A Code Set for a Robust 8.25 Mb/s Data Rate for IEEE 802.11b WLANs
Subhendu Batabyal, Bikash Kumar Dey, Member, IEEE, and V. Umapathi Reddy, Life Fellow, IEEE

Abstract—In this paper, a code set for an 8.25-Mb/s data rate is proposed for the Institute of Electrical and Electronics Engineers 802.11b system, which can act as a better fallback option than a 5.5-Mb/s data rate when an 11-Mb/s data rate is not sustainable in a large delay-spread channel environment. We present a systematic procedure for the design of the code set. The proposed code set works at significantly higher path loss when compared to an 11-Mb/s data rate in moderate to large delay-spread channel environment. We also discuss the resource requirement for including the proposed data rate in the existing 802.11b transceiver and note that the additional hardware is minimal. Since the current Wireless Fidelity chipsets are enabled with the 802.11g option, we investigated the performance of the proposed data rate and the 6- and 9-Mb/s data rates provided by 802.11g, and the results show that the 8.25-Mb/s data rate is a reliable fallback option compared to the 9-Mb/s rate when the 11-Mb/s data rate is not sustainable. The 8.25-Mb/s data rate is shown to work at 1.7 dB higher path loss when compared to the 9-Mb/s rate in a frequently occurring moderate delay-spread channel.

Index Terms—Complementary code set, differential phase-shift keying (DPSK), fast Walsh transform, spread spectrum based wireless local area network (WLAN) system, 802.11 b/g WLAN.

I. INTRODUCTION

THE IEEE 802.11 standard for wireless local area networks (WLANs) defines the physical and medium access control layers for WLANs. Data rates of 1 and 2 Mb/s are supported by direct sequence spread spectrum (DSSS). The IEEE 802.11b, which is the high rate extension of the 802.11 standard, uses a coding scheme known as complementary code keying (CCK) for data rates of 5.5 and 11 Mb/s. A code set consisting of 64 codewords is used for the 11-Mb/s rate, and a subset with four codewords is used for the 5.5-Mb/s rate. The choice of CCK was due to the good distance and autocorrelation properties as well as robustness of the complementary codes against multipath [1], [2], [8]. Also, these codes can be decoded using a simple modified fast Walsh transform (FWT) [3].

The 64 codewords used for the 11-Mb/s data rate have inferior cross correlation compared to that of four codewords (used for 5.5 Mb/s data rate), which are mutually orthogonal.

Typical CCK receivers employ channel matched filter (CMF) [2], [4] and a decision feedback equalizer (DFE) [4] to combat intersymbol interference (ISI). In view of the poor correlation properties of the 64 code sets, it is difficult to realize the 11-Mb/s data rate even in a moderate delay-spread channel environment. When this rate is not sustainable, the present 11b falls back to a significantly lower data rate of 5.5 Mb/s. This motivated us to look for an intermediate data rate that is more robust to delay-spread channels.

Table I shows the modulation schemes used for the 5.5- and 11-Mb/s data rates and the possible schemes for an intermediate rate of 8.25 Mb/s that we propose. The first option for this rate with differential 8-phase-shift keying (D8PSK) is not desirable because of the high signal-to-noise ratio (SNR) penalty of 7–8 dB in moving from differential quadrature phase-shift keying (DQPSK) to D8PSK [7]. Hence, we focus on the second option, which requires a search for 16 codewords with good correlation properties. Selection of the code set must be such that the receiver structure for decoding is similar to the standard 802.11b receiver. Further, for robustness against multipath, we want the selected code set to form a complementary set.

We first arrive at the desired properties of the code set and then design a set of 16 codewords that will possess these properties. We illustrate the performance of the designed set in typical WLAN channel environments. We also compare the performance of the proposed data rate with that of the 6- and 9-Mb/s data rates, which are provided by 802.11g mode, as the current Wireless Fidelity chipsets are enabled with the 802.11g option. The results show that the 8.25-Mb/s data rate is a reliable fallback option compared to the 9-Mb/s rate when the 11-Mb/s data rate is not sustainable. We further bring out that the additional resource requirement for including the proposed data rate in the existing 802.11b transceiver is minimal.

Section II outlines the design criteria, while Section III presents the methodology of the code set design. Section IV describes the design of specific code sets. Section V brings out the additional hardware required for including the 8.25-Mb/s data rate in the existing 802.11b transceiver. In Section VI, we discuss simulation set up and the results. In Section VII, we...
Comment on interoperability issues for the 8.25-Mb/s data rate, and finally, in Section VIII, we conclude this paper.

II. CODE SET DESIGN CRITERIA

Consider a typical 802.11b system consisting of a delay-spread channel, a CMF, a correlator bank, and a code decision block that decides which codeword has been transmitted. The blocks are shown in Fig. 1, where \( z[n] \) is a white Gaussian noise process, \( r[n] \) is the input to CMF, and \( b[n] \) is its output shifted by \( (M - 1) \) samples, where \( M \) denotes the channel impulse response length. Here, the sampling rate is assumed to be same as the chip rate. However, the same analysis will hold if we use a sampling rate that is an integer multiple of chip rate and downsampling the CMF output to the chip rate.

Hereafter, a discrete time sequence will be denoted by a bold-faced lowercase letter, e.g., \( x, y, c \), possibly with subscripts. The \( n \)th sample of \( x \) will be denoted by \( x[n] \). For any integer \( l \), \( x[l] \) will denote the sequence obtained by delaying \( x \) by \( l \) samples. For two finite length sequences \( x \) and \( y \), \( \langle x, y \rangle \) will denote the sum \( \sum_{i=-\infty}^{\infty} x[i]y[i] \), in which only finitely many terms are nonzero. Since we will restrict to finite length sequences, the summation is always well defined. A codeword will also be viewed as a sequence with the first component as the 0th sample.

Let us denote the channel impulse response (spaced at chip interval) by \( \mathbf{h} \) and assume it to be of length \( M \) (for \( i < 0 \) and \( i > M - 1 \)). Assuming perfect channel estimation, the \( i \)th coefficient of the CMF impulse response is given by \( h^*[M - 1 - i] \), where the superscript * denotes complex conjugation. Suppose that a codeword \( x \) is transmitted. To keep the analysis simple, we assume that the ISI is completely removed through equalization, and hence, a single codeword can be considered in isolation ignoring the previous and future transmitted codewords. We also assume perfect symbol synchronization. Then, the input to the CMF is

\[
r[n] = \sum_{k=0}^{M-1} h[k] x[n - k] + z[n]
\]

(1)

and the output of the CMF can be shown to be

\[
b'[n] = \sum_{i=0}^{M-1} |h[i]|^2 x[n - (M - 1)] + \sum_{p \neq k} h[k] h^*[p] x[n - ((M - 1) - (p - k))] + \sum_{r=0}^{M-1} h^*[r] z[n - (M - 1 - r)]
\]

(2)

where \( S = [0, M - 1] \) (for \( m \) \( < \) \( n \), \( [m, n] \) denotes the set of integers from \( m \) to \( n \)). Shifting the CMF output by \( (M - 1) \) samples and denoting the shifted sequence by \( b \), we have

\[
b[n] = x[n] \sum_{i=0}^{M-1} |h[i]|^2 + \sum_{r=0}^{M-1} h^*[r] z[n + r] + \sum_{l=1}^{M-1} \sum_{(p-k) \neq l} (h[k] h^*[p] x[n - l] + h[p] h^*[k] x[n + l])
\]

and finally, in Section VIII, we conclude this paper.
The output of the correlator, with a reference codeword, say \( y \), other than the transmitted codeword, is given by

\[
\langle y, b \rangle = \langle y, x \rangle \sum_{i=0}^{M-1} |h[i]|^2 + \langle y, z' \rangle + 2R \langle h[k]h^*[p] \langle y, x^{(l)} \rangle \rangle.
\]

In an ideal case, where all the codewords are uncorrelated with each other, the first two terms would have been zero. Since they are not, we treat them as interference. Denoting them by \( I_2 \) and \( I_3 \), we have

\[
I_2 = R_{yx}[0] \sum_{i=0}^{M-1} |h[i]|^2 \tag{6}
\]

\[
I_3 = \sum_{l=1}^{M-1} \sum_{(p-k)\in S} 2R \langle h[k]h^*[p]R_{yx}[l] \rangle \tag{7}
\]

where \( R_{yx}[l] = \langle y, x^{(l)} \rangle \) is the cross correlation between \( y \) and \( x \) for lag \( l \).

Now consider the channel gain term \( g_{ch} = \sum_{i=0}^{M-1} |h[i]|^2 \). For a transmit power of \( P \) dBm and a channel gain \( g_{ch} \), an equivalent system is one in which the channel taps are scaled by \( (1/\sqrt{g_{ch}})(h[i]) = (1/\sqrt{g_{ch}})h[i] \), and the transmit power is changed to \( P' = P + G_{ch} \), where \( G_{ch} = 10\log(g_{ch}) \). Hence, without loss of generality, we assume \( g_{ch} = 1 \). Then, we can rewrite (4) and (6) as

\[
D = \|x\|^2 \quad \text{and} \quad I_2 = R_{yx}[0] \tag{8}
\]

which indicate that the interference term \( I_2 \) does not depend on the channel and depends only on the zero lag cross correlation of \( x \) and \( y \). In an additive white Gaussian noise (AWGN) channel, \( I_1 \) and \( I_3 \) are zero, and therefore, \( I_2 \) is the only interference term, and \( R_{xz'} = \langle x, z' \rangle \) and \( R_{yz'} = \langle y, z' \rangle \) are the noise terms. In other words, the AWGN performance does not depend on the nonzero lag autocorrelation and cross correlation of the codewords. Thus, in AWGN, the decision will be made in favor of the transmitted codeword \( x \) if for all \( \neq x \)

\[
\|x\|^2 + R_{xz}\| > R_{yx}[0] + R_{yz}\| \tag{9}
\]

which suggests that for maximizing the decision in favor of \( x \), we need to minimize the maximum value of \( |R_{yx}[0]| \).

In a delay-spread channel, \( I_1 \) and \( I_3 \) depend on the channel profile, but \( I_2 \) is unchanged. In this case, the decision will be made in favor of the correct codeword \( x \) if for all \( \neq x \)

\[
\|x\|^2 + I_1 + R_{xz}\| > R_{yx}[0] + I_3 + R_{yz}\|. \tag{10}
\]

Thus, in the case of delay-spread channel, both \( I_1 \) and \( I_3 \) have a role in the decision process. If these terms are made as small as possible, the performance in the delay-spread channel tends to that in the AWGN channel.

The following steps are followed in designing the code set:

1) minimize the maximum magnitude of zero lag cross correlation between any two codewords for ensuring best performance in an AWGN channel;

2) minimize the maximum magnitude of nonzero lag autocorrelation (of the codewords) and the maximum magnitude of nonzero lag cross correlation (between any two codewords) for ensuring best performance in the delay-spread channel.

III. METHODOLOGY FOR THE DESIGN OF THE CODE SET

A set of \( N \) codewords \( \{c_1, c_2, \ldots, c_N\} \) of length \( L \) is said to be complementary [11]–[13] if

\[
\sum_{k=1}^{N} R_{ck}(l) = NL\delta(l), \quad l = 0, 1, 2, \ldots, L - 1 \tag{11}
\]

where \( \delta(l) \) is the Kronecker delta, and \( L = 8 \) for the codewords used in 802.11b.

For constructing a set of \( 16 \) codewords with good cross correlation and autocorrelation, we take a greedy approach and proceed as follows.

1) Choose a set \( C = \{c_1, c_2, \ldots, c_8\} \) of eight orthogonal codewords that form a complementary set.

2) Find another set \( C' = \{c'_1, c'_2, \ldots, c'_8\} \) of eight orthogonal codewords such that the maximum magnitude of the zero lag cross correlation between a codeword in \( C' \) with codewords in \( C \) is minimized, checking simultaneously that the derived set of codewords \( (C') \) has good nonzero lag autocorrelation and the combined set \( C \cup C' \) has good nonzero lag cross correlation.

Henceforth, with abuse of notation, an uppercase \( C \) (possibly with superscript or subscript) will denote a set of codewords as well as the matrix with those codewords forming the rows. Since \( C \) is a complementary set, \( C' \) needs to be a complementary set for the combined set \( C \cup C' \) to be complementary. For any orthogonal set \( C' \) with same norm as that of \( C \), there is a unique orthonormal matrix \( V \) such that \( C' = VC \).

To arrive at a proper \( V \), we need to analyze the effect of \( V \) on the cross-correlation properties of the combined set of codewords \( C = C \cup C' \). We note that \( CC^H = 8I \) and \( CC'^H = 8I \), and the magnitude of the zero lag cross correlation of a codeword \( c_i \) from the set \( C \) with a codeword \( c'_p \) in the set \( C' \) is

\[
|\langle c_i, c'_p \rangle| = \left| \sum_{r=1}^{8} v_{pr}c_r \right| = \sum_{r=1}^{8} |v_{pr}c_r| = 8|v_{pr}|. \tag{12}
\]

Here, \( v_{pr} \) denotes the element of matrix \( V \) in the \( p\text{th} \) row and \( r\text{th} \) column. Now, the maximum value of the interference term \( I_2 \) [see (6)] is determined by the quantity \( \max |\langle c_i, c'_p \rangle| \) and hence by \( \max |v_{pr}| \). Under the constraint that

\[
\sum_{r=1}^{8} |v_{pr}|^2 = 1 \quad \forall p \in [1, 8] \tag{13}
\]

\( \max |v_{pi}| \) is minimized when \( |v_{pi}| = 1/\sqrt{8} \forall i \in [1, 8] \) (recall that \( [1, 8] \) denotes the set \{1, 2, \ldots, 8\}).
Although a fully populated \( V \) matrix results in a zero lag cross correlation of minimum magnitude among the codewords, as we will see later, the resulting code set may not possess good autocorrelation. Also, the use of a partly populated \( V \) matrix would result in a lower number of codes that have nonzero lag cross correlation with any given code. We therefore set some elements of \( V \) to zero, in which case the constraint equation becomes

\[
\sum_{t \in S_p} |v_{pt}|^2 = 1
\]  

(14)

where \( S_p = \{ t \in [1, 8] | v_{pt} \neq 0 \} \forall p \in [1, 8] \). We make one convenient choice for \( V \), where we assume the same number (\( \lambda \)) of nonzero elements in each row, in which case the maximum magnitude of the zero lag cross correlation is minimized when

\[
|v_{pt}| = \frac{1}{\sqrt{\lambda}} \quad \forall t \in S_p \text{ and } \forall p \in [1, 8].
\]  

(15)

Now we analyze the effect of \( V \) on the complementary property of the code set \( C' \). Recall that \( R_{c'}(0) = (c'_n, c'_n) = 8 \). The nonzero lag autocorrelation of a codeword is given by

\[
R_{c'}(\Delta) = \left\langle c'_n, c'_n(\Delta) \right\rangle = \left\langle \left( \sum_{p=1}^{8} v_{np} c_p \right), \left( \sum_{k=1}^{8} v_{nk} c_k(\Delta) \right) \right\rangle
\]

which can be simplified as

\[
R_{c'}(\Delta) = \sum_{l=1}^{8} |v_{nl}|^2 \left\langle c_l, c_l(\Delta) \right\rangle + \sum_{p,k \in S_n \text{ } p \neq k} v_{np} v_{nk}^* \left\langle c_p, c_k(\Delta) \right\rangle
\]

\[
= \frac{1}{\lambda} \sum_{l \in S_n} R_{c_l}(\Delta) + \sum_{p,k \in S_n \text{ } p \neq k} v_{np} v_{nk}^* R_{c_p c_k}(\Delta)
\]

(16)

where nonzero elements of \( V \) are assumed to be of equal magnitude [see (15)]. Summing \( R_{c'}(\Delta) \) over \( n = 1 \) to \( 8 \), we obtain

\[
\sum_{n=1}^{8} R_{c'}(\Delta) = \sum_{n=1}^{8} \frac{1}{\lambda} \sum_{l \in S_n} R_{c_l}(\Delta)
\]

\[
+ \sum_{n=1}^{8} \sum_{p,k \in S_n \text{ } p \neq k} v_{np} v_{nk}^* R_{c_p c_k}(\Delta)
\]

(17)

which simplifies to

\[
\sum_{n=1}^{8} R_{c'}(\Delta) = \sum_{n=1}^{8} \frac{1}{\lambda} \sum_{l \in S_n} R_{c_l}(\Delta).
\]  

(18)

For \( C' \) to be complementary, we need

\[
\sum_{n=1}^{8} R_{c'}(\Delta) = 0 \quad \forall \Delta \in \{1, 2, \ldots, 7\}
\]

which will be met if

\[
\sum_{n=1}^{8} \frac{1}{\lambda} \sum_{l \in S_n} R_{c_l}(\Delta) = 0 \quad \forall \Delta \in \{1, 2, \ldots, 7\}.
\]  

(19)

A sufficient condition to achieve this is given by

\[
\sum_{n \in S_n} R_{c_l}(\Delta) = 0 \quad \forall n \in [1, 8]
\]

(20)

which means that \( \{c_l | l \in S_n\} \) is a complementary set \( \forall n \in [1, 8] \).

IV. DESIGN OF SPECIFIC CODE SETS

For \( C \), we choose a complementary set of eight orthogonal codewords among the 64 codewords (we denote the set of 64 codewords by \( X_{64} \)) used for the 11-Mb/s data rate. Arranging them as the rows of matrix \( C \), we have

\[
C = \begin{pmatrix}
1 & 1 & 1 & -1 & 1 & 1 & -1 & 1 \\
-1 & -1 & -1 & 1 & 1 & -1 & 1 & -1 \\
-1 & -1 & 1 & -1 & -1 & -1 & 1 & -1 \\
1 & -1 & -1 & 1 & 1 & -1 & 1 & -1 \\
1 & -1 & 1 & 1 & -1 & -1 & 1 & -1 \\
-1 & 1 & -1 & -1 & 1 & -1 & 1 & -1 \\
1 & -1 & -1 & -1 & 1 & -1 & 1 & -1 \\
-1 & 1 & 1 & 1 & 1 & 1 & 1 & 1
\end{pmatrix}
\]

This set is a complementary set and has several subsets of different sizes (2, 4, and 6) that are also complementary sets. For example, the first four codes and the last four codes of this set form complementary sets.

\[
V_1 = \frac{1}{4} \begin{pmatrix}
-1 - j & -1 + j & 1 - j & -1 - j & 1 + j & 1 - j & -1 - j & -1 + j \\
1 - j & 1 + j & 1 + j & -1 + j & -1 - j & -1 - j & 1 + j & -1 + j \\
-1 + j & -1 - j & 1 + j & -1 + j & -1 - j & -1 - j & 1 + j & -1 + j \\
1 + j & 1 - j & -1 - j & -1 + j & 1 - j & 1 + j & -1 - j & -1 + j \\
1 - j & 1 + j & 1 + j & -1 + j & 1 - j & 1 + j & -1 - j & -1 + j \\
1 - j & 1 + j & -1 + j & -1 + j & 1 - j & -1 - j & -1 + j & 1 + j \\
1 - j & 1 + j & -1 + j & -1 + j & 1 - j & -1 - j & -1 + j & 1 + j \\
1 + j & 1 - j & -1 - j & -1 + j & 1 - j & -1 - j & -1 + j & 1 + j
\end{pmatrix}
\]  

(21)
We first choose a dense unitary matrix $V_1$, as shown in (21) at the bottom of the previous page. Note that all the elements of $V_1$ are of equal magnitude. Transforming $C$ with $V_1$, we get


From (18) and (20), the code set $C'_1$ forms a complementary set.

Now consider the set of 16 codewords $C \cup C'_1$. From (12) and (15), it is clear that $|R_{c_i c'_p}(0)| = \sqrt{8}$ for all $i, p \in [1, 8]$, where $c_i$ is a codeword in $C$, and $c'_p$ is a codeword in $C'_1$. The maximum magnitude of zero lag cross correlation between any two codewords in $C \cup C'_1$ is found to be $3$ dB less than that for the original set of 64 ($X_{64}$). Table II shows the number of pairs of codewords with nonzero lag cross-correlation magnitude in the neighborhood of the maximum value, which is same for code sets $X_{64}$ and $C \cup C'_1$. We note from the table that in the set $C \cup C'_1$, there are fewer pairs of codewords with nonzero lag cross correlation in the neighborhood of the maximum value compared to the set $X_{64}$. However, the maximum magnitude of the nonzero lag autocorrelation for the resulting code set $C'_1$ (see Fig. 2) is $1.38$ dB higher than that of $X_{64}$. Simulation results for the AWGN channel, as given in Table III, show that for a bit error rate (BER) of $10^{-5}$, the required $E_s/N_0$ by the code set $C \cup C'_1$ is $2.8$ dB less than that required by the code set $X_{64}$ and only $0.9$ dB more when compared to the code set corresponding to the $5.5$-Mb/s data rate. However, this code set was seen to perform poorly in large delay-spread channel because of its poor nonzero lag autocorrelation.

We would like to improve the performance in delay-spread channel without sacrificing performance in the AWGN channel significantly. For this, we have to improve the nonzero lag autocorrelation without significant degradation in the cross correlation.

If we choose semi-dense $V$ matrix with four ($\lambda = 4$) nonzero elements of equal magnitude per row, which are placed at appropriate positions according to (20), it ensures the complementarity of the resulting code set and also minimizes the maximum magnitude of zero lag cross correlation for this choice of $\lambda$. This maximum magnitude with such $V$ is $1.5$ dB ($10 \log \sqrt{8}/\sqrt{4}$) higher compared to that with a dense $V$. However, we will still have a $1.5$-dB advantage compared to the original set ($X_{64}$).

In the case of semi-dense $V$, the number of codewords with which any given codeword in the resulting code set has nonzero lag cross correlation reduces to four (from eight in the case of fully populated $V_1$), and the number of codewords orthogonal to any given codeword increases to 11 (from seven in the case of $V_1$). We thus choose a unitary matrix $V_2$, as shown in (22).
at the bottom of the previous page. Note that we have chosen the positions of nonzero elements in accordance with (20), since the first four and the last four codes in $C$ form a complementary set. Transforming $C$ with $V_2$, we get $C_2'$, as shown in (23) at the bottom of the page.

Table III shows that the BER performance of $C \cup C_2'$ in the AWGN channel is only 0.6 dB poorer than that of $C \cup C_1'$. The maximum magnitude of nonzero lag cross correlation for $C \cup C_2'$ is the same as for $C \cup C_1'$ and $X_{64}$. From Table II, we see that the number of pairs of codewords with nonzero lag cross correlation in the neighborhood of the maximum value is similar to that in $C \cup C_1'$. However, Fig. 2 shows that the maximum magnitude of nonzero lag cross correlation while ensuring that the cross correlation (for zero as well as nonzero lags) is not significantly sacrificed compared to that of $C \cup C_1'$. Thus, with the new set $C \cup C_2'$, we have significantly improved the nonzero lag autocorrelation and also show the suitability of this code set for delay-spread channels. We thus propose $C \cup C_2'$ as the choice of 16 codewords for 8.25-Mb/s data rate.

V. HARDWARE MODIFICATION REQUIRED FOR 8.25-Mb/s DATA RATE

Figs. 3 and 4 show the block level architecture of the transmitter and receiver for the Extended Rate PHY-DSSS (ERP-DSSS) mode of 802.11g (5.5- and 11-Mb/s data rates). The blocks with double borders are the ones that need to be implemented differently for the 8.25-Mb/s rate. The remaining blocks are unchanged. At the transmitter side, the CCK code mapper is the only block that undergoes a change to accommodate 8.25 Mb/s. A typical CCK code mapper for the 11-Mb/s rate generates the codewords on the fly based on the received sequence of 6 bits. The CCK code mapper for the 5.5-Mb/s rate would use the two data bits to select one out of four codewords that are stored. The CCK code mapper for the 8.25-Mb/s rate could follow the same approach as that for the 5.5-Mb/s rate, i.e., storing 16 codewords and using 4 bits to select one of them. At the receiver side, the FWT is the only block that needs to be modified, as explained below, for the 8.25-Mb/s data rate. The Code Decision block, which selects the output with the maximum magnitude, is same as in the 11-Mb/s case except that we need to select 1 out of 16 in the 8.25-Mb/s rate as opposed to 1 out of 64 in the 11-Mb/s rate.

$$C_2' = V_2 C = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 + i & 1 + i & 1 - i & -1 + i & -1 + i & 1 - i & 1 + i & 1 + i \\ -1 - i & 1 + i & 1 - i & -1 + i & -1 + i & 1 - i & 1 + i & 1 + i \\ -1 - i & -1 - i & 1 - i & -1 + i & -1 + i & 1 - i & 1 + i & 1 + i \\ 1 + i & -1 - i & -1 + i & -1 + i & -1 + i & 1 - i & 1 + i & 1 + i \\ -1 - i & 1 - i & 1 + i & 1 + i & 1 + i & -1 - i & 1 + i & 1 + i \\ 1 - i & 1 - i & 1 + i & -1 - i & -1 - i & 1 + i & 1 + i & 1 + i \\ -1 - i & -1 - i & 1 + i & -1 - i & -1 - i & 1 + i & 1 + i & 1 + i \\ 1 - i & -1 + i & -1 - i & -1 - i & -1 - i & 1 + i & 1 + i & 1 + i \end{pmatrix}$$ (23)
Fig. 5 shows the Basic Fast Walsh Block (BFWB) for the 11-Mb/s data rate, where $\phi_2$ assumes values of 1, $j$, $-1$, and $-j$ in succession. $F$, $G$, and $H$ represent the first, second, and final stage outputs, respectively. That is, $F^s[q], G^s[q], \text{ and } H^s[q]$ denote $q$th output of the first, second, and final stages of BFWB, respectively, with $\phi_2 = s$. Fig. 6 shows the block level architecture of the FWT and Biggest Picker (BP) for the 11-Mb/s data rate—each BFWB generates 16 complex correlations between the input vector $b$ and 16 codewords, while BP selects the correlation with the maximum magnitude and outputs this correlation (as symbol) and its index.

Fig. 7 shows the architecture of FWT and BP for the 8.25-Mb/s data rate. This consists of two BFWBs of 11-Mb/s data rate and two stages of simple arithmetic operations (denoted as 8.25-Mb/s BFWB) involving 16 complex additions, as illustrated in Fig. 8. As shown in the figure, the inputs to the 8.25-Mb/s BFWB are obtained from the first stage outputs ($F^1[0, \ldots, 3]$ and $F^{-1}[0, \ldots, 3]$) of the two 11-Mb/s BFWBs with $\phi_2 = 1$ and $-1$. The 16 outputs of the 8.25-Mb/s FWT comprise eight outputs of the 8.25-Mb/s BFWB (which are correlations with the codewords in $C'_2$) and eight outputs from the two 11-Mb/s BFWBs (which are correlations with the codewords in $C$). Table IV shows the correspondence between the outputs of 11-Mb/s BFWBs with $\phi_2 = 1$ and $-1$ and the correlation outputs with codewords in $C$ required for the 8.25-Mb/s rate. In any implementation of 8.25 Mb/s FWT,
VI. SIMULATION RESULTS

The set up used for simulations is the Hellosoft WLAN 11g reference C Code that implements ERP-DSSS and ERP orthogonal frequency division multiplexing (ERP-OFDM) modes. The set up implements the transmitter and receiver, and the simulator for analog front-end (AFE) including models for radio frequency (RF) impairments and wireless channel. The transmitter and receiver blocks for the two modes are described here briefly.

A. Simulation Set Up

1) ERP-DSSS Mode:
- The transmitter for 5.5- and 11-Mb/s data rates of ERP-DSSS Mode of 11g is illustrated in Fig. 3. The CCK code mapping and DQPSK symbol mapping modules are as specified in the standard for 5.5- and 11-Mb/s data rates. For the 8.25-Mb/s data rate mode, 4 bits are fed to the CCK Code mapper, which selects 1 code out of 16 using these bits. Transmit filtering is implemented such that the signal at the output of the transmitter conforms to the transmit spectral mask specified in the standard.
- The receiver for 5.5- and 11-Mb/s data rates of ERP-DSSS Mode of 11g is illustrated in Fig. 4. The analog-to-digital converter (ADC) operates at 44 MHz. The output of the ADC is filtered and downsampled to 22 MHz. The combined matched filter and interpolator block includes CMF to cater to the 364-ns delay-spread channel. The FWT and the code decision (also referred to as biggest picker) blocks make up the CCK decoder. For the 8.25-Mb/s rate, the code decision block yields 4 bits; for 5.5 and 11 Mb/s, it yields 2 and 6 bits, respectively. DQPSK symbol decision yields 2 bits. Code and symbol decisions are used to generate the DFE output, which in turn is used to subtract the ISI from the succeeding symbol.

2) ERP-OFDM Mode:
- Fig. 9 shows the transmit block diagram of an OFDM transmitter as specified by the IEEE standard. Information bytes come from the Medium Access Controller to the Physical Layer Convergence Procedure, which in turn passes the bytes to the transmit bit-processing modules, namely scrambler, convolutional encoder, puncturing, and interleaver. Bits at the output of the interleaver are mapped to suitable constellation points, which modulate 48 subcarriers. Four pilot carriers are inserted for the purpose of clock and carrier tracking. Digital modulation of 52 subcarriers is performed through 64-point inverse fast Fourier transform (IFFT). Sixteen samples at sampling rate of $f_{st} = 20$ MHz are prepended to each OFDM symbol as a cyclic prefix (CP) to combat the ISI encountered in a delay-spread channel. These data are upsampled to 40 MHz, filtered, and passed through 10-bit digital-to-analog converters (DACs). Analog in-phase (I) and quadrature (Q) outputs from DACs are fed to the transmit AFE to amplify and up-convert to the RF passband.
- Fig. 10 shows the receiver block diagram. ADC operates at 40 MHz. The output of the ADC is filtered and downsampled to 20 MHz. The downsampled output is fed to the rotor block, which compensates for the overall carrier offset at the receiver. CP removal block takes out prepended CP samples. 64-point FFT generates the received constellation points, which are compensated by the frequency-domain equalizer (FDE). The output of the FDE, after being compensated for residual frequency offset (RFO) by the subcarrier derotation block (pilots are used to determine RFO), is passed on to the

the first stage outputs of the two 11-Mb/s BFWBs are needed for the 8.25-Mb/s BFWB to begin computations. In a power-efficient implementation, only those final stage 11-Mb/s BFWB outputs, which are shown in Table IV, are computed.

### TABLE IV

<table>
<thead>
<tr>
<th>Output</th>
<th>Correlation Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>$H^1[12]$</td>
<td>$(b, c_1)$</td>
</tr>
<tr>
<td>$H^1[14]$</td>
<td>$(b, c_2)$</td>
</tr>
<tr>
<td>$H^1[4]$</td>
<td>$(b, c_3)$</td>
</tr>
<tr>
<td>$H^0[6]$</td>
<td>$(b, c_4)$</td>
</tr>
<tr>
<td>$H^-1[12]$</td>
<td>$(b, c_5)$</td>
</tr>
<tr>
<td>$H^-1[14]$</td>
<td>$(b, c_6)$</td>
</tr>
<tr>
<td>$H^-1[4]$</td>
<td>$(b, c_7)$</td>
</tr>
<tr>
<td>$H^-1[6]$</td>
<td>$(b, c_8)$</td>
</tr>
</tbody>
</table>
Fig. 9. Hellosort transmitter blocks for 802.11g ERP-OFDM mode.

Fig. 10. Hellosort receiver blocks for 802.11g ERP-OFDM mode.

**TABLE V**

<table>
<thead>
<tr>
<th>RF Impairment</th>
<th>Magnitude/Level</th>
</tr>
</thead>
<tbody>
<tr>
<td>D.C. offset</td>
<td>0.5V (10% of ADC dynamic range)</td>
</tr>
<tr>
<td>IQ gain imbalance</td>
<td>1 dB</td>
</tr>
<tr>
<td>IQ phase imbalance</td>
<td>$3^\circ$</td>
</tr>
<tr>
<td>Phase Noise</td>
<td>$2^\circ$ (RMS)</td>
</tr>
</tbody>
</table>

constellation demapper. The output of the demapper, which maps constellation points to bits with associated confidence metrics, is de-interleaved and passed on to the Soft Viterbi decoder. The Viterbi decoder output is passed on to the descrambler followed by CRC-32 computation.

### B. Simulation Experiment

The channel is simulated in baseband (hence it is complex) at a sampling rate of 40 and 44 MHz for ERP-OFDM and ERP-DSSS, respectively. JTC indoor residential models [5] are used for testing the performance of different data rates in delay-spread channel. Channel realizations are random (different for each packet), with channel tap amplitudes Rayleigh distributed and phase uniformly distributed in 0 to $2\pi$. The mean power profile of the taps is given by the channel model specification. Channel taps are normalized so that the received signal strength, and hence the quiescent SNR point, does not fluctuate from realization to realization. The channel simulator also adds a frequency offset of $\alpha$ times the RF center frequency, where $\alpha$ is selected randomly between $\pm50$ ppm for each packet ([6] specifies a value of $\pm25$ ppm, and this applies to both transmitter and receiver front-ends). The receiver is assumed to have a noise figure of 10 dB. The transmitter output power level is at 20 dBm, as mentioned in the standard [6]. Other front-end impairments [14] that have been added are summarized in Table V. Each packet has three portions, namely preamble, header, and payload. The preamble portion is used by the receiver (during training mode) to estimate various parameters like automatic gain control (AGC) gain, symbol boundary, frequency offset, channel taps, etc., and these estimated parameters are employed in the data mode, where the actual payload is decoded according to the data rate and payload length information given in the header. The experiment was repeated 1000 times for each SNR, and the number of times the payload was decoded incorrectly was noted. These results are shown as plots of packet error rate (PER) as a function of path loss in decibels. The 8.25-Mb/s rate simulations are performed with code set $C \cup C'_2$; the results of simulations for this and other data rates (5.5 and 11 Mb/s of ERP-DSSS mode and 6 and 9 Mb/s of ERP-OFDM mode) are given in Figs. 11–13. The benign channel in Fig. 11 refers to the AWGN channel plus transmit and receive filters; Figs. 12 and 13 show the performance in JTC Residential channel models [5] (Residential A and B, respectively) with transmit and receive filters. The delay spread of Residential B (350 ns) is higher than that of Residential A (100 ns).

### C. Simulation Results

Figs. 11–13 show the PER versus path loss (in decibels) for different data rates for the benign, Residential A, and Residential B channels, respectively. Table VI gives the
Fig. 11. PER performance of various data rates in benign channel (AWGN channel + transmit and receive filters).

Fig. 12. PER performance of various data rates in JTC Residential A channel (delay spread of 100 ns).

Fig. 13. PER performance of various data rates in JTC Residential B channel (delay spread of 350 ns).

Table VI

<table>
<thead>
<tr>
<th>Channel</th>
<th>Data rate (Mbps)</th>
<th>Path Loss (dB)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>11</td>
<td>102.9</td>
</tr>
<tr>
<td></td>
<td>9</td>
<td>105.1</td>
</tr>
<tr>
<td></td>
<td>8.25</td>
<td>105</td>
</tr>
<tr>
<td></td>
<td>6</td>
<td>105.9</td>
</tr>
<tr>
<td></td>
<td>5.5</td>
<td>106.4</td>
</tr>
<tr>
<td></td>
<td>9</td>
<td>99.3</td>
</tr>
<tr>
<td>Residential A</td>
<td>8.25</td>
<td>103</td>
</tr>
<tr>
<td></td>
<td>6</td>
<td>104.9</td>
</tr>
<tr>
<td></td>
<td>5.5</td>
<td>104.9</td>
</tr>
<tr>
<td>Residential B</td>
<td>8.25</td>
<td>99.8</td>
</tr>
<tr>
<td></td>
<td>9</td>
<td>99</td>
</tr>
<tr>
<td></td>
<td>6</td>
<td>104.1</td>
</tr>
<tr>
<td></td>
<td>5.5</td>
<td>103</td>
</tr>
</tbody>
</table>

maximum path loss (obtained from the plots) at which PER remains below the standard specified 8% threshold for different data rates in different channels. The benign channel performance as seen in Fig. 11 is as expected: data rates of 5.5 and 6 Mb/s have the best performance, the 11-Mb/s rate has the poorest performance, and 8.25 and 9 Mb/s have performances in between. Note that since the performance of the 6-Mb/s rate is close to that of the 5.5-Mb/s rate in terms of information throughput, it is an alternative to the 5.5-Mb/s rate and not an intermediate data rate. Fig. 12 shows that in JTC Residential A channel, the performance of the 9-Mb/s rate is 1.7 dB poorer compared to that of the 8.25-Mb/s rate. Thus, in moderate delay spread, the 8.25-Mb/s rate is a better fallback option than the 9-Mb/s rate, although forward error correction is present in the 9-Mb/s data rate mode. The reasons are listed as follows:

- The correlation properties of the code set designed for 8.25 Mb/s make it highly robust against moderate delay spread.
- The 9-Mb/s data rate mode uses punctured convolutional coding (rate 1/2 is punctured to give rate 2/3), which tends to underperform in the presence of spectral nulls in the channel frequency response, thereby leading to higher bit errors.

The JTC Residential B performance, as shown in Fig. 13, shows that the 11-Mb/s rate suffers severe degradation in high delay spread, requiring a fallback to a lower data rate essential. This degradation is caused by the following:

- higher number of components in interference terms $I_1$ and $I_3$ [see (5) and (7), respectively] due to higher delay spread;
- higher magnitude of components in interference term $I_3$ due to poorer nonzero lag cross-correlation properties of the 64 codewords;
- higher magnitude of interference term $I_2$ [see (6)] due to poorer zero lag cross-correlation properties of the 64 codewords.

Among the aforementioned factors affecting the performance of the 11-Mb/s data rate, the first factor also affects the 8.25-Mb/s rate. In Residential B channel, the 8.25-Mb/s rate performs only 0.8 dB better compared to the 9-Mb/s rate, which has the
advantage of forward error correction. However, Residential A is a more commonly occurring channel than Residential B [5], and hence, in most cases of indoor delay spread, the 8.25-Mb/s rate is a better fallback option than the 9-Mb/s rate. Also, since the 8.25-Mb/s data rate mode is based on a spread spectrum technique, it has superior robustness against radio interference (e.g., from microwave ovens) that tends to affect the crowded and noisy 2.4-GHz spectrum [10] significantly.

**Comment 1:** The channel is simulated in baseband (hence it is complex) at a sampling rate of 44 MHz, and its output is down sampled to 22 MHz. JTC indoor residential models are used for testing the delay-spread channel performance for different data rates. Channel realizations are random (different for each packet), with the channel tap amplitudes Rayleigh distributed and the phase uniformly distributed in 0 to $2\pi$. The mean power profile of the taps is given by the channel model specification. Channel taps are normalized so that the received signal strength, and hence the quiescent SNR point, does not fluctuate from realization to realization. The experiment was repeated 1000 times for each SNR, and the number of times the payload was decoded incorrectly was noted. The SNR at which this number corresponds to nearly 8% [6] of the total number of experiments is taken as the required SNR.

The simulations are performed with code set $C \cup C'$, and the results are given in Table I. The benign channel in Table I refers to the AWGN channel plus the transmit and receive filters, and RES A and RES B refer to the JTC Residential channel models. The delay spread of RES B (350 ns) is higher than that of RES A (100 ns). Note from the results that the designed code set requires 2.3–10 dB less SNR when compared to the 11-Mb/s data rate, and only 1.7 to 3.1 dB more SNR when compared to the 5.5-Mb/s data rate, in benign to large delay-spread channel environment.

**VII. INTEROPERABILITY ISSUES FOR 8.25-Mb/s DATA RATE**

The 8.25-Mb/s data rate may not be made a mandatory feature in the latest 802.11b/g standard. However, inclusion of the 8.25-Mb/s data rate as a proprietary rate in either the station or the Access Point (AP) will in no way disturb interoperability, as explained below.

An 802.11 WLAN comprises several Basic Service Sets (BSSs) each controlled by an AP. When a station wants to access an existing BSS, it first needs to get synchronization information from the Beacon Frames transmitted by the AP. After this, it has to go through the Authentication Process and then the Association Process. Association is the process of exchange of information about the stations and BSS capabilities. The capability of the stations and the AP to support the 8.25-Mb/s rate would be known during the Association Process. If any one of them does not have 8.25-Mb/s rate provision, they will fall back to the 5.5-Mb/s rate. If both of them have the capability to support the 8.25-Mb/s rate, then the intermediate fallback option would be used.

An 802.11 WLAN may operate in an ad hoc mode, where there is no AP, and information is transferred directly between peer stations. In such a mode, two stations having 8.25-Mb/s rate capability can use this rate as an intermediate fallback option.

**VIII. CONCLUSION**

We presented a design methodology for a complementary code set for an intermediate data rate between 5.5- and 11-Mb/s rates, which performs well in benign as well as large delay-spread channels. In the present 802.11b system, when the 11-Mb/s data rate is not sustainable, one has to fall back to a much lower data rate of 5.5 Mb/s, and this would mean sacrificing throughput to a large extent. Even in a benign channel case, other impairments like adjacent channel interference and sampling clock offset may limit the performance of the 11-Mb/s data rate to such an extent that most of the time one may fall back to the 5.5-Mb/s data rate. Under such situations, the 8.25-Mb/s data rate option would increase the overall network throughput. In designing the code set for the new data rate, we kept in mind that the additional resources required for including this in the existing 802.11b transceiver is minimal. Since the 802.11 b/g chipsets are widely available now, and they have intermediate data rates of 6 and 9 Mb/s, we have compared the performance of the 8.25-Mb/s rate with that of 6 and 9 Mb/s. The results show that the 8.25-Mb/s rate is a better fallback option when the 11-Mb/s rate is not sustainable.

**ACKNOWLEDGMENT**

S. Batabyal would like to thank K. Barman of Hellosoft for the helpful discussions. This work was done when B. K. Dey and V. U. Reddy were with the International Institute of Information Technology, Hyderabad, India.

**REFERENCES**

Subhendu Batabyal received the Engineering degree from the Indian Institute of Technology, Kharagpur, India, in 1999 and the M.S. degree in communication systems and signal processing from the International Institute of Information Technology, Hyderabad, India, in 2005.

He has held various engineering positions at Sasken Communication Technologies, Ubisense India Ltd., and Hellosoft India Ltd. He is presently with The MathWorks, Inc., Natick, MA. His research interests are in OFDM, wireless communications, and fixed point signal processing.

Mr. Batabyal received the Jagadish Bose National Science Talent Search Encouragement Award in 1995.

Bikash Kumar Dey (S’00–M’04) received the B.E. degree in electronics and telecommunication engineering from Bengal Engineering College, Howrah, India, in 1996, the M.E. degree in signal processing from the Indian Institute of Science, Bangalore, India, in 1999, and the Ph.D. degree in electrical communication engineering from the Indian Institute of Science, in 2003.

From August 1996 to June 1997, he was with Wipro Infotech Global R&D. In February 2003, he was with Hellosoft India Pvt. Ltd., Hyderabad, as a Technical Member. In June 2003, he was with the International Institute of Information Technology, Hyderabad, as an Assistant Professor. Since May 2005, he has been with the Department of Electrical Engineering, Indian Institute of Technology (IIT), Bombay, India, as an Assistant Professor. His research interests include error correcting codes, signal processing, and wireless communications.

Dr. Dey was awarded the Prof. I. S. N. Murthy Medal from IISc as the best M.E. student in the Department of Electrical Communication Engineering and Electrical Engineering for 1998–1999 and the Alumni Medal for best Ph.D. dissertation in the division of electrical sciences for 2003–2004.

V. Umamaheswar Reddy (SM’82–F’99–LF’05) received the B.E. degree in telecommunication engineering from the Osmania University, Hyderabad, India, in 1962, the M.Tech. degree in electronics and electrical communication engineering from the Indian Institute of Technology, Kharagpur, India, in 1963, and the Ph.D. degree in electrical engineering from the University of Missouri, Columbia, in 1971.

He was with the Indian Institute of Technology (IIT), Madras, India; IIT, Kharagpur; Osmania University, and Indian Institute of Science (IISc), Bangalore, India. At Osmania University, he established the Research and Training Unit for Navigational Electronics (he was its Founding Director). After retiring from IISc in 2001, he was with Hellosoft India Pvt. Ltd., Hyderabad, as CTO. In June 2003, he moved to the International Institute of Information Technology, Hyderabad, as Microsoft Chair Professor. He held several visiting appointments with Stanford University, Stanford, CA, and the University of Iowa, Iowa City. Since December 2005, he has been with Hellosoft as the Chief Scientist. His areas of research have been adaptive and sensor array signal processing. During the last ten years, he has been focusing on the design of OFDM-based physical layer with applications to DSL, WLAN, and WiMax modems.

Dr. Reddy is a Fellow of the Indian Academy of Sciences, Indian National Academy of Engineering, and Indian National Science Academy. He received the S.K. Mitra Memorial Award from IETE (India) in 1989. He was on the editorial boards of the Indian Journal of Engineering and Materials Sciences and PROCEEDINGS OF THE IEEE. He was the Chairman of the Indian National Committee for International Union of Radio Science (URSI) during 1997–2000.