TSG-RAN Working Group 1 **TSGR1#6(99)782**
July 13-16, 1999

Source: Texas Instruments

Title: Further results on channel estimation for TDD using pilot symbols

1.0 SUMMARY

The 8 special midambles listed in [1] have a basic length of 192 chips, and each user in a cell uses the same midamble with a different offset. After the midambles for all the users are summed, a cyclic prefix is inserted to increase the midamble length to 256 chips. The midambles were designed to allow the receiver to perform channel estimates on all the users with low complexity [2]. Instead of the special midambles, we propose that channel estimation be performed using pilot symbol sequences based on maximal length sequences. The two techniques are compared in this paper with the following results:

1) The performance of the two techniques was found to be similar, with the midambles performing better by 0.1 to 0.4 dB, depending on the channel model. They were compared both in raw BER and using joint detectors (the zero-forcing block linear equalizer (ZF-BLE) and the parallel interference canceller (PIC)).

2) ***Because the same basic sequence is repeated 8 times, the special midambles are not robust to intercell interference***. If a user receives signals from two base stations with equal power, then we find that more than 50% of the time the cross correlation with the midamble of the interfering base station will cause a false path to appear that is only 3 to 6 dB down from the strongest path of the home base station. The simulations corresponding to (1) above do not take the intercell interference into account. Taking intercell interference into account, the performance of midamble based channel estimation will degrade compared to the pilot symbol based channel estimation.

3) With the special midamble, if the number of users is 8 then the maximum length of the search window is limited to 24 chips (6 μs). ***With the pilot symbol sequences up to 16 users can be supported with no constraint on the length of the search window.***

4) The special midamble approach is not well suited to provide channel estimation to users having different delay spreads. The spacing between the repeated basic sequences must be designed to handle the largest possible delay spread, which will limit the number of users that can be supported in an outdoor environment. ***The pilot symbol approach does not have this limitation and can support users with mixed delay spreads***.

5) The complexity of channel estimation (including channel estimates at every chip position) is reduced by about 50% in most scenarios using the pilot symbol approach.

Thus, since the pilot symbol based channel estimation has similar performance, no constraints on the delay spread and reduced complexity as compared to the midamble based channel estimation, we propose that channel estimation be performed using pilot symbols.

2.0 PILOT SYMBOL SEQUENCES

Maximal-length sequences (m-sequences) are known to have good autocorrelation properties [3]. There are two m-sequences of length 15 which are

[1 1 1 1 –1 –1 –1 1 –1 –1 1 1 –1 1 –1] and [–1 1 –1 1 1 –1 –1 1 –1 –1 –1 1 1 1 1]

Note that the second sequence is the first sequence in reverse order. In order to form sequences of length 16, we simply append a 1 to the end of the first sequence to get

Sequence 1 = [1 1 1 1 –1 –1 –1 1 –1 –1 1 1 –1 1 –1 1] and
Sequence 2 = [1 –1 1 –1 1 1 –1 –1 1 –1 –1 –1 1 1 1 1]

There are 16 circular shifts of each sequence, so there are a total of 32 sequences of length 16. These sequences are numbered s0 to s15 for circular shifts of Sequence 1 and s16 to s31 for circular shifts of Sequence 2 and are given in Tables 1(a) and 1(b). Each pilot symbol sequence is then generated by multiplying (1+j) times one of these sequences.

Table 1(a): 16 pilot symbol sequences generated by left circular shifts of Sequence 1.

	Pilot Symbol Number															
	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
s0	1	1	1	1	-1	-1	-1	1	-1	-1	1	1	-1	1	-1	1
s1	1	1	1	-1	-1	-1	1	-1	-1	1	1	-1	1	-1	1	1
s2	1	1	-1	-1	-1	1	-1	-1	1	1	-1	1	-1	1	1	1
s3	1	-1	-1	-1	1	-1	-1	1	1	-1	1	-1	1	1	1	1
s4	-1	-1	-1	1	-1	-1	1	1	-1	1	-1	1	1	1	1	1
s5	-1	-1	1	-1	-1	1	1	-1	1	-1	1	1	1	1	1	-1
s6	-1	1	-1	-1	1	1	-1	1	-1	1	1	1	1	1	-1	-1
s7	1	-1	-1	1	1	-1	1	-1	1	1	1	1	1	-1	-1	-1
s8	-1	-1	1	1	-1	1	-1	1	1	1	1	1	-1	-1	-1	1
s9	-1	1	1	-1	1	-1	1	1	1	1	1	-1	-1	-1	1	-1
s10	1	1	-1	1	-1	1	1	1	1	1	-1	-1	-1	1	-1	-1
s11	1	-1	1	-1	1	1	1	1	1	-1	-1	-1	1	-1	-1	1
s12	-1	1	-1	1	1	1	1	1	-1	-1	-1	1	-1	-1	1	1
s13	1	-1	1	1	1	1	1	-1	-1	-1	1	-1	-1	1	1	-1
s14	-1	1	1	1	1	1	-1	-1	-1	1	-1	-1	1	1	-1	1
s15	1	1	1	1	1	-1	-1	-1	1	-1	-1	1	1	-1	1	-1

Table 1(b): 16 pilot symbol sequences generated by left circular shifts of Sequence 2.

	Pilot Symbol Number															
	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
s16	1	-1	1	-1	1	1	-1	-1	1	-1	-1	-1	1	1	1	1
s17	-1	1	-1	1	1	-1	-1	1	-1	-1	-1	1	1	1	1	1
s18	1	-1	1	1	-1	-1	1	-1	-1	-1	1	1	1	1	1	-1
s19	-1	1	1	-1	-1	1	-1	-1	-1	1	1	1	1	1	-1	1
s20	1	1	-1	-1	1	-1	-1	-1	1	1	1	1	1	-1	1	-1
s21	1	-1	-1	1	-1	-1	-1	1	1	1	1	1	-1	1	-1	1
s22	-1	-1	1	-1	-1	-1	1	1	1	1	1	-1	1	-1	1	1
s23	-1	1	-1	-1	-1	1	1	1	1	1	-1	1	-1	1	1	-1
s24	1	-1	-1	-1	1	1	1	1	1	-1	1	-1	1	1	-1	-1
s25	-1	-1	-1	1	1	1	1	1	-1	1	-1	1	1	-1	-1	1
s26	-1	-1	1	1	1	1	1	-1	1	-1	1	1	-1	-1	1	-1
s27	-1	1	1	1	1	1	-1	1	-1	1	1	-1	-1	1	-1	-1
s28	1	1	1	1	1	-1	1	-1	1	1	-1	-1	1	-1	-1	-1
s29	1	1	1	1	-1	1	-1	1	1	-1	-1	1	-1	-1	-1	1
s30	1	1	1	-1	1	-1	1	1	-1	-1	1	-1	-1	-1	1	1
s31	1	1	-1	1	-1	1	1	-1	-1	1	-1	-1	-1	1	1	1

We assign different pilot symbol sequences to each user in order to minimize crosscorrelations between users. Instead of Walsh codes, we use chip sequences based on Gold –like sequences to minimize cross correlations [4].

Table 2: 16 chip sequences based on Gold-like sequences.

	Chip number															
Signature	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
0	1	1	1	-1	-1	-1	1	-1	-1	1	1	-1	1	-1	1	1
1	-1	1	-1	-1	1	1	1	-1	1	1	1	-1	-1	1	-1	1
2	1	-1	1	1	1	-1	1	1	-1	1	1	1	-1	1	-1	1
3	-1	1	-1	1	-1	-1	-1	-1	-1	1	-1	1	-1	1	1	1
4	1	-1	-1	-1	-1	1	1	-1	-1	-1	-1	1	-1	-1	-1	1
5	-1	-1	1	-1	1	-1	1	-1	1	-1	-1	1	1	1	1	1
6	-1	1	1	1	-1	-1	1	1	1	-1	-1	-1	-1	-1	-1	1
7	1	1	-1	-1	-1	-1	-1	1	1	-1	1	1	1	1	-1	1
8	1	-1	1	-1	-1	1	-1	1	1	1	-1	-1	-1	1	1	1
9	-1	1	1	-1	1	1	-1	1	-1	-1	1	1	-1	-1	1	1
10	1	1	1	1	1	1	-1	-1	1	1	-1	1	1	-1	-1	1
11	1	1	-1	1	1	1	1	1	-1	-1	-1	-1	1	1	1	1
12	1	-1	-1	1	1	-1	-1	-1	1	-1	1	-1	-1	-1	1	1
13	-1	-1	-1	1	-1	1	1	1	1	1	1	1	1	-1	1	1
14	-1	-1	-1	-1	1	-1	-1	1	-1	1	-1	-1	1	-1	-1	1
15	-1	-1	1	1	-1	1	-1	-1	-1	-1	1	-1	1	1	-1	1

The pilot sequences modulate the combination of the Gold-like sequence for that user and the scrambling code for the cell as in Figure 1. The users can be numbered according to their Walsh code as they are listed in the standard order in Table 3. If a user is assigned multiple Walsh codes, the number for the first Walsh code can be used, and if Walsh codes higher in the code tree are used (Walsh codes of length 4 or 8), then the number for the first leaf than corresponds to that node can be used. The Gold-like sequences are rotated between the users to further randomize the interference. If the frame number is equal to m mod 16, then user number k will use Gold sequence number (k+m) mod 16.

Table 3: Walsh codes for a spreading factor of 16.

User Number	Walsh Code
0	1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
1	1 1 1 1 1 1 1 1 -1 -1 -1 -1 -1 -1 -1 -1
2	1 1 1 1 -1 -1 -1 -1 1 1 1 1 -1 -1 -1 -1
3	1 1 1 1 -1 -1 -1 -1 -1 -1 -1 -1 1 1 1 1
4	1 1 -1 -1 1 1 -1 -1 1 1 -1 -1 1 1 -1 -1
5	1 1 -1 -1 1 1 -1 -1 -1 -1 1 1 -1 -1 1 1
6	1 1 -1 -1 -1 -1 1 1 1 1 -1 -1 -1 -1 1 1
7	1 1 -1 -1 -1 -1 1 1 -1 -1 1 1 1 1 -1 -1
8	1 -1 1 -1 1 -1 1 -1 1 -1 1 -1 1 -1 1 -1
9	1 -1 1 -1 1 -1 1 -1 -1 1 -1 1 -1 1 -1 1
10	1 -1 1 -1 -1 1 -1 1 1 -1 1 -1 -1 1 -1 1
11	1 -1 1 -1 -1 1 -1 1 -1 1 -1 1 1 -1 1 -1
12	1 -1 -1 1 1 -1 -1 1 1 -1 -1 1 1 -1 -1 1
13	1 -1 -1 1 1 -1 -1 1 -1 1 1 -1 -1 1 1 -1
14	1 -1 -1 1 -1 1 1 -1 1 -1 -1 1 -1 1 1 -1
15	1 -1 -1 1 -1 1 1 -1 -1 1 1 -1 1 -1 -1 1

The pilot symbol sequences are assigned according to the user number and the base station number. The sequences are varied from cell to cell in order to reduce the effect of intercell interference. There are 128 scrambling codes given in [5], so if we assign the base station number in a one-to-one correspondence with the scrambling code number, then there is no additional network planning needed to assign the pilot sequences. There can be up to 16 users with a spreading factor of 16 in a cell, so sets of 16 pilot sequences can be assigned to each cell. Table 4 shows the 128 sets of pilot sequences in abbreviated form. There are 8 basic sets of pilot symbols, and each set has 16 circular shifts, so there are a total of 128 sets of pilot symbols. Set numbers 0 to 15 are generated by left circular shifts of the set for BTS 0. Similarly, the sets for BTS 16, 32, 48, 64, 80, 96, and 112 also have 16 circular shifts each, but these are not shown in the table due to lack of space.

Table 4: Pilot symbol sequences used for each user. Each of the 8 basic sets of sequences (numbered BTS 0,16,32,48,64,80,96,112) has 16 circular shifts, so there are 128 sets of sequences represented in the table. This table has been constructed so that if any pair of the 128 sets is chosen, there will be at most one Gold code which uses the same pilot sequence in both sets. Users assigned this Gold code will still have different scrambling codes.

	User Number															
	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
BTS 0	s0	s1	s2	s3	s4	s5	s6	s7	s8	s9	s10	s11	s12	s13	s14	s15
BTS 1	s1	s2	s3	s4	s5	s6	s7	s8	s9	s10	s11	s12	s13	s14	s15	s0
BTS 2	s2	s3	s4	s5	s6	s7	s8	s9	s10	s11	s12	s13	s14	s15	s0	s1
BTS 3	s3	s4	s5	s6	s7	s8	s9	s10	s11	s12	s13	s14	s15	s0	s1	s2
BTS 4	s4	s5	s6	s7	s8	s9	s10	s11	s12	s13	s14	s15	s0	s1	s2	s3
BTS 5	s5	s6	s7	s8	s9	s10	s11	s12	s13	s14	s15	s0	s1	s2	s3	s4
BTS 6	s6	s7	s8	s9	s10	s11	s12	s13	s14	s15	s0	s1	s2	s3	s4	s5
BTS 7	s7	s8	s9	s10	s11	s12	s13	s14	s15	s0	s1	s2	s3	s4	s5	s6
BTS 8	s8	s9	s10	s11	s12	s13	s14	s15	s0	s1	s2	s3	s4	s5	s6	s7
BTS 9	s9	s10	s11	s12	s13	s14	s15	s0	s1	s2	s3	s4	s5	s6	s7	s8
BTS 10	s10	s11	s12	s13	s14	s15	s0	s1	s2	s3	s4	s5	s6	s7	s8	s9
BTS 11	s11	s12	s13	s14	s15	s0	s1	s2	s3	s4	s5	s6	s7	s8	s9	s10
BTS 12	s12	s13	s14	s15	s0	s1	s2	s3	s4	s5	s6	s7	s8	s9	s10	s11
BTS 13	s13	s14	s15	s0	s1	s2	s3	s4	s5	s6	s7	s8	s9	s10	s11	s12
BTS 14	s14	s15	s0	s1	s2	s3	s4	s5	s6	s7	s8	s9	s10	s11	s12	s13
BTS 15	s15	s0	s1	s2	s3	s4	s5	s6	s7	s8	s9	s10	s11	s12	s13	s14
BTS 16	s16	s17	s18	s19	s20	s21	s22	s23	s24	s25	s26	s27	s28	s29	s30	s31
Sets for BTS 17-31 formed by left circular shifts of BTS 16																
BTS 32	s0	s2	s4	s6	s8	s10	s12	s14	s16	s18	s20	s22	s24	s26	s28	s30
Sets for BTS 33-47 formed by left circular shifts of BTS 32																
BTS 48	s1	s3	s5	s7	s9	s11	s13	s15	s17	s19	s21	s23	s25	s27	s29	s31
Sets for BTS 49-63 formed by left circular shifts of BTS 48																
BTS 64	s23	s22	s21	s20	s19	s18	s17	s16	s15	s14	s13	s12	s11	s10	s9	s8
Sets for BTS 65-79 formed by left circular shifts of BTS 64																
BTS 80	s31	s30	s29	s28	s27	s26	s25	s24	s7	s6	s5	s4	s3	s2	s1	s0
Sets for BTS 81-95 formed by left circular shifts of BTS 80																
BTS 96	s0	s3	s6	s9	s12	s15	s19	s22	s25	s28	s31	s2	s7	s11	s16	s23
Sets for BTS 97-111 formed by left circular shifts of BTS 96																
BTS 112	s14	s17	s20	s23	s26	s29	s1	s4	s7	s10	s13	s18	s21	s24	s27	s30
Sets for BTS 113-127 formed by left circular shifts of BTS 112																

Figure 1 illustrates how the pilot sequences are constructed. Each user has a pilot sequence of 16 symbols with a spreading factor of 16. The Gold-like code for that user is combined with the scrambling code of the base station. Each pilot symbol is then modulated with a +1 or a –1 times (1+j), which is determined by the maximal-length sequence found in Table 1. As an example, suppose that a user has been assigned the 16-chip Walsh code [1 1 1 1 -1 -1 -1 -1 1 1 1 1 -1 -1 -1 -1], so that it is denoted user number 2, and that the base station is numbered BTS 1. The pilot sequence for this user is determined in a 3-step process.

1) Find the set of sequences for the base station in Table 4. The set of sequences for BTS 1 is [s1 s2 s3 s4 s5 s6 s7 s8 s9 s10 s11 s12 s13 s14 s15 s0].
2) Choose the particular maximal length sequence for the user from the set found in step 1. The user is assigned the third Walsh code, so it uses the third pilot sequence is the set, which is s3. When multiplied by (1+j), the sequence is

(1+j)[1 –1 –1 –1 1 –1 –1 1 1 –1 1 –1 1 1 1 1]

The sixteen pilot symbols used every time slot by this user are 1+j, -1-j, -1-j, -1-j, 1+j, -1-j, -1-j, 1+j, 1+j, -1-j, 1+j, -1-j, 1+j, 1+j, 1+j, 1+j.

3) Find the Gold-like code from Table 2. As an example, let the frame number be 5. The user number is 2, so the Gold-like sequence to be used is (2+5) mod 16 = 7. From Table 2 this sequence is [1 1 –1 –1 –1 –1 –1 1 1 –1 1 1 1 1 –1 1].

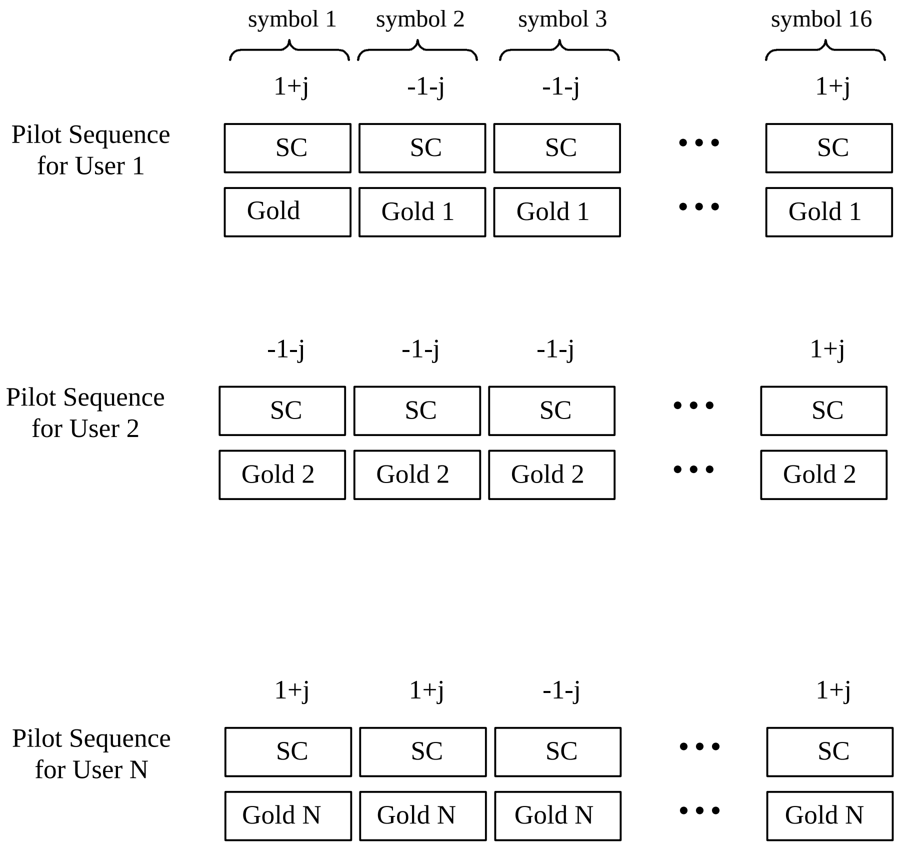

Figure 1: Construction of pilot sequences. The scrambling code is represented by SC and is common for all users in one cell. This scrambling code is combined with the Gold code of each user. Finally, the maximal-length sequence (which is different for each user) modulates the pilot symbols.

3.0 COMPARISON OF PILOT SYMBOL APPROACH AND SPECIAL MIDAMBLE APPROACH

3.1 Performance comparison

We now do link level simulations to evaluate the performance gains of using these pilot symbol sequences instead of the specially constructed midamble. The link level simulation parameters used are given in Table 5:

Table 5: The simulation parameters used to compare the performance of the midamble versus the performance with pilot symbols.

	Vehicular	Indoor-to-outdoor pedestrian
Velocity	120 kmph (Figures 2 and 4)	3 kmph (Figure 3 and 5)
Spreading gain (SF)	16	16
Number of users	8	8
Midamble parameters	256 chips, basic sequence is 192 chips	256 chips, basic sequence is 192 chips
Midamble channel estimation	Circular correlation performed with FFT, Mult., and IFFT	Circular correlation performed with FFT, Mult., and IFFT
Pilot symbol parameters	16 pilot symbols with spreading factor of 16	16 pilot symbols with spreading factor of 16
Pilot symbol channel estimation	Average over 16 pilot symbols	Average over 16 pilot symbols
Joint detection	Figures 4 and 5	Figures 4 and 5

3.1.1 Performance comparison with no joint detection

The performance with the Vehicular B channel is shown in Figure 2. The performance with perfect channel estimates is found by assuming that the receiver knows the channel exactly at the center of each time slot, and this perfect channel estimate is used throughout the time slot. The performance with perfect channel estimates is given so that the absolute loss due to imperfect channel estimation can be determined. With a spreading factor of 16 and with 8 users, the BER curve with the special midamble is about 0.7 dB worse than the curve generated with perfect channel estimates at a raw BER of 0.10. The BER curve with the pilot symbols is about 0.1 dB worse than the curve with the special midamble. The performance with the Outdoor-to-Indoor and Pedestrian channel is shown in Figure 3. The special midamble performs about 0.6 dB worse than the perfect channel estimates, and the pilot symbols perform about 0.4 dB worse than the perfect channel estimates at a raw BER of 0.03.

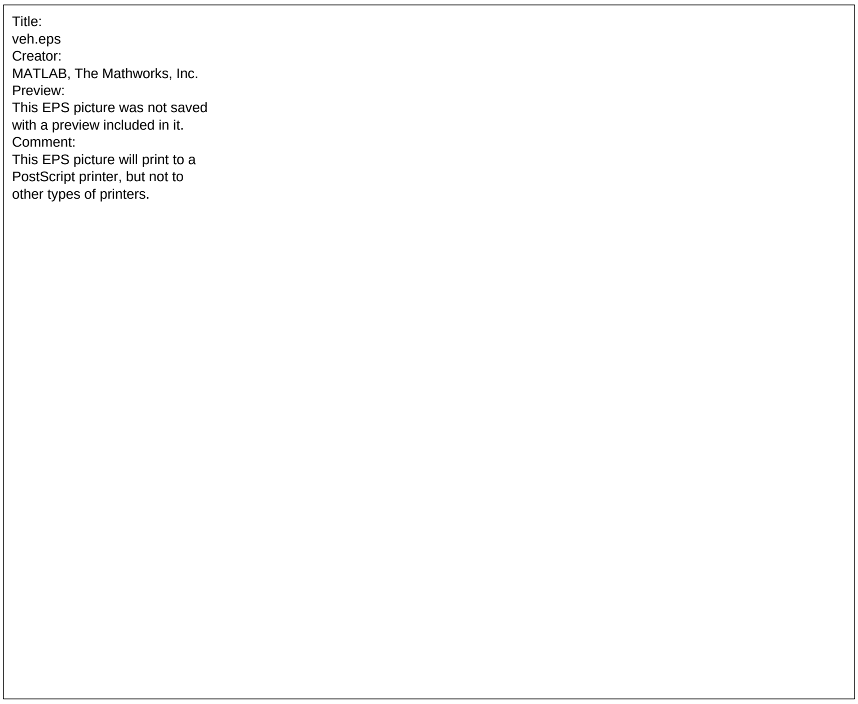

Figure 2: Link level simulations comparing the raw BER performance with the midamble and with pilot symbols for the downlink using the Vehicular B channel model. The spreading gain is 16 and the number of users is 8. The special midamble performs about 0.7 dB worse than the perfect channel estimates at a BER of 0.10. The pilot symbols perform about 0.1 dB worse than the special midamble.

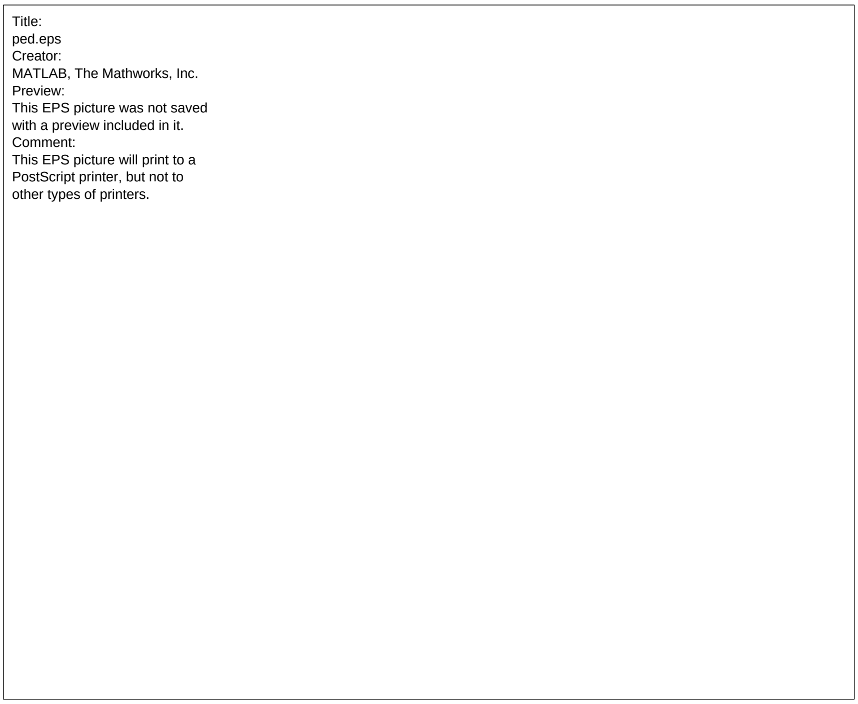

Figure 3: Link level simulations comparing the raw BER performance with the midamble and with pilot symbols for the downlink using the Outdoor-to Indoor and Pedestrian channel model. The spreading gain is 16 and the number of users is 8. The special midamble performs about 0.6 dB worse than the perfect channel estimates at a BER of 0.03. The pilot symbols perform about 0.4 dB worse than the special midamble.

3.1.2 Performance comparison with joint detection

In WG1 #5 in Cheju, it was requested that performance comparisons including joint detection should be made. There are two main types of joint detectors that could be used at the mobile -- linear detectors such as the decorrelating detector and subtractive interference cancellers such as the parallel interference canceller. The decorrelating detector used in these simulations is the zero-forcing block linear equalizer (ZF-BLE) [6], and the subtractive interference canceller used is the partial parallel interference canceller with soft decisions [7]. Only one stage of partial PIC was used with a cancellation factor of 0.5. The type of joint detection to be used will not be standardized, so vendors will be able to achieve differentiation be employing different joint detectors. Only the simplest PIC was used in these simulations, and better results should be achievable by using multi-stage interference cancellers.

Figures 4 and 5 show the results of simulations for the Vehicular and Pedestrian channels, respectively. There are two sets of curves in each plot. The solid lines show the BER performance with the ZF-BLE, and the dashed lines show the performance with the 1-stage partial PIC.

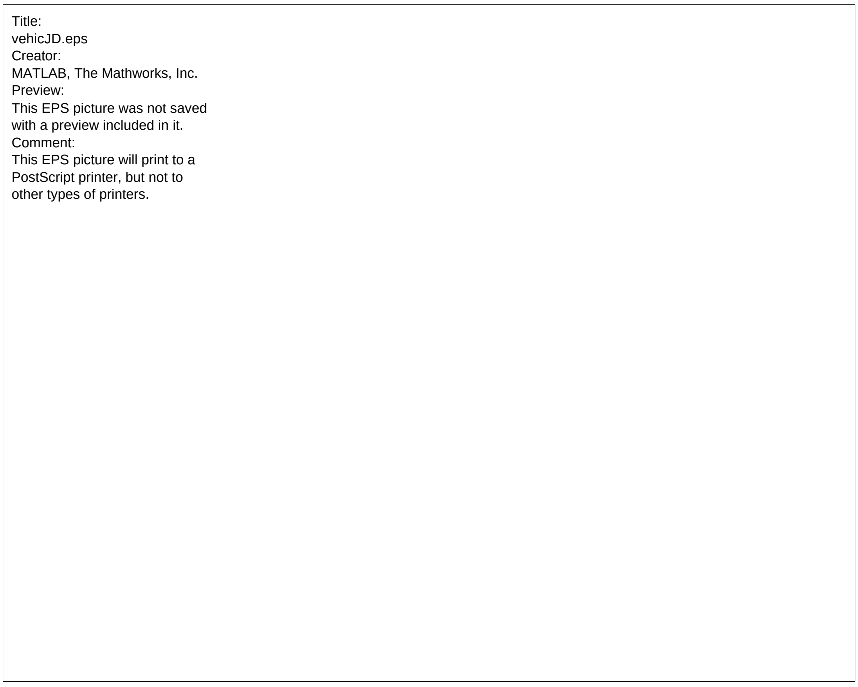

Figure 4: Link level simulations comparing the BER performance with joint detection for the special midamble and for pilot symbols for the downlink using the Vehicular B

channel model. The spreading gain is 16 and the number of users is 8. The pilot symbols perform about 0.1 dB worse than the midambles, and the 1-stage PIC outperforms the ZF-BLE.

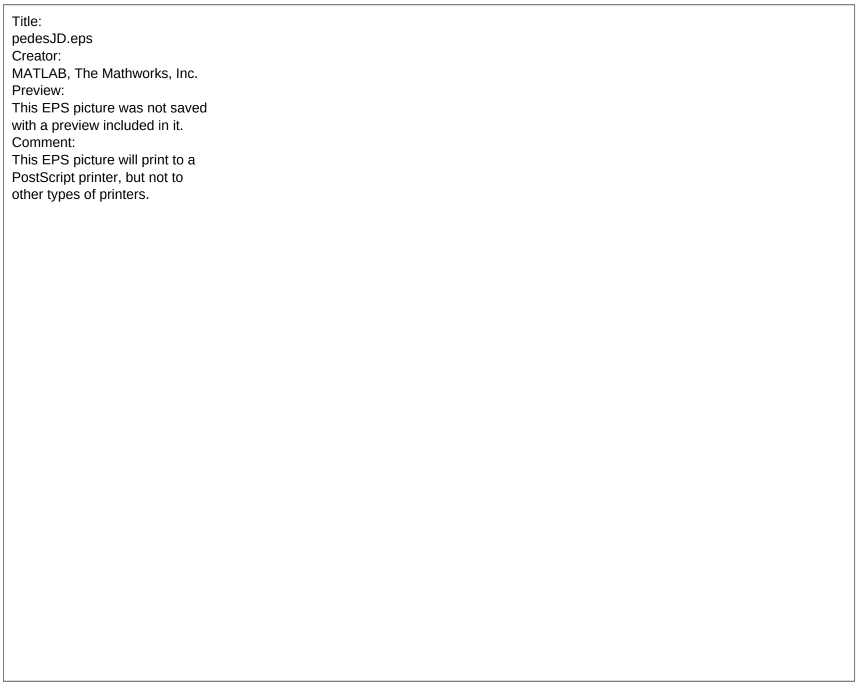

Figure 5: Link level simulations comparing the BER performance with joint detection for the special midamble and for pilot symbols for the downlink using the Outdoor-to Indoor and Pedestrian channel model. The spreading gain is 16 and the number of users is 8. The pilot symbols perform about 0.4 dB worse than the midambles, and the 1-stage PIC outperforms the ZF-BLE.

3.2 Cross correlations in a multi-cell environment

TDD systems will operate in a multi-cell environment, so channel estimation performance in such an environment must be studied. Figure 6 shows how the cross correlations between base stations will affect channel estimation. In Figure 6, there are 2 base stations each with 8 users. The midamble from a different base station will have some cross correlation with the midamble of the home base station and will affect channel estimation. If channel estimates are performed over 24 chips, then the cross correlation can be calculated at each position. The position with the maximum cross correlation is used to generate Figure 6. The ratio of the power of the channel estimate from the home base station to the power of a channel estimate computed using the signal from the interfering base station is computed. An interference rejection of 6 dB means that if a channel estimate from the home base station has a power of 1 (amplitude of 1), then the interfering base station will cause a channel estimate on one of the 24 paths to have a power of 0.25 (amplitude of 0.5) if both base stations are received with equal power.

For example, let a mobile be at the boundary of two cells, and let it receive a signal from two base stations with equal power. Let the mobile receive a single path from each base station. If the interference rejection is 6 dB, then when the mobile measures the channel from base station 1, it will measure one path with amplitude 1 and a second path (due to the cross correlation with the interfering base station) with amplitude 0.5. The mobile will assume that there are two paths coming from base station 1 and will use a maximal ratio combiner with 2 paths. Since the correct path is weighted with amplitude 1 and the false path is weighted with amplitude 0.5, the mobile suffers a loss in E_b/N_o of 1 dB. In order for a system to function well in a multi-cell environment the interference rejection must be kept as high as possible.

In Figure 6, the curve for the midambles was generated by computing the cross correlations between all the possible pairs of the 8 midambles of length 192 given in [1] assuming that each cell supports 8 users. In over 50% of the cases, the interference rejection was less than 6 dB. Similar simulations were performed with the pilot symbol sequences. With the pilot symbols, if the power in the channel estimates is averaged noncoherently over 16 time slots, then the interference rejection is on average about 6 dB better than with the special midambles. This is possible since the Gold codes used by each user change from frame to frame, but the midambles do not change from frame to frame.

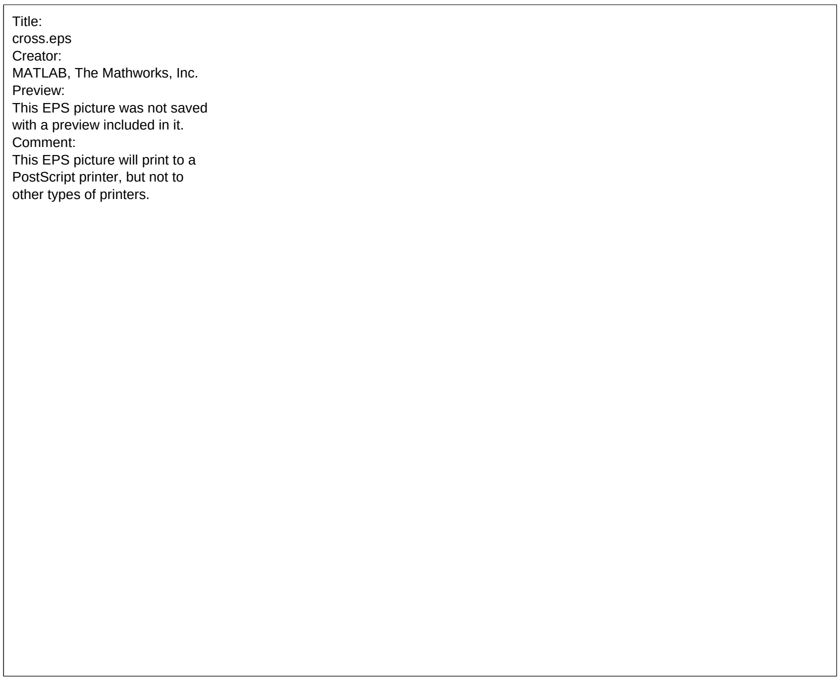

Figure 6: Intercell rejection for pilot symbols and the special midambles. 50% of the time the intercell rejection with the midambles is less than 6 dB. The intercell rejection with pilot symbols is about 6 dB better than that of the midambles.

3.3 Discussion on flexibility of the two approaches

The special midamble with 8 users has the ability to calculate channel estimates in a window of 24 chips. This window must be large enough to contain the entire channel impulse response and extra margin for any timing errors. If any paths extend outside the search window, the user will assume that they belong to another user and will not include them in the maximal ratio combiner. Any paths from other users that intrude into the window will be assumed to belong to that user, and the user will include them in the maximal ratio combiner even though these paths do not exist. There are penalties when the impulse response does not fit entirely within the window, and the length of the impulse response can change over time, so the window should be large enough to guarantee that the entire impulse response is contained within the window.

The penalty for having a window too small for the channel impulse response should not be overlooked. Assume that only one path extends outside of the channel estimation window so that a particular user will not include this path in the maximal-ratio combiner (MRC) and will include a false path from another user in the MRC. Then the user suffers the following losses:

1) Loss in E_b/N_o from not including the one path that is located outside the channel estimation window.
2) Loss in diversity from not including this path in the MRC. This will be especially significant if there are only 2 or 3 paths arriving from the base station.
3) Extra noise from including the false path in the MRC.

As an example, if there are 2 paths arriving from the base station with powers of 0 dB and –10 dB, and the weaker path falls outside the channel estimation window, the user will suffer a loss in E_b/N_o of over 1 dB.

Thus, the use of the special midamble for channel estimation severely limits the length of channel impulse response that can be tolerated. In order to increase the length of the search window, fewer users can be supported. The pilot symbol approach provides much more flexibility in the number of users and the lengths of channel impulse responses that can be supported. Similar to the FDD system, the maximum number of users is 16, and there are no hard limits on the lengths of the channel impulse responses. Channel estimates can be computed for all positions in each window.

3.4 Support of users with different delay spreads

In an indoor environment, generally the delay spreads of all users in a cell will be small. In an outdoor environment, however, users generally have widely varying delay spreads with users located at the edge of a cell having larger delay spreads than users closer to the base station. With the special midamble, the number of users that can be supported is inversely proportional to the largest delay spread expected by any user in the cell. The midamble is generated by repeating the same basic sequence with equal spacing between repetitions. If users at the edge of the cell have large delay spreads and require large spacings between repetitions, then the number of users supported in a cell will be severely limited.

3.5 Computational complexity comparison

3.5.1 Hierarchical structure for channel estimation

In the following comparison, channel estimates are computed at all positions in the search window. With a midamble of length 192 chips and 8 users, the search window size is 24 chips long (6 μs). Any paths that occur outside this 24-chip window will appear to belong to the next user and will lead to higher bit error rates. Complexity comparisons will be made with search window sizes of 8, 24, and 50 chips. The complexity of the channel estimation with and without joint detection (JD) will be analyzed here. The difference is that channel estimates will either be computed for all users or only for a single user. ***When computing channel estimates with pilot symbols, a key observation is to note that the pilot symbols form a hierarchical sequence.*** For a particular user, the combination of the Gold code and scrambling code will be the same for all 16 symbols. The only difference between the 16 symbols is the pilot symbol sequence. Thus the hierarchical structure of Figure 7 can be used to compute channel estimates at all the desired positions. This structure is very flexible and can handle any search window size [e.g. 1 us (4 chips) or 30 us (123 chips)]. The complexity is approximately proportional to the size of the search window.

In Figure 7, the hierarchical structure is set up to minimize the memory required. The complexity would be the same if the two sets of blocks were swapped, but the memory requirements would increase. The first block in the upper left corner of Figure 7 shows a 240-element complex memory for the samples at chip rate. For each user, every sixteenth sample has the PN sequence removed, and there are 15 additions to compute the output to feed to the second stage. The 240-element memory is shared by all the users. In the blocks on the right side of Figure 7 the spreading and Walsh codes are removed for each user, and there are 15 additions to compute the channel estimates for each position within the window. Since the channel estimates are not available until the middle of the slot, samples for the entire time slot will probably be stored in a buffer of size (154 symbols)(16 chips)(2 for complex) = 4928 bytes. The 240-element memory is simply part of this buffer. The total memory required with this hierarchical structure is

4928 + (8*15)(2 for complex) = 5168 bytes

With the special midamble structure the FFT's can be computed in place, so the total memory required is 4928 bytes.

It should be noted that if all the possible midambles are stored at the mobile, this will take up additional memory with the midamble approach. Storing the 128 midambles of length 192 and 128 midambles of length 456 will take over 10 kbytes.

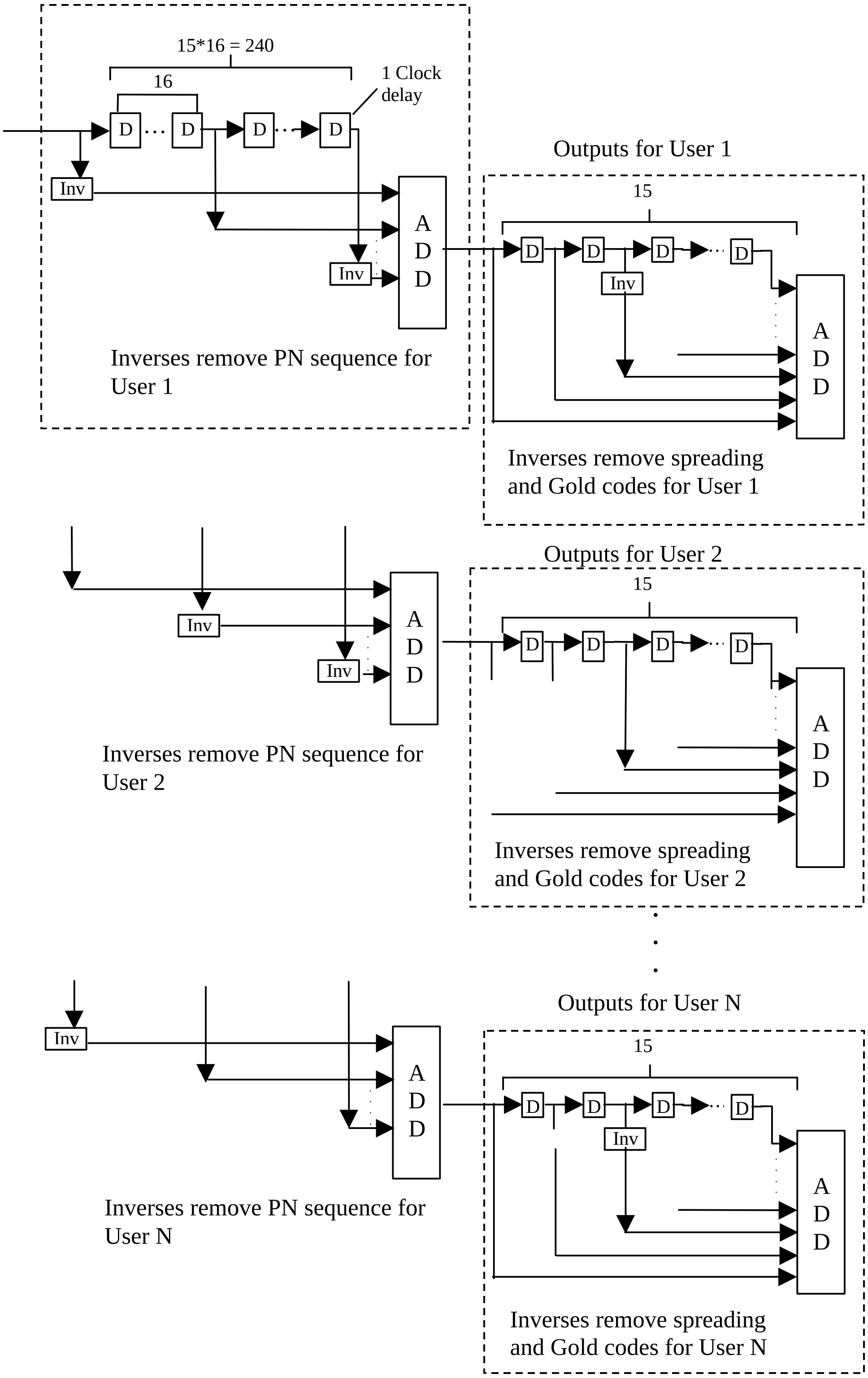

Figure 7:Hierarchical structure to efficiently compute channel estimates for multiple users.

3.5.2 Complexity of channel estimation with the special midamble

When joint detection is used at the mobile, the mobile estimates the channel for all the users in the cell which receive data in the same time slot. With 8 users, 8 separate channel estimates must be made for each of the positions in the search window. The analysis of the complexity of computing 2 FFT's of length 192 is given in [8] as 1.38 MIPS when the user receives data on one time slot per frame. The channel estimation is performed with an FFT, a complex multiplication, and an IFFT. If the complex multiplication takes 4 operations, then the total number of operations for channel estimation with the complex midamble is

FFT and IFFT: 1.38 MIPS
Multiplication:(192 chips)(4 for complex multiply)(100 frames) = 0.08 MIPS
TOTAL of 1.46 MIPS for channel estimation with joint detection

If joint detection is not used, then channel estimates can be computed for just one user. This requires

Channel estimate for 1 path: (191 adds)(2 for complex)(100 frames) = 0.0382 MIPS
For 24 paths, this requires 0.92 MIPS
TOTAL of 0.92 MIPS for channel estimation without joint detection

3.5.3 Complexity of Channel Estimation with Pilot Symbols

The computation of the channel estimate for the first position in the window for one user requires 255 complex additions. It requires 15*15 = 225 complex additions to fill up the hierarchical structure and then 15+15=30 complex additions to compute the channel estimate for each position. The total number of computations to compute channel estimates for all users over a 24 chip window is

(225+30*24)(2 for complex)(8 users)(100 frames) = 1.51 MIPS
TOTAL of 1.51 MIPS for channel estimation with joint detection

This complexity is almost the same as the complexity with the special midamble. However, the pilot symbols with the hierarchical structure are much more flexible than the special midamble. With the pilots there is no limitation that channel estimation has to be performed over 24 chips. If joint detection is not used at the mobile, the complexity of channel estimation is

(225+30*24)(2 for complex)(1 user)(100 frames) = 0.19 MIPS
TOTAL of 0.19 MIPS for channel estimation and DPE for multiuser detection case

A summary of all these results is presented in Table 6 below.

Table 6: Comparison of complexity of channel estimation and delay profile estimation using either pilot symbols or the special midamble. The complexity of channel estimation with the special midamble is at least twice that needed with pilot symbols in most scenarios.

	Search window length: 8 chips (2 µs)		Search window length: 24 chips (6 µs)		Search window length: 50 chips (12 µs)	
	No joint detection	With joint detection	No joint detection	With joint detection	No joint detection	With joint detection
Pilot symbols	0.10 MIPS	0.74 MIPS	0.19 MIPS	1.51 MIPS	0.35 MIPS	2.76 MIPS
Special midamble	0.31 MIPS	1.46 MIPS	0.92 MIPS	1.46 MIPS	FAILS	FAILS

4.0 Conclusions

We have shown that the use of pilot symbol sequences constructed with maximal length sequences for channel estimation gives the following advantages over the special midamble technique:

1) The performance of the two techniques was shown to be similar with and without joint detection with the pilot symbols performing 0.1 to 0.4 dB worse than the special midambles.

2) In a multi-cell environment the robustness of the special midambles to intercell interference is very limited. When a user receives signals from two base stations with equal power, in over 50% of the cases the cross correlation with the midamble of the interfering base station will cause false paths to appear which are only 3 to 6 dB down from the strongest path of the home base station, which will degrade the performance with the special midambles

3) The pilot symbol approach is much more flexible than the special midamble approach. With pilot symbols twice the number of users can be supported for channel estimation, and larger (or smaller) search window sizes can be used.

4) The special midamble must be designed to provide channel estimation in windows that are larger than the largest expected delay spread. In outdoor environments there may be a few users at the edge of the cell with large delay spreads. The channel estimation windows must be large enough to accommodate these large delay spreads, so very few users can be supported in these cells.

5) In most cases the complexity of channel estimation is much lower with the pilot symbols approach.

Thus, since the pilot symbol based channel estimation has similar performance, no constraints on the delay spread and reduced complexity as compared to the midamble based channel estimation, we propose that channel estimation be performed using pilot symbols.

References

[1] 3GPP RAN WG1, "Transport channels and physical channels description (TDD)", Tdoc 267/99, (S1.21) V1.0.1., March 1999.

[2] B. Steiner and P. Baier, "Low cost channel estimation in the uplink receiver of CDMA mobile radio systems", Frequenz, 47 (1993) 11-12, pp. 292-298.

[3] Sarwate, D. and Pursley, M., "Crosscorrelation properties of pseudorandom and related sequences," *Proceedings of the IEEE*, Vol. 68, No. 5, May 1980, pp. 593-619.

[4] 3GPP RAN WG1, TS 25-213 V2.1.0 (1999-4), section 4.3.3.2.

[5] Panasonic, "Cross correlation performance of TDD 16 chip scrambling codes", TSGR1#4(99)413, Yokohama, Japan, 19-20 April 1999.

[6] A. Klein, G. Kaleh, and P. Baier, "Zero forcing and minimum mean-square-error equalization for multiuser detection in code-division multiple-access channels", *IEEE Transactions on Vehicular Technology*, Vol. 45, No. 2, May 1996, pp. 276-287.

[7] D. Divsalar, M. Simon, and D. Raphaeli,, "Improved parallel interference cancellation for CDMA", *IEEE Transactions on Communications*, Vol. 46, No. 2, Feb. 1998, pp. 258-268.

[8] Motorola, "Complexity of multiple channel estimations and the SU", TSGR1#4(99)389, Yokohama, Japan, April 1999.