise is assumed to be white stationary noise. Hence the sum of intercell interference and noise, $\boldsymbol{v}_l$, represented at chip rate, is stationary (if the scrambler is considered unknown and hence i.i.d.).

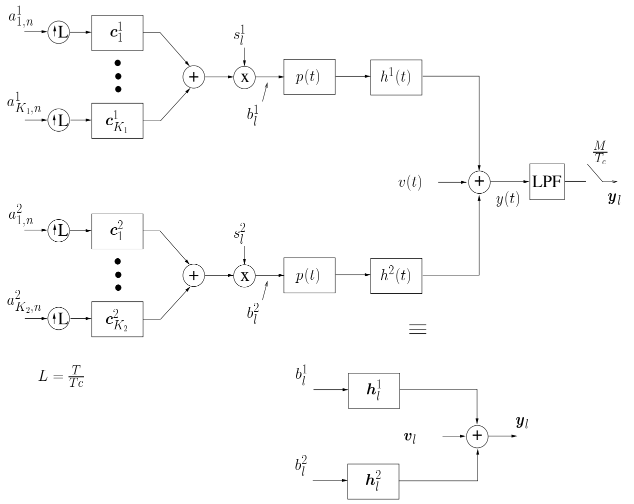

Figure 1: Baseband downlink transmission model

2. GLOBAL MMSE ZF POLYNOMIAL EXPANSION (PE) AND ITERATIVE INTERFERENCE CANCELLING (IC) STRUCTURES

In this section we develop intra- and intercell interference cancelling structures based on polynomial expansion (PE), see [1] for the introduction of PE. A vector of received signal over one symbol period n can be written as:

$$\begin{aligned} \boldsymbol{Y}[n] &= \boldsymbol{H}_1(z)\boldsymbol{S}_1[n]\,\boldsymbol{C}_1\boldsymbol{A}_1[n] \\ &+ \boldsymbol{H}_2(z)\boldsymbol{S}_2[n]\,\boldsymbol{C}_2\boldsymbol{A}_2[n] + \boldsymbol{V}[n] \end{aligned} \quad (4)$$

assuming w.l.o.g. a single interfering BS. As shown in Fig. 2, $\boldsymbol{H}_j(z) = \sum_{i=0}^{M_j-1} \boldsymbol{H}[i]\, z^{-i}$ is the $Lmq \times L$ channel transfer function at symbol rate (z^{-1} corresponds here to a symbol period delay operator). The block coefficients $\boldsymbol{H}_j[i]$ correspond to the $M_j = \lceil \frac{L+N_j+d_j-1}{L} \rceil$ parts of the block Toeplitz matrix (with $mq \times 1$ sized blocks) with $\boldsymbol{h}^j$ as first column. d_j is the TX delay between BS j and the mobile, expressed in chip periods. $\boldsymbol{h}^j$ comprises the delay in the channel and hence its top entries may be zero. The $L \times L$ matrices $\boldsymbol{S}_j[n]$ are diagonal and contain the scrambler of BS j for symbol period n.

The vectors $\boldsymbol{A}_j$ contain the K_j symbols of BS j for symbol period n, and $\boldsymbol{C}_j$ is the $L \times K_j$ matrix of the K_j active codes for BS j. Symbol and chip rate equivalents of channels and scramblers are presented in Fig. 3. One can pass between symbol and chip rate representations via

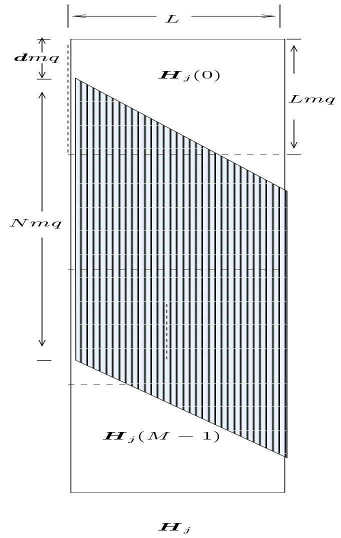

Figure 2: Channel impulse response of $\boldsymbol{H}(z)$.

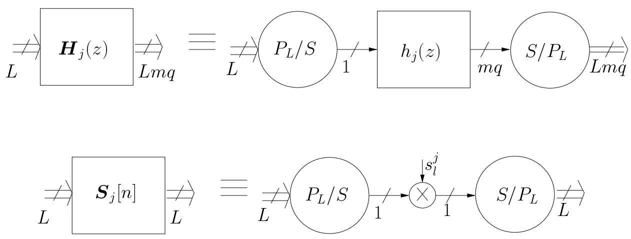

Figure 3: Symbol rate and chip rate equivalent representations of channel filtering and scrambling.

serial-to-parallel and parallel-to-serial converters by vectoring and sample rate conversion by a factor L. With these equivalents and similar ones, one can go from the chip rate representation in Fig. 1 to the symbol rate model captured by (4). Equation (4) can be rewritten as

$$\begin{aligned} \boldsymbol{Y}[n] &= [\boldsymbol{H}_1(z)\boldsymbol{S}_1[n]\boldsymbol{C}_1 \;\; \boldsymbol{H}_2(z)\boldsymbol{S}_2[n]\boldsymbol{C}_2]\boldsymbol{A}[n] + \boldsymbol{V}[n] \quad (5) \\ &= \widetilde{\boldsymbol{G}}(n,z)\boldsymbol{A}[n] + \boldsymbol{V}[n] \quad (6) \end{aligned}$$

where $\boldsymbol{A}$ is the $(K = K_1 + K_2) \times 1$ vector containing all the transmitted symbols for the two BS at symbol period n. $\widetilde{\boldsymbol{G}}(n,z)$ is a $Lmq \times K$ channel-plus-spreading symbol rate filter, and is time-varying due to the scrambling. The condition $K < Lq$ (or even $K < Lmq$) guarantees the existense of an FIR left inverse filter for $\widetilde{\boldsymbol{G}}(n,z)$, and allows zero-forcing interference cancellation. The calculation of an exact inverse of the channel would lead to high complexity due to the time-varying nature of the scrambler. Hence we will provide a good approximation to this inverse with acceptable complexity based on polynomial expansion [1].

Instead of basing the receiver directly on the received signal, we shall first introduce a dimensionality reduction (from Lmq to K) step (generating a minimal set of sufficient statistics), by equalizing the channels with (zero delay) MMSE ZF (zero forcing) chip rate equalizers $\boldsymbol{F}_j(z)$ followed by a bank of correlators. The sense of the MMSE criterion in the MMSE ZF equalizers corresponds to considering both noise and intercell interference as noise (this will reduce the response of an equalizer to the channel of another BS). The ZF character of the equalizers reduces the cascade of an equalizer and its corresponding channel to an identity operation. Let $\boldsymbol{X}[n]$ be the $K \times 1$ correlator output (which would correspond to the RAKE outputs if chanel matched filters instead of channel equalizers were used), then:

$$\begin{aligned} \boldsymbol{X}[n] &= \widetilde{\boldsymbol{F}}(n,z)\boldsymbol{Y}[n] && (7)\\ &= \begin{bmatrix} \boldsymbol{C}_1^H \boldsymbol{S}_1^H[n]\boldsymbol{F}_1(z) \\ \boldsymbol{C}_2^H \boldsymbol{S}_2^H[n]\boldsymbol{F}_2(z) \end{bmatrix} (\widetilde{\boldsymbol{G}}(n,z)\boldsymbol{A}[n] + \boldsymbol{V}[n]) && (8)\\ &= \boldsymbol{M}(n,z)\boldsymbol{A}[n] + \widetilde{\boldsymbol{F}}(n,z)\boldsymbol{V}[n] && (9) \end{aligned}$$

where $\boldsymbol{M}(n,z) = \widetilde{\boldsymbol{F}}(n,z)\widetilde{\boldsymbol{G}}(n,z)$. ZF equalization results in
$\boldsymbol{F}_j(z)\boldsymbol{H}_j(z) = I$. Hence, $\boldsymbol{M}(n,z) = \sum_i \boldsymbol{M}[n,i]z^{-1} = \begin{bmatrix} I & * \\ * & I \end{bmatrix}$ due to proper normalization of the code energies. In order to obtain the estimate of $\boldsymbol{A}[n]$, we initially consider the processing of $\boldsymbol{X}[n]$ by a decorrelator (which would lead to a global symbol rate MMSE ZF linear RX if the chip rate MMSE ZF equalizers $\boldsymbol{F}_j(z)$ would have ignored the intercell interference in the MMSE consideration):

$$\begin{aligned} \widehat{\boldsymbol{A}}[n] &= \boldsymbol{M}(n,z)^{-1}\boldsymbol{X}[n] && (10)\\ &= (\boldsymbol{I} + \overline{\boldsymbol{M}}(n,z))^{-1}\boldsymbol{X}[n] && (11) \end{aligned}$$

The correlation matrix $\boldsymbol{M}(n,z)$ has a coefficient $\boldsymbol{M}[n,0]$ with a dominant unit diagonal in the sense that all other elements of the $\boldsymbol{M}[n,i]$ are much smaller than one in magnitude. Hence, the polynomial expansion approach suggests to develop $(\boldsymbol{I} + \overline{\boldsymbol{M}}(n,z))^{-1} = \sum_{i=0}^{\infty}(-\overline{\boldsymbol{M}}(n,z))^i$ up to some finite order, which leads to

$$\widehat{\boldsymbol{A}}^{(-1)}[n] = 0 \qquad (12)$$

$$i \geq 0 \quad \widehat{\boldsymbol{A}}^{(i)}[n] = \boldsymbol{X}[n] - \overline{\boldsymbol{M}}(n,z)\,\widehat{\boldsymbol{A}}^{(i-1)}[n] \ . \qquad (13)$$

In practice, we may want to stop at the first-order expansion, the quality of which depends on the degree of dominance of the diagonal of the static part of $\boldsymbol{M}(n,z)$ with respect to its off-diagonal elements and the dynamic part.

At first order, the expression for the user of interest (user one) becomes:

$$\begin{aligned} &\widehat{\boldsymbol{a}}_1^1[n] = \boldsymbol{e}_1^H\widehat{\boldsymbol{A}}^{(1)}[n] = \boldsymbol{e}_1^H(\boldsymbol{X}[n] - \overline{\boldsymbol{M}}(n,z)\,\widehat{\boldsymbol{A}}^{(0)}[n])\\ &= \boldsymbol{e}_1^H(2\boldsymbol{X}[n] - \boldsymbol{M}(n,z)X[n])\\ &= \boldsymbol{e}_1^H\widetilde{\boldsymbol{F}}(n,z)(2\boldsymbol{Y}[n] - \widetilde{\boldsymbol{G}}(n,z)X[n])\\ &= \boldsymbol{c}_1^{1\,H}\boldsymbol{S}_1^H[n]\boldsymbol{F}_1(z)\,(2\boldsymbol{Y}[n] - \widetilde{\boldsymbol{G}}(n,z)\widetilde{\boldsymbol{F}}(n,z)\boldsymbol{Y}[n]) \end{aligned} \qquad (14)$$

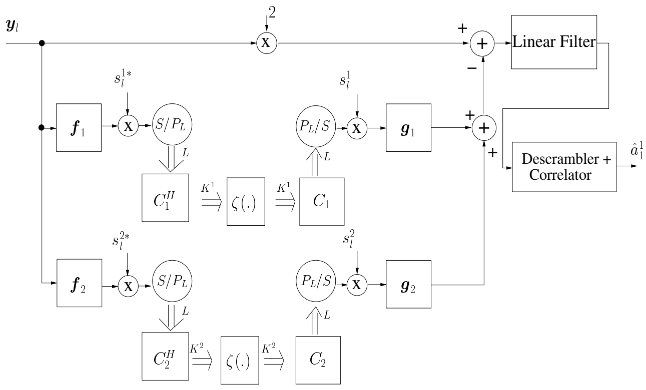

Figure 4: Symbol Rate Polynomial Expansion Structure.

From this last symbol rate equation one can obtain the chiprate signal processing diagram in Fig. 4 by using the symbol and chiprate equivalences in Fig. 3 and similar ones. Each branch in the IC block is formed by the cascade of a linear filter, a descrambler, a despreader, optionally a nonlinear operation (here in first instance an identity for the used code outputs), a respreader, a scrambler and a re-channeling filter.

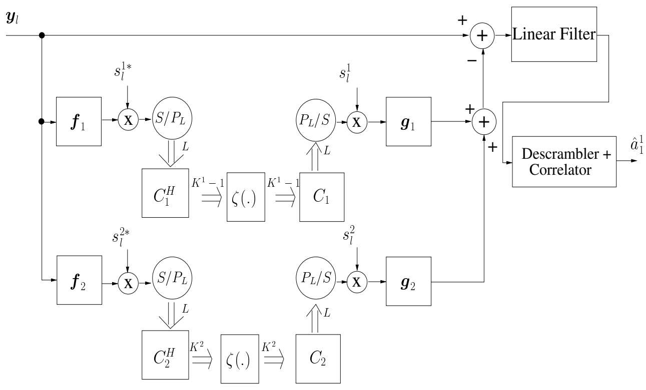

Figure 5: Interference canceller structure

Fig. 5 shows a modified form of the first-order polynomial expansion receiver structure which now has the form of an interference canceller. In this case C_1, C_1^H correspond in fact to (de)spreading with all intracell interferer's codes. So the top interference canceling branch corresponds to all intracell interference whereas the bottom branch corresponds to all users of an interfering base station. Let $I_{K\times K} = [\boldsymbol{e}_1\,\overline{\boldsymbol{e}}_1]$. Then the structure of Fig. 5 can be shown to be equivalent to that of Fig. 4. Indeed, from (14),

$$\begin{aligned} &\widehat{\boldsymbol{a}}_1^1[n] = \boldsymbol{e}_1^H\widetilde{\boldsymbol{F}}(n,z)(2\boldsymbol{Y}[n] - \widetilde{\boldsymbol{G}}(n,z)\widetilde{\boldsymbol{F}}(n,z)\boldsymbol{Y}[n]) =\\ &\boldsymbol{e}_1^H\widetilde{\boldsymbol{F}}(n,z)(2\boldsymbol{Y}[n] - \widetilde{\boldsymbol{G}}(n,z)(\boldsymbol{e}_1\boldsymbol{e}_1^H + \overline{\boldsymbol{e}}_1\overline{\boldsymbol{e}}_1^H)\widetilde{\boldsymbol{F}}(n,z)\boldsymbol{Y}[n])\\ &= \boldsymbol{e}_1^H\widetilde{\boldsymbol{F}}(n,z)(\boldsymbol{Y}[n] - \widetilde{\boldsymbol{G}}(n,z)\overline{\boldsymbol{e}}_1\overline{\boldsymbol{e}}_1^H\widetilde{\boldsymbol{F}}(n,z)\boldsymbol{Y}[n]) \end{aligned} \qquad (15)$$

since $\boldsymbol{e}_1^H \widetilde{\boldsymbol{F}}(n,z)\widetilde{\boldsymbol{G}}(n,z)\boldsymbol{e}_1 = 1$. Let $\boldsymbol{a}_1[n] = \boldsymbol{e}_1^H \boldsymbol{A}[n]$, $\overline{\boldsymbol{a}}_1[n] = \overline{\boldsymbol{e}}_1^H \boldsymbol{A}[n]$. From (13), we can obtain higher-order interference cancelling iterations:

$$
\begin{aligned}
&\widehat{\boldsymbol{a}}_1^{(-1)}[n] = 0,\ \widehat{\overline{\boldsymbol{a}}}_1^{(-1)}[n] = 0,\ i \geq 0\ :\\
&\widehat{\overline{\boldsymbol{a}}}_1^{(i-1)}[n] =\\
&\widehat{\overline{\boldsymbol{a}}}_1^{(i-2)}[n] + \overline{\boldsymbol{e}}_1^H \widetilde{\boldsymbol{F}}(n,z)\left(\boldsymbol{Y}[n] - \widetilde{\boldsymbol{G}}(n,z)\widehat{\boldsymbol{A}}^{(i-2)}[n]\right)\\
&\widehat{\boldsymbol{a}}_1^{(i)}[n] = \boldsymbol{e}_1^H \widetilde{\boldsymbol{F}}(n,z)\left(\boldsymbol{Y}[n] - \widetilde{\boldsymbol{G}}(n,z)\overline{\boldsymbol{e}}_1\widehat{\overline{\boldsymbol{a}}}^{(i-1)}[n]\right)\\
&= \widehat{\boldsymbol{a}}_1^{(i-1)}[n] - \boldsymbol{e}_1^H \widetilde{\boldsymbol{F}}(n,z)\widetilde{\boldsymbol{G}}(n,z)\overline{\boldsymbol{e}}_1\left(\widehat{\overline{\boldsymbol{a}}}^{(i-1)}[n] - \widehat{\overline{\boldsymbol{a}}}^{(i-2)}[n]\right)
\end{aligned}
\tag{16}
$$

3. LOCAL MMSE OPERATIONS

One of the advantages of a MMSE ZF approach as opposed to a MMSE approach is that clear symbol and chip sequence estimates appear at various points in the receiver structure. These estimates can now be improved locally by replacing whatever the global MMSE ZF structure yields as estimates by improved estimates in the MMSE sense. Any local MMSE improvements should lead to global MMSE improvement. In an iterative PE approach, such modifications should also lead to smaller off-diagonal power and hence faster convergence of the iterations, to an estimate that is closer to a MMSE estimate.

The interpretations to be discussed apply to any iteration. So we can concentrate on the first iteration displayed in Fig. 5. Of course, the values of a number of quantities to be discussed may depend on the iteration index. In Fig. 5, the role of filters $\boldsymbol{f}_1$ and $\boldsymbol{f}_2$ is to produce estimates of the chip sequences of BS one and two. Those estimates can be improved by replacing the MMSE ZF chip rate equalizers $\boldsymbol{f}_1$ and $\boldsymbol{f}_2$ considered so far by (unbiased) MMSE equalizers. The estimated chip sequence then gets descrambled and passes by correlators to produce symbol estimates for the intracell and intercell interferers. These symbol estimates can be improved in a variety of ways. This will be done by symbol-wise linear or non-linear functions $\zeta(.)$. In a first instance, one can exploit the symbol variance to introduce a LMMSE weighting factor. In a second instance, one may want to exploit the symbol constellation. This could be done quite simply by taking hard decisions. However, that may not be optimal or not even represent an improvement. One may replace the hard decision by a variety of soft decisions. A locally MMSE estimate is obtained by using a hyperbolic tangent function as shown in Fig. 6. The estimated symbols are then respread, scrambled and added to produce again an estimate of the chip sequence. The purpose of the rechanneling filters $\boldsymbol{g}_j$ is to produce an (intracell or intercell) interference estimate at the level of the received signal, on the basis o the chip sequences estimates. To minimize the MSE of this interference cancelling step, one may replace the channel impulse response $\boldsymbol{h}_j$ choices for $\boldsymbol{g}_j$ by Wiener filters.

For the spreading and despreading with the Walsh-Hadamard codes of the actives users, it is suggested to (de)spread with all WH codes simutaneously, which can be done with the Fast Walsh-Hadamard transform (FWHT). The unused code correlator outputs can be left unused (the detection of used vs. unused codes occurs implicitely when the powers associated with the codes get estimated).

So far we have considered a unique spreading factor. The multirate case corresponding to variable spreading factors can be reformulated as a multicode case. The tree structure of the orthogonal variable spreading factors (OVSFs) allows to interpret the interfering users as equivalent low rate users (at least if the symbol constellation of the interferers will not get exploited, in which case only second-order statistics count). Hence in that case one needs at most to consider the multi-code case at low rate for the user of interest. If on the other hand one also desires to exploit the interferers' symbol constellation, then one needs to detect and use the actual spreading factors of the interferers. The tree structure of the OVSFs can be exploited to progressively explore increasing spreading factors and investigate the finite alphabet hypothesis at the various tree branches. In this case one cannot use the FWHT as such but the ingredients leading to its derivation (OVSF in fact) can still be used for fast (de) spreading.

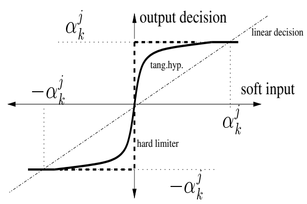

Figure 6: Linear, tangent hyperbolic and hard limiter decisions

3.1. Estimation of Interfering User Powers

The implementation of the LMMSE weighting factors or the hyperbolic tangent requires the estimation of the powers of the various users. Let $\boldsymbol{z}_{i,j}$ represent the channel-MMSE filter cascade of h_j and f_i. Then the expected value of the received power for interfering user k on the first branch after despreading is equal to

$$
\beta_k^1 = (a_k^1)^2 + \frac{1}{L}(\|\boldsymbol{z}_{1,1}\|^2 - z_{1,1}(d))\sum_{i=1}^{K^1}(a_i^1)^2
$$

$$
+\frac{1}{L}\|\boldsymbol{z}_{1,2}\|^2\sum_{i=1}^{K^2}(a_i^2)^2 + \sigma_v^2\|f_1\|^2, \quad d : equalizer\ delay \tag{17}
$$

since on the average the correlation coefficient between any two nonorthogonal codes or between two shifted versions of the same code has variance $\frac{1}{L}$. We assume that we know the channels, the total received power from both base stations (which can also be estimated) and noise variance but we do not know each user's power. Therefore the estimate of $(a_k^1)^2$, namely $(\alpha_k^1)^2$, can be calculated by subtracting second, third and fourth terms on the right hand side of (17) from long term average power. The powers of users in the second cell are estimated similarly.

3.2. Filtering Alternatives

As already mentioned, there are various filtering alternatives. First of all, the filters on the branches can either be MMSE-ZF equalizers or MMSE filters. The former enhances the intercell interference plus noise alot but preserves the code orthogonality, whereas the latter perturbs the structure of the signal received from the base station but does not increase so much the intercell interference plus noise. Note that although it is not possible to completely equalize a single FIR channel by an FIR equalizer, it is possible with multichannels [2]. In our case, multichannels exist due to oversampling. Secondly, the linear filter after the subtractor can either be a RAKE receiver or again an MMSE filter.

4. SIMULATIONS AND CONCLUSIONS

In this section we present some of the simulations performed to evaluate the various structures. The K^j users of base station j are considered synchronous between them, with the same spreading factor $L = 32$ and using the same downlink channel $\boldsymbol{h}^j$ which is a FIR filter, convolution of a sparse Vehicular A UMTS channel and a root-raised cosine pulse shape with roll-off factor of 0.22. The channel length is $N = 19$ chips due to the UMTS chip rate of 3.84 Mchips/sec. An oversampling factor of $m = 2$ and one receive antenna $q = 1$ are used. User symbols are from QPSK constellation. The user of interest has 10dB less power than average single interferer power which represents the near-far situation. Fig. 7 and Fig. 8 show the SINR performance of the proposed receivers and the classical reference Rake receiver. As seen, there is no receiver which performs the best at all Eb/No values. Rake receiver performs the best below a certain Eb/No threshold. Hard limiter and tangent hyperbolic nonlinearities beat projection and are better alternatives to Rake at moderate to high Eb/No values. Between the two, hard limiter is a reliable choice for low loading fractions but is outperformed by tangent hyperbolic nonlinearity at higher loading fractions. For the linear filter after the subtractor, MMSE filter is better than Rake. In brief, proposed receivers are good alternatives to Rake receiver with reasonable complexity.

5. REFERENCES

[1] E. Kanterakis S. Moshavi and D. L. Schilling, "Multistage linear receivers for ds-cdma systems," *International Journal of Wireless Information Networks, Vol.3, No.1*, 1996.

[2] C. Papadias and D.T.M. Slock, "Fractionally spaced equalization of linear polyphase channels and related blind techniques based on multichannel linear prediction," *IEEE Transactions on Signal Processing, Vol.47, No.3*, March 1999.

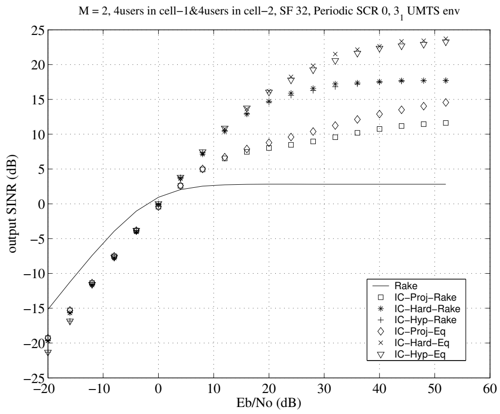

Figure 7: Output SINR vs Eb/No, 12.5% loaded BSs, near-far situation, aperiodic scrambling, MMSE filters.

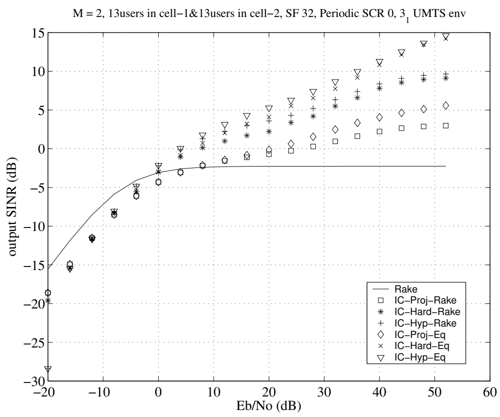

Figure 8: Output SINR vs Eb/No, 40% loaded BSs, near-far situation, aperiodic scrambling, MMSE filters.