Multistream faster than Nyquist signaling

Rusek, Fredrik; Anderson, John B

Published in:
IEEE Transactions on Communications

DOI:
10.1109/TCOMM.2009.05.070224

2009

Link to publication

Citation for published version (APA):

General rights
Copyright and moral rights for the publications made accessible in the public portal are retained by the authors and/or other copyright owners and it is a condition of accessing publications that users recognise and abide by the legal requirements associated with these rights.

- Users may download and print one copy of any publication from the public portal for the purpose of private study or research.
- You may not further distribute the material or use it for any profit-making activity or commercial gain.
- You may freely distribute the URL identifying the publication in the public portal.

Take down policy
If you believe that this document breaches copyright please contact us providing details, and we will remove access to the work immediately and investigate your claim.
Multistream Faster than Nyquist Signaling

Fredrik Rusek and John B. Anderson
Electrical and Information Technology Dept. and
Swedish Strategic Center for High Speed Wireless Communication
Lund University, Box 118, 221 00 Lund, Sweden

Abstract—We extend Mazo’s concept of faster-than-Nyquist (FTN) signaling to pulse trains that modulate a bank of subcarriers, a method called two dimensional FTN signaling. The signal processing is similar to orthogonal frequency division multiplex (OFDM) transmission but the subchannels are not orthogonal. Despite nonorthogonal pulses and subcarriers, the method achieves the isolated-pulse error performance; it does so in as little as half the bandwidth of ordinary OFDM. Euclidean distance properties are investigated for schemes based on several basic pulses. The best have Gaussian shape. An efficient distance calculation is given. Concatenation of ordinary codes and FTN are introduced. The combination achieves the outer code gain in as little as half the bandwidth. Receivers must work in two dimensions, and several iterative designs are proposed for FTN with outer convolutional coding.

I. INTRODUCTION

Consider baseband signals of the form

\[ s(t) = \sqrt{2E_a/T} \sum_n a_n h(t - nT), \]

in which \( a_n \) are data values over an \( M \)-ary alphabet and \( h(t) \) is a unit-energy baseband pulse. This simple form underlies QAM, TCM, and the subcarriers in orthogonal frequency division multiplex (OFDM), as well as many other transmission systems. In these schemes, \( h(t) \) is a \( T \)-orthogonal pulse, meaning that the correlation \( \int h(t - nT)h^*(t - mT)dt \) is zero, \( m \neq n \). In 1975 Mazo [1] pointed out that binary sinc\((t/T)\) pulses in (1) could be sent every \( T_s \) seconds, \( T_s < T \), without loss in asymptotic error probability. This he called faster than Nyquist (FTN) signaling, because the pulses appear faster than allowed by Nyquist’s limit for orthogonal pulses. FTN signaling has since been generalized in a number of ways.

This paper extends the FTN concept to the frequency dimension. This extension to two dimensions opens up a number of attractive possibilities. Many signals of type (1) are stacked in frequency through modulation by a set of carriers at frequencies \( f_0 + \{f_k\} \) to form the in-phase and quadrature (I/Q) signal given by the real part of

\[ s(t) = \sqrt{2E_a/T} \sum_{k=0}^{K-1} \sum_{n=0}^{N-1} a_{k,n}^I h(t - nT) e^{j2\pi(f_0 + f_k)t}. \]

This is a superposition of \( 2NK \) linear carrier modulations, and it carries \( 2NK \) data values. The \( K \times N \) matrix \( A = \{a_{k,n}\} \) is called the data matrix and consists of the complex data values \( a_{k,n} = a_{k,n}^I + j a_{k,n}^Q \). The \( K \) rows in this matrix correspond to subcarriers, and the \( N \) columns to pulse positions. If \( f_k = k f_\Delta \), \( k = 0, 1, \ldots, K-1 \) and \( f_\Delta \) is equal to the single-sided bandwidth of \( h(t) \), the \( 2K \) carrier signals are orthogonal; if \( h(t) \) is \( T \)-orthogonal, all \( NK \) pulses are mutually orthogonal. In OFDM, \( h \) is ordinarily the width-\( T \) square pulse, the subcarriers are \( 1/T \)-spaced frequency orthogonal, and all pulses are again mutually orthogonal.

The signal design in the sinc and OFDM cases is thus based on orthogonality. According to theory, there exist about \( 2W^2 \) orthogonal signals in \( W \) positive Hertz and \( \pi \) seconds. Data values that modulate the amplitude of each can be maximum-likelihood detected independently, and therefore about \( 2W^2 \) symbols can be transmitted this way. Take for example \( h(t) = \sqrt{T/T_s} \text{sinc}(t/T), \) and measure time and bandwidth of (2) by some reasonable method (such as 99% power bandwidth). Then as \( K \) and \( N \) grow, \( \pi N^2 \rightarrow T, W = K f_\Delta = K/T, \) and the product \( 2W^2 \) tends in ratio to \( 2(K/T)(NK) = 2KN; \) thus Eq. (2) carries as many data values for large \( NK \) as any scheme based on orthogonality can carry. A similar outcome occurs when \( h \) is a root RC pulse. For a given number of symbols carried by (2), \( T \) may be varied, which trades off \( W \) against \( \pi \). \( N \) may also be traded against \( K \). The time–bandwidth product is unaffected, and (2) always carries about twice \( W^2 \) symbols.

If the aim is to achieve the error rate of a stacked orthogonal-signal system (2), without necessarily using orthogonal signals, the story is more complex, and more can be achieved. By error rate is meant the error probability of the maximum likelihood sequence estimation (MLSE) receiver when \( h \) is employed in (1) with additive white Gaussian noise (AWGN) of density \( N_0/2 \) in the channel. As the signal-to-noise ratio \( E_b/N_0 \) grows, the probability of incorrect detection of an \( a_n \) is asymptotically \( P_e \sim Q(\sqrt{d_{\text{min}}^2 E_b/N_0}) \), where \( d_{\text{min}} \) is the minimum distance of the set of signals and \( d_{\text{min}} \leq d_{\text{MF}} \). Here \( E_b = E_t / \log_2 M \), \( E_t \) is the average symbol energy, and \( d_{\text{MF}} \) is the matched-filter bound distance for the data alphabet. \( d_{\text{MF}} \) measures the performance of simple orthogonal-pulse signaling with the same data values. The paper will concentrate on the binary case, for which \( d_{\text{MF}}^2 = 2 \), so the target orthogonal-pulse error rate is \( Q(\sqrt{2E_b/N_0}) \). If the \( K \) I/Q signals in (2) do not overlap in frequency, the same asymptotic error rate applies there.

Achieving more at the same \( E_b \) and error probability means that FTN signals need to consume less bandwidth. We need to define it carefully. With independent and identically distributed (IID) data symbols the power spectral density (PSD) of the \( k \)th subcarrier \( S_k(f) \) is proportional to

\[ (H(f - k f_\Delta) - f_0)^2 + (H(f + k f_\Delta) + f_0)^2, \]

where \( H(f) \) is the Fourier transform of \( h(t) \). With \( K \) subcarriers, the total PSD satisfies (take positive \( f \) only)

\[ S(f) \propto \sum_{k=0}^{K-1} S_k(f) = \sum_{k=0}^{K-1} |H(f - k f_\Delta - f_0)|^2, \]

where \( k f_\Delta \). The \textit{normalized time–bandwidth product} (NTB) of this
What total time–bandwidth product is occupied by the transmission? The question may be approached in several ways. Spectral and temporal sidelobes interfere with adjacent users of frequency and time, and contribute to the signal’s occupancy. For a moderate NK product, a packet of say 100–10000 bits, the sidelobes make a significant contribution. In a companion paper [6] we treat this contribution, seek to minimize it, and find rather different outcomes than given here. In this paper we let N and K grow large, so that the sidelobes are insignificant. The lattice area is about $NKf_{\Delta}T_{\Delta}$ Hz-s/bit and the NTB tends to $f_{\Delta}T_{\Delta}$ Hz-s/bit. The ratio of N and K can be changed at will so long as they have the same product. The orthogonal binary sinc$(t/T)$ pulse case has $f_{\Delta} = 1/T$ and $T_{\Delta} = T$, and the NTB is $f_{\Delta}T_{\Delta} = 1$ Hz-s/bit. This provides a useful benchmark for other pulses and systems. Since the value of T does not change the NTB, we henceforth take $T = 1$.

When $f_{\Delta}$ is less than the subcarrier bandwidth, the signal interrelations that produce $d_{\text{min}}$ work in new ways and the distance structure is time varying. Analytical results are available only in special cases [15], [16]. Finding $d_{\text{min}}$, in this new situation is challenging but possible. The subject of minimum distance is taken up in Section II. Distance studies with various pulses $h(t)$ show that $d_{\text{min}}^2 = 2$ can occur at 0.5 Hz-s/bit, i.e., half the sinc benchmark. The section also gives a number of properties. Euclidean distance computations appear in Appendix 1. An advanced algorithm to find minimum distance appears in the Appendix 2.

Section III proposes several MFTN receiver designs and introduces concatenation of convolutional codes with MFTN. An iterated receiver works particularly well here, and these concatenations form a successful practical application of MFTN.
II. SIGNALS, PROPERTIES AND MINIMUM DISTANCE

The real part of Eq. (2) is

\[ \sqrt{2E_s/T} [I(t) \cos 2\pi f_0 t - Q(t) \sin 2\pi f_0 t] \]

where the in-phase and quadrature signals \( I(t) \) and \( Q(t) \) are

\[
I(t) = \sum_{k=0}^{K-1} \sum_{n=0}^{N-1} [a_{k,n}^I h(t - nT) \cos 2\pi f_k t - a_{k,n}^Q h(t - nT) \sin 2\pi f_k t]
\]

\[
Q(t) = \sum_{k=0}^{K-1} \sum_{n=0}^{N-1} [a_{k,n}^Q h(t - nT) \cos 2\pi f_k t + a_{k,n}^I h(t - nT) \sin 2\pi f_k t].
\]  

Equation (5) as well be written

\[
\int |\Delta I(t) - \Delta Q(t)|^2 \, dt
\]

as \( f_0 \to \infty \), in which \( \Delta I(t) \) and \( \Delta Q(t) \) are as in (5) with \( \Delta A \) instead of \( A \). For binary signaling the elements in \( \Delta A \) take values in \( \{ \pm 2 \pm 2j, \pm 2, \pm 2j \} \).

An error event is a region of nonzero difference components that begins at some position \((n, k)\), which we may as well take as \((0, 0)\). Without loss of generality, we can restrict \( \Delta a_{0,0} \) to \( \{2, 2 + 2j\} \). The minimum distance \( d_{\text{min}} \) of a signal set is the minimum of (7) over all such events, and an event leading to \( d_{\text{min}} \) is called a critical event. Since there are very many error events, finding \( d_{\text{min}} \) is difficult, but it is possible to find reliable estimates (i.e., tight overbounds) by searching over limited sets of events. In this paper the critical error events are typically of size \( 3 \times 3 \) or smaller. Distance may be computed by direct integration of the difference matrix, but a much more efficient method is based on autocorrelations of \( h(t) \); this is given in Appendix 1.

An important property of MFTN distance is that it depends on the start time of the event. That is, for a given error event \( \Delta A \) whose first column corresponds to pulses centered at \( t = 0 \), the distance will change if the event starts with pulses centered at \( t = nT_\Delta \). The fundamental reason is that the pulse rate is not in general synchronized with the subcarrier frequencies; it is mathematically seen in the derivations in Appendix 1. An example of the phenomena appears in Figure 3 for a 30\% root RC pulse, \( T_\Delta = 7 \), \( f_\Delta = .8 \), and the event

\[
\Delta A = \begin{bmatrix} 2 & -2 + 2j & -2j \\ -2 & 2 + 2j & -2j \end{bmatrix}.
\]  

The figure plots square distance against the event start time \( t_0 \). Since the plot repeats every \( 1/f_\Delta = 1.25 \) s, the time axis can be taken as \( t_0 \mod 1/f_\Delta \). Dots show this event’s distance when it starts at times \( t_0 = 0, .8, 1.6, \ldots \), which lead mod 1.25 to all the multiples of .05. If \( 1/f_\Delta T_\Delta \) is not a rational number, then in principle starts at all \( t_0 \mod 1/f_\Delta \) in \( [0, 1/f_\Delta] \) are possible, and the worst case distance is the minimum, which is 1.13 at start \( t_0 \mod 1/f_\Delta \approx 24 \).
The worst-case distance is the minimum of the allowed time points. Synchronism can thus lead to a better distance, but in reality there is little gain unless \( i \) and \( \ell \) are very small integers. In the figure, with 25 points, there is virtually no distance gain. But the MFTN case shown in Figure 2 is a synchronous one with \( f_\Delta T_\Delta = 1/2 \); the subcarrier structure repeats precisely every \( 2T_\Delta \). With error event (9), the distance from start time \( t_0 = T_\Delta \) or \( 2T_\Delta \) are both 8.72. These avoid the worst-case start time, which is 0.25, with the much poorer distance 0.77. (However, there exist other error events with distance less than 2, so this MFTN is beyond the Mazo limit). Synchronism can thus be of benefit when the MFTN parameters allow it, but only a few cases of synchronous FTN are worth reporting. We will ordinarily take distance to be the minimum of the continuous distance versus \( t_0 \) curve.

![Distance versus event start time plot](image)

**Fig. 3.** Distance versus event start time \( t_0 \) modulo \( 1/f_\Delta \), for \( f_\Delta = .8 \) and \( T_\Delta = .7 \) with error event (9).

When not more than two carriers overlap in spectrum, the curve in Figure 3 can be shown to be a sinusoid, and its minimum can be computed easily from any three points. Another simplifying property is that very few error events lead to a distance near \( d_{\text{min}} \). As well, groups of symmetric events, typically 4–8 in number, have identical distance for the entire range of \( T_\Delta \) and \( f_\Delta \). We call these event families.

The results presented in this paper in fact stem from only 20 families.

**Delayed Pulses.** Subjecting subcarriers to different delays has the potential to improve minimum distance because pulses can come into more favorable alignments with each other and with the sines and cosines in the I and Q signals in (5). Many ways to execute the delays can be imagined. A simple but effective way is to delay the I and Q pulses \( h(t-nT_\Delta) \) in carriers 0, 1, \ldots, \( K-1 \) by \( \delta_0, \delta_1, \ldots, \delta_{K-1} \), where each \( \delta \) satisfies \( 0 \leq \delta < T_\Delta \). Delaying all \( K \) carrier pulses by the same \( \delta \) is the same as a time shift to the subcarrier system; this can prevent a subcarrier in (2) from passing through zero at the moment when a pulse is largest, which can severely reduce distance. Pulse trains \( k = 0, 1, \ldots \) can be delayed by linearly growing amounts, e.g., \( \delta_0, \delta_1, \ldots = (0, 2, 4, \ldots)/T_\Delta \). A particularly successful scheme is delays of the form \( .5, .5, 0, 0, \ldots \) times \( T_\Delta \). Instead of pulse delays, schemes can be based on delaying the subcarriers by fractions of their own period \( 1/f_\Delta \).

**The Mazo Limit for Root RC and Gaussian Pulses.** Figures 4 and 5 plot the outcome of a search for non-synchronous combinations of \( f_\Delta \) and \( T_\Delta \) that have least product. Dotted lines show contours of constant \( f_\Delta T_\Delta \) product. Figure 4 plots the case for the Gaussian pulse

\[ h(t) = \sqrt{1/2\pi}\exp(-t^2/2\sigma^2) \]

normalized, with \( \sigma^2 = .399^2 \). The minimum distance searches here are over all start times in the error events. Overlapping curves show the trajectory for each critical event family; the “northeast”-most of all curves is the Mazo limit. Consider the trajectory for one critical error sequence, with distance \( d^2 \). As \( T_\Delta \) drops, the \( f_\Delta \) needed to maintain \( d^2 < 2 \) rises, creating, typically, a convex-up \( f_\Delta, T_\Delta \) relationship. Eventually, time compression alone prevents \( d^2 = 2 \); no \( f_\Delta \) allows it, and the result is a horizontal section at the lower right end of the convex section (this is most visible in Fig. 5). At the upper left of a convex section, \( T_\Delta \) is large and it can happen that no \( f_\Delta \) leads to \( d^2 < 2 \). The section simply stops at some \( f_\Delta, T_\Delta^2 \) (square blocks mark two such points in Fig. 4).

If this section is part of the ultimate Mazo limit, then at \( T_\Delta = T_\Delta^2 \) the Mazo limit must move horizontally left to another event’s convex section.

Note that \( h(t) \) here is not orthogonal for any \( T \). The Gauss pulse has important properties when simultaneous time and frequency side lobes are important [6].

![Mazo limit plot](image)

**Fig. 4.** Estimated position of the two-dimensional Mazo limit for non-synchronous binary Gaussian pulse signaling. Dashed line shows limit with alternate pulse trains delayed 0.5 symbol. Each of the five component curves here represents an event family. Dotted curves are contours of constant \( f_\Delta T_\Delta \).

Figure 5 plots the non-synchronous Mazo limit for 10 and 30% root RC pulses, plus the 10% case when alternate pulse trains are delayed by \( T_\Delta/2 \) (dashed). The searches are again over all start times in the error events. It can be seen that the least product for 30% is about 0.60, at \( (f_\Delta \approx .67, T_\Delta \approx .88) \); for 10% pulses this improves to product 0.556 at \( (f_\Delta \approx .660, T_\Delta \approx .843) \). The delays improve the 10% case to 0.534 at \( (f_\Delta \approx .66, T_\Delta \approx .80) \); the 30% pulse is similarly improved by delays. These products are excellent but we have found a few synchronous 10% cases, with \( T_\Delta \) in the range 0.78–0.9, for which \( f_\Delta T_\Delta = 1/2 \). This is a doubling of the spectral efficiency of the sinc benchmark and OFDM.

The estimated minimum distance of all combinations is 2. Searches are generally performed with the method in Appendix 2 and are over error events out to size \( 4 \times 7 \)
A trellis description of the system consists of $4^{KL}$ states, where $L$ is the support of the model of the FTN-induced intersymbol interference (ISI). Clearly, reduced complexity methods have to be used, and in order to avoid simply trading the bandwidth reduction for a higher symbol energy, the receiver must achieve essentially the full MLSE performance. Reduced complexity detection methods appear in [8], [9], [10], [17], but these are all operating too far from MLSE performance to fully exploit the bandwidth improvement. In [5] a simple $M$-algorithm is considered, but it turns out that the method only works for 2–4 subcarriers. In [14] a more sophisticated iterative detector is proposed. The detector’s BER at high signal to noise ratio closely follows the basic reference $P_e \approx Q(\sqrt{E_b/\gamma_0})$. Thus the distance computations from Section II are verified. The complexity, however, is still significant; the detector consists of two steps, each involving 6–8 iterations. In brief, detection is possible, but with high complexity.

If an outer code is concatenated via an interleaver to the MFTN system, detection is much simpler and a more practical MFTN system is the outcome. We will now focus on this case and take the outer code as a rate 1/2 feedforward convolutional code; such codes at low memory have 4–5 dB coding gain. Just as an uncoded system from Section II was able to preserve its BER down to a certain critical compression product $f_\Delta T_\Delta$, a concatenated system can preserve the coding gain of the outer code down to another product. It will turn out that this critical product is in fact smaller than the one for an uncoded system. Thus, the potential of MFTN increases for concatenated systems, and this is a major reason to focus on them. Concatenated coding with a two dimensional ISI channel has been investigated e.g. in [11], [18], [19], but not with MFTN-induced ISI.

The system model consists of the sequence: Convolutional Code $\rightarrow$ Interleaver $\rightarrow$ Binary to K Stream Mapper $\rightarrow$ MFTN Modulator. The memory of the convolutional code is $\nu$. A sequence of $10000 - \nu$ IID information bits are first convolutionally encoded; this produces a codeword $\nu$ of length 20000, which is randomly interleaved to produce $\nu'$. Mapping from $v_1, v_{k+1}$ to one symbol $a_{k,n} \in \{ \pm 1 \pm j \}$ follows, with $k$ and $n$ found from $l$ by some predefined pattern. The pattern is not of interest because of the interleaver, and will not be discussed further. The transmitted signal $s(t)$ is formed according to (2) from the symbol sequence $\alpha$; we will use 20 subcarriers, i.e. $K - 20$ and thus $N = 500$. We have also performed tests with fewer subcarriers. When the number increases, the BER generally degrades. But at 5–8 subcarriers a saturation of the BER occurs, that is, the BER for $K = 10$ is virtually identical to the BER for $K = 20$ presented here. We are therefore confident that the BER does not change if many subcarriers are used, e.g. $K = 64$ or 256.

Due to the interleaver, straightforward iterative detection is possible, and a block diagram is given in Figure 6. Two soft-input soft-output detectors are needed, one for the convolutional encoder and one for the MFTN signaling system, which together with the mapper is considered as an inner encoder. Hereafter, the detector for the MFTN system is referred to as a detector and the outer detector as a decoder. A standard full BCJR algorithm will be used as decoder for the convolutional code, but the MFTN detector is not standard. If optimum detection is desired, the $4^{KL}$ state complexity still comes about, and here also a reduced complexity method is needed. The difference from the uncoded case is that the output from the reduced complexity detector there is the final result, while it feeds an iterative process in concatenated systems. The outer and the inner detectors jointly work their way to a near-optimal result; the fact that the inner detector is not full complexity more or less only slows down the convergence speed. The ultimate compression product limits, however, will be somewhat worse due to the reduced complexity.

We now turn to the structure of the MFTN detector. The baseband representation of (2) is

$$s_{bh}(t) = \sum_n \sum_k a_{k,n} h(t - nT_\Delta)e^{j2\pi f_k t},$$  \hspace{1cm} (10)

The detector encounters a noisy signal $r(t) = s_{bh}(t) + n(t)$, where $n(t)$ is complex-valued baseband white Gaussian noise. The first step in the receiver is to project $r(t)$ onto the basis functions $h(t - nT_\Delta)e^{j2\pi f_k t}$, i.e. to compute

$$R_{bh,n} = \int_{-\infty}^{\infty} r(t)h^*(t - nT_\Delta)e^{-j2\pi f_k t} dt,$$  \hspace{1cm} (11)

The elements $R_{bh,n}$ form the matrix $R_{bh}$, which can in practice be efficiently implemented by a bank of matched filters with rate-1/T_\Delta sampling. The matrix $R_{bh}$ comprises a sufficient statistic for estimating $\{a_{k,n}\}$, and in the sequel $R_{bh}$ is simply
called the received signal. We may write \( R = S + N \). The sent part \( S \) of \( R \) equals
\[
S_{k,n} = \sum_{m,l} a_{l,m} h(t-mT_{\Delta}) e^{2\pi i t(k-m)\Delta} h^*(t-nT_{\Delta}) dt.
\]
(12)
Further manipulations of (12) are made in Appendix 1. The noise matrix \( N \) is given by
\[
N_{k,n} = \int_{-\infty}^{\infty} n(t) h^*(t-nT_{\Delta}) e^{-2\pi i t(nT_{\Delta})} dt.
\]
(13)
The variables \( N_{k,n} \) are not white.

The goal of the detector is to maximize the a posteriori probability (APP) of an individual bit, i.e.
\[
a^{1/Q}_{k,n} = \arg \max_a \Pr \left( a^{1/Q}_{k,n} = a | R \right).
\]
(14)
(Superscript \( I/Q \) means “I respectively Q”; when this superscript is omitted we intend a complex \( a \).) Instead of working with probabilities it is convenient to work with log-likelihood ratios (LLRs)
\[
L \left( a^{1/Q}_{k,n} \right) = \log \frac{\Pr \left\{ a^{1/Q}_{k,n} = 1 \right\}}{\Pr \left\{ a^{1/Q}_{k,n} = -1 \right\}}.
\]
(15)
From \( L(a^{1/Q}_{k,n}) \) it is straightforward to find the LLRs \( L(v_i) \). Since the data symbols are independent we can as usual express the conditional LRL \( L(a^{1/Q}_{k,n} \mid R) \) as
\[
L \left( a^{1/Q}_{k,n} \mid R \right) = L_{\text{ext}} \left( a^{1/Q}_{k,n} \mid R \right) + L \left( a^{1/Q}_{k,n} \right)
\]
(16)
where \( L_{\text{ext}}(a^{1/Q}_{k,n} \mid R) \) denotes the extrinsic information about \( a^{1/Q}_{k,n} \) contained in \( R \).

The true APPs of the data bits can be found by a multidimensional BCJR algorithm, but its complexity grows exponentially with \( K \), and the APPs have to be approximated by simpler means. The detector will thus consider only a fraction of the symbols at once; the rest will act as noise. The symbols \( \{ a_{k,n} \} \) are grouped into two sets, \( A_{\text{dec}} \) and \( A_{\text{int}} \); symbols in \( A_{\text{dec}} \) are the ones that we try to decode at the moment and those in \( A_{\text{int}} \) act as noise. The transmitted baseband signal \( s_{\text{bb}}(t) \) can be expressed as \( s_{\text{bb}}(t) = s_{\text{dec}}(t) + s_{\text{int}}(t) \), where \( s_{\text{dec}}(t) \) and \( s_{\text{int}}(t) \) are the contributions from symbols in \( A_{\text{dec}} \) and \( A_{\text{int}} \), respectively. The detector will be based on successive interference cancellation [13]; when decoding the signal \( s_{\text{dec}}(t) \) a soft estimate \( \hat{s}_{\text{int}}(t) \) of \( s_{\text{int}}(t) \) is formed based on soft information about all symbols in \( A_{\text{int}} \):
\[
\hat{s}_{\text{int}}(t) = \sum_{\delta} b^{1/Q}_{k,n} h(t-nT_{\Delta}) e^{2\pi i t\Delta f_{\delta}},
\]
(17)
where \( B = \{(k,n) : a_{k,n} \in A_{\text{int}}\} \) and \( b^{1/Q}_{k,n} \) are the soft estimates of \( a^{1/Q}_{k,n} \) defined by
\[
b^{1/Q}_{k,n} = \Pr \{ a^{1/Q}_{k,n} = 1 \} - \Pr \{ a^{1/Q}_{k,n} = -1 \} = \tanh \left( \frac{1}{2} L_{\text{ext}} \left( a^{1/Q}_{k,n} \mid R \right) \right).
\]
Then \( \delta_{\text{int}}(t) \) is projected onto the basis functions, with the projection denoted \( \hat{S}(A_{\text{int}}) \). Finally, the tentative received signal when detecting symbols in \( A_{\text{dec}} \) is formed as
\[
\hat{R}(A_{\text{dec}}) = R - \hat{S}(A_{\text{int}}).
\]
(18)
Together with the extrinsic information \( L_{\text{ext}}(a^{1/Q}_{k,n} \mid \hat{R}(A_{\text{dec}})) \) about the symbols \( a^{1/Q}_{k,n} \) in \( A_{\text{dec}} \), the signal \( R(A_{\text{dec}}) \) is fed to some detection algorithm.
If \( A_{\text{dec}} = \{a_{k',n}, \forall n\} \) and \( A_{\text{int}} = \emptyset \) there is no complexity reduction at all. We will only consider the partition \( A_{\text{dec}} = \{a_{k',n}, \forall n\} \); this is all symbols on carrier \( k' \). For this partition, the signal \( \delta_{\text{dec}}(t) \) can be viewed as a single carrier signal, based on a real-valued pulse shape \( h(t) \). It then follows that its real- and complex-valued parts can be detected independently; it is thus a matter of detecting binary symbols through an ISI channel in AWGN. If the detector takes \( L \) ISI taps as being significant, the complexity of a full BCJR is \( 2^L \). In this paper the full BCJR will be used, with \( L \leq 5 \), and thus there are at most 32 states in the BCJR detector. One technical detail remains. The classical BCJR requires white noise \( N \), which is not the case here. Since \( h(t) \) has long duration, in theory infinite, whitening of the outputs from the sampled matched filters is a difficult task. However, an equivalent algorithm to the BCJR that operates directly on the samples of the matched filter output has been derived in [12]. This algorithm assumes colored noise and is not a true BCJR, but it has the same output as a true BCJR acting on a whitened version of \( R \), and we still refer to it as a BCJR.

The innermost part of Figure 6 can now be drawn. It is shown for subcarrier \( k' \) in Figure 7. Some words on the partition \( A_{\text{dec}} = \{a_{k,n}, \forall n\} \) are needed. If \( T_{\Delta} = T \), there is no ISI, and the BCJR in Figure 7 becomes meaningless, since there is no dependence along each subchannel in the signal \( R(A_{\text{dec}}) \). If \( T_{\Delta} \) is close to 1, but \( f_{\Delta} \ll 1 \), there is some ISI, but the ICI is much stronger. In that case it makes sense to use the partition \( A_{\text{dec}} = \{a_{k,n}, \forall k\} \), that is, to attack the ICI only. For \( f_{\Delta}, T_{\Delta} \) of roughly the same size, hybrid methods can be used; some iterations can use \( A_{\text{dec}} = \{a_{k,n}, \forall n\} \) and some can use \( A_{\text{dec}} = \{a_{k,n}, \forall k\} \). Brevity prevents pursuing this idea. The partition \( A_{\text{dec}} = \{a_{k,n}, \forall n\} \) turns out to work quite well in this paper.

Let us turn to actual receiver tests. The outer code was taken as the (7,5) convolutional code, and \( h(t) \) was a 30%
root RC pulse. The number of receiver iterations is limited to 7. In each receiver test, 1000 blocks of 10000 information bits have been detected.

In Figure 8, three parameter combinations are tested, \( f_\Delta = 1.174, T_\Delta = .46 \) (product .54), \( f_\Delta = .5682, T_\Delta = .88 \) (product .5) and \( f_\Delta = .75, T_\Delta = .60 \) (product .45). Also shown is the performance of the convolutional code over an ISI-free channel. As seen from the figure, at high \( E_b/N_0 \), the encoded MFTN signaling systems are in fact able to retain the full coding gain of the outer code, but at a much narrower bandwidth. A similar phenomenon is often observed in turbo equalization. If the compression is too heavy, causing too severe ISI/ICI, the system is bounded away from the outer code performance; but eventually it will reach it if \( T_\Delta \) is large enough. From the figure it is seen that the system with the smallest product, \( f_\Delta T_\Delta = .45 \), did not converge to the outer code performance. This may be a result of badly chosen parameters \( T_\Delta \) and \( f_\Delta \) rather than a too small product \( f_\Delta T_\Delta \).

In order to see which systems will converge at practical \( E_b/N_0 \), there exists a strong tool, Extrinsic Information Transfer (EXIT) charts. Although their validity is open for discussion and they are not perfect in the case of different block lengths and non-ideal interleavers, they quickly provide insight into the iterative convergence mechanism. In our case, the noise is in fact not Gaussian because of the interference from the non-optimal \( S(A_{int}) \), and this will slightly degrade the EXIT charts. We have not used EXIT charts when establishing which combinations of \( f_\Delta \) and \( T_\Delta \) result in the full coding gain; these are based solely on receiver tests.

We will say full coding gain is achieved if the encoded MFTN system requires less than 0.5 dB more power than the outer code alone to achieve BER \( 5 \times 10^{-5} \). The definition is illustrated in Figure 8; encoded MFTN systems that pass through the "full gain region" achieve full coding gain.

It is now straightforward to exhaust the parameter combinations and find out where the full coding gain is present. In Figure 9 the combinations of \( f_\Delta \) and \( T_\Delta \) are shown that result in the full coding gain for the convolutional codes (7,5) and (74,54), with \( \times \) denoting (7,5) and \( \circ \) (74,54). The figure only applies to the block size and iterations stated above; more iterations can improve the results somewhat. An exhaustive search of \( f_\Delta \) and \( T_\Delta \) is of course not tractable. We have tested \( f_\Delta \) in steps of .01. If the full coding gain is present for some \( T_\Delta = x \) and \( f_\Delta = y \) but not at \( T_\Delta = x \) and \( f_\Delta = y - .01 \), the point \( (x, y) \) is plotted in Figure 9. Dotted lines show constant \( f_\Delta T_\Delta \) products. The smallest product with full coding gain occurs for both codes at \( T_\Delta = .79 \), with \( f_\Delta T_\Delta \approx .43 \) and .455 respectively for the (7,5) and (74,54) codes. There is thus \( \approx 55\% \) bandwidth reduction without loss of BER.

Receiver Complexity. In the iterative detection process we allowed 7 iterations only. The decoder for the outer code is a full complexity BCJR; for the (7,5) and the (74,54) codes this results in 4 and 8 states respectively. The state complexity of the MFTN detector was limited to 32, although this can be reduced for \( T_\Delta \) close to 1. In total, one four-state BCJR and one 32 state BCJR need to be applied 7 times.

A particular combination of \( f_\Delta \) and \( T_\Delta \) is worth mention. When \( T_\Delta = 1 \) it can be seen that the smallest product \( f_\Delta T_\Delta \) such that the full gain is present is only slightly worse than optimal. From a decoding point of view, \( T_\Delta = 1 \) is a very good choice as there will be no ISI to defeat. This implies that for the MFTN detector above there is no memory and the detection is only symbol by symbol. The complexity is only to apply the BCJR to the outer code a number of times. For the (7,5) code there are only 4 states, only 7 iterations are required, and the overall complexity is thereby very small.

There is in fact nothing that prohibits \( T_\Delta \) from being larger than 1. As seen in Figure 9, this leads to reasonable, but not superior, performance. There will be ISI to detect in this case.

Concluding Observations. We make three observations. First, Figure 9 applies to a certain type of turbo receiver. The \( f_\Delta T_\Delta \) products may be smaller with an MLSE receiver or with a receiver that attacks the ICI and ISI in a different pattern. Second, the spectral efficiency (bit rate/Hz) is independent of the basic pulse \( h(t) \); it depends solely on \( f_\Delta \), \( T_\Delta \) and the rate of the outer code. This makes the pulse \( h(t) \) a free optimization parameter. It should be chosen such that full coding gain can be reached at the

![Fig. 7. MFTN detector for subcarrier k'.](image-url)

![Fig. 8. Receiver tests of (7,5) encoded MFTN.](image-url)
smallest open problem $f_{\Delta} T_{\Delta}$. How to perform this optimization is an open problem. Finally, by observing signals, one can see that MFTN signals, like most bandwidth-compressed transmissions, have a higher peak to average power ratio than Nyquist signals. How this ratio compares to that of other methods should be studied in the future.

IV. CONCLUSIONS

We have demonstrated that the idea of faster than Nyquist time compression can be applied at the same time across frequency carriers, to achieve transmission throughput up to twice that of OFDM-like signaling at the same energy and error rate. We have sketched the MFTN limit for multistream Gaussian and root RC pulses, that is, the least time–frequency compression products that yield the antipodal signal error rate. Both synchronization and pulse delays can improve this product. Finally, we have investigated concatenations of convolutional codes and MFTN. An iterated receiver continues to achieve the convolutional coding/orthogonal modulation error performance even under strong compression. Multistream faster than Nyquist techniques thus show great promise as practical, bandwidth saving transmission methods.

ACKNOWLEDGMENTS

Parts of this work were supported by the Swedish Science Board through grant 621-2003-3210, and by the Swedish Foundation for Strategic Research through its Center for High Speed Wireless Communication at Lund University.

APPENDIX 1

In this appendix we derive the Euclidean distance of an MFTN error event set; the derivation assumes an arbitrary starting time of the error event and an arbitrary delay pattern. We start with notation. Let $r(\cdot)$ be the r-step operator; for matrices it is defined as

$$r(\Delta A) = \begin{bmatrix} 0 & \ldots & \Delta a_{0,0} & \ldots & \Delta a_{0,N-1} \\ \vdots & & \Delta a_{K-1,0} & \ldots & \Delta a_{K-1,N-1} \end{bmatrix}.$$  

Then $d^2(\tau_r(\Delta A))$ denotes the normalized distance of the error event $\Delta A$ delayed $r$ steps. Assume also that error event $\Delta A$ starts at time index 0; i.e., for some $k$, $\Delta a_{k,0} \neq 0$. Let $d^2(\Delta A)$ denote the portion of distance contained in the frequency range $f \in {\mathcal F}$. We can compute $d^2(\tau_r(\Delta A))$ by (19)-(21).

The last expression in (19) is the distance of $\Delta A$ at time–frequency offset $\phi = f_{\Delta} T_{\Delta} \gamma \mod 2\pi$; this distance we write as

$$d^2(\Delta A; \phi) = \sum_{k=0}^{K-1} \sum_{l=0}^{K-1} e^{2\pi \phi (l-k)} \beta_{k,l}(\Delta A).$$  

(22)

We can compute $d^2(\Delta A)$ by replacing (21) with

$$\gamma_{k,l}[m, n] = \int_{\mathcal F} H(f) H(f - f_{\Delta}(l-k)) e^{2\pi \phi (n+\delta_k)T_{\Delta} f} \times e^{-j2\pi(m+\delta_k)T_{\Delta} (f-f_{\Delta}(l-k))} df.$$  

(23)

To find the worst case $\phi$ we take the derivative of $d^2(\Delta A; \phi)$,

$$\frac{\partial d^2(\Delta A; \phi)}{\partial \phi} = \sum_{k=0}^{K-1} \sum_{l=0}^{K-1} j2\pi (l-k) e^{2\pi \phi (l-k)} \beta_{k,l}(\Delta A).$$  

(24)

For $f_{\Delta} \geq W$ we have $\gamma_{k,l}[m, n] = 0$ for $|l-k| > 1$; this implies that $\beta_{k,l}(\Delta A) = 0$ for $|l-k| > 1$ as well. Setting the derivative equal to zero we get

$$\sum_{k=1}^{K-1} e^{2\pi \phi} \beta_{k,k-1}(\Delta A) = \sum_{k=0}^{K-1} e^{-j2\pi \phi} \beta_{k,k+1}(\Delta A).$$  

(25)

Equation (25) is the second order equation

$$(e^{2\pi \phi})^2 \sum_{k=1}^{K-1} \beta_{k,k-1}(\Delta A) = \sum_{k=0}^{K-1} \beta_{k,k+1}(\Delta A)$$  

(26)

which can be analytically solved (the case $f_{\Delta} < W$ gives a fourth order equation and is omitted here). The two solutions $\phi_1, \phi_2$ to (26) are

$$\{2\pi \phi_1, 2\pi \phi_2\} = \arg \left\{ \pm \frac{\sum_{k=0}^{K-1} \beta_{k,k+1}(\Delta A)}{\sum_{k=1}^{K} \beta_{k,k-1}(\Delta A)} \right\}.$$  

(27)

where arg{ } denotes the angle in radians of a complex number. Whether $\phi_1$ is the minimum or the maximum can be determined from the sign of $\sum_{k=0}^{K-1} \beta_{k,k+1}(\Delta A)$. This method of determining the worst time offset of an error event is only valid for irrational products $f_{\Delta} T_{\Delta}$. For rational products we can find the worst point by finding the two closest allowed points to the minimizing $T_{\Delta}$.

APPENDIX 2

The derivations in Appendix 1 find the worst time offset $\phi$ for a given error event $\Delta A$. Computing $\hat{p}_{\min}^2$ requires in principle an efficient search over all error events and that is the aim of this appendix. Assume that the search is limited to events of size $K_0 \times N_0$ symbols. We give now an efficient search for this case. The algorithm assumes a pulse $h(t)$ perfectly bandlimited to $W$ positive Hz and is stated here only for the case $f_{\Delta} \geq W$ (if this is not the case, the search can still be used by approximating the bandwidth). Moreover, when we formulate the algorithm, we omit the time dependency of the distance function for notational
convenience: All error events considered are already time shifted such that the minimum distance is obtained. Let $d^2_{\min}|_{\mathcal{F}}$ denote the minimum distance of any error event in the frequency interval $\mathcal{F}$, that is:

$$d^2_{\min}|_{\mathcal{F}} = \min_{\Delta A} d^2(\Delta A)|_{\mathcal{F}}.$$  

(28)

Also, define the frequency intervals

$$\mathcal{F}_k^+ \triangleq \left[-W, -W + (k + 1)f_\Delta\right],$$

$$\mathcal{F}_k^- \triangleq \left[(k_0 - k - 2)f_\Delta + W, (k_0 - 1)f_\Delta + W\right],$$

$$\mathcal{F}_{k_0}^+ \triangleq \left[kf_\Delta + W, kf_\Delta + W\right]$$

for $k \leq k_0 - 2$ and $k_0 = 4$. These intervals are illustrated in Figure 10. In the interval $\mathcal{F}_{k_0}$, the $k_0$th subchannel is free from interference from other channels. Thus we can easily calculate $d^2_{\min}|_{\mathcal{F}_{k_0}}$ by methods that apply for single carrier systems. But since there is no interference in $\mathcal{F}_{k_0}$, we have $d^2_{\min}|_{\mathcal{F}_{k_0}} = d^2_{\min}|_{\mathcal{F}_{k_0+1}} = \cdots = d^2_{\min}|_{\mathcal{F}_{k_0-2}}$; this distance we will refer to as just $d^2_{\min}|_{\mathcal{F}_{k_0}}$. In $\mathcal{F}_0^+$ the first subchannel is

4If time shifts were included, (28) would become $d^2_{\min}|_{\mathcal{F}} = \min_{\Delta A} d^2(\tau_\Delta(\Delta A))|_{\mathcal{F}}$.

free from interference from the others, so we can compute $d^2_{\min}|_{\mathcal{F}_0^+}$ easily. Let $\Delta A_k = [\Delta a_{k,0} \ldots \Delta a_{k,n_0-1}]$ and define the rowwise transpose

$$\Delta A_k^* = \begin{bmatrix} \Delta A_0^* \\ \vdots \\ \Delta A_{k_0-1}^* \end{bmatrix}.$$ 

Throughout, let $\Delta A^*$ denote componentwise complex conjugation. It is easy to show that

$$d^2(\Delta A)|_{\mathcal{F}_0^+} = d^2(\Delta B^*)|_{\mathcal{F}_0^-}$$

(30)

where

$$\Delta B = \begin{bmatrix} \Delta A_{k_0-1}^* \\ \cdots \\ \Delta A_0^* \end{bmatrix}.$$ 

This implies that $d^2_{\min}|_{\mathcal{F}_0^-} = d^2_{\min}|_{\mathcal{F}_0^+}$, all $k$.

The algorithm computes the minimum distance under the assumption that exactly $k_0$ adjacent subchannels are involved in the error event, with $k_0 = 2, \ldots, k_0$ in sequence ($k_0 \geq 2$, otherwise the signaling is one dimensional).

(i) First set $k_0 = 2$. Consider the error symbols in the first subchannel. There must be at least a distance $d^2_{\min}|_{\mathcal{F}_0^+} = d^2_{\min}|_{\mathcal{F}_0^-}$ in the frequency interval $\mathcal{F}_0^-$. Therefore, only events that pile up an amount $d^2(\Delta A)|_{\mathcal{F}_0^+} < 2 - d^2_{\min}|_{\mathcal{F}_0^+}$ need to be considered. Collect these in the set $M_0^+$; that is

$$M_0^+ = \left\{ \Delta A_0 : d^2(\Delta A_0)|_{\mathcal{F}_0^+} < 2 - d^2_{\min}|_{\mathcal{F}_0^+} \right\}.$$ 

(31)

This also implies that only the events $M_0^+ \subseteq \{\Delta A_{k_0-1} : \Delta A_{k_0-1} \in M_0^+ \}$ need to be considered at the last subchannel. What number of subcarriers $k_0$ needs to be tested? The distance of any error event will at least pile $d^2_{\min}|_{\mathcal{F}_0^+} + d^2_{\min}|_{\mathcal{F}_0^-} + (k_0 - 2)d^2_{\min}|_{\mathcal{F}_{k_0-2}}$. Therefore $k_0$ needs to be at most

$$k_0 \leq \frac{2 - d^2_{\min}|_{\mathcal{F}_0^+} + d^2_{\min}|_{\mathcal{F}_0^-}}{d^2_{\min}|_{\mathcal{F}_{k_0-2}}}.$$ 

(32)

$$d^2(\tau_\Delta(\Delta A)) = \int_{-\infty}^{\infty} \sum_{k,l,m,n} \Delta a_{k,m} \Delta a_{l,n}^* h(t - (r + m + \delta_k)T_\Delta) h^*(t - (r + n + \delta_l)T_\Delta) e^{2\pi f_\Delta(t-k)^t} dt$$

$$= \int_{-\infty}^{\infty} \sum_{k,l,m,n} \Delta a_{k,m} \Delta a_{l,n}^* h(t - (m + \delta_k)T_\Delta) h^*(t - (n + \delta_l)T_\Delta) e^{2\pi f_\Delta(t-k)^t} dt$$

$$= \sum_{k,l,m,n} \Delta a_{k,m} \Delta a_{l,n}^* e^{2\pi f_\Delta(t-k)^t} \int_{-\infty}^{\infty} h(t - (m + \delta_k)T_\Delta) h^*(t - (n + \delta_l)T_\Delta) e^{2\pi f_\Delta(t-k)^t} dt$$

$$= \sum_{k,l,m,n} \Delta a_{k,m} \Delta a_{l,n}^* e^{2\pi f_\Delta(t-k)^t} \gamma_{k,l}[m,n]$$

$$= \sum_{k,l} e^{2\pi f_\Delta(t-k)^t} \beta_{k,l}(\Delta A),$$

where

$$\beta_{k,l}(\Delta A) = \sum_{m=0}^{N-1} \sum_{n=0}^{N-1} \Delta a_{k,m} \Delta a_{l,n}^* \gamma_{k,l}[m,n],$$

(20)

and

$$\gamma_{k,l}[m,n] = \int_{-\infty}^{\infty} h(t - (m + \delta_k)T_\Delta) h^*(t - (n + \delta_l)T_\Delta) e^{2\pi f_\Delta(t-k)^t} dt.$$
We can now find the minimum distance of events with two subchannels involved. We exhaust all possible error events of the form

\[
\Delta A = \begin{bmatrix}
\Delta A_0 \\
\Delta A_1 \\
\Delta A_2
\end{bmatrix}
\]

with \(\Delta A_0 \in M_0^+\), \(\Delta A_1 \in M_0\), \(\Delta A_2 \in M_0^+\).

With an abuse of notation, we write the set of all such \(\Delta A\) as \(M_0^+ \times M_0 \times M_0^+\).

(ii) Continue with three subchannels \((k_0' = 3)\). Consider the error event on the second subchannel. As stated above, it is free from interference in \(F_M\). What events are possible? Since we know that we will accumulate up at least an amount \(2d_{\text{min}}^2\) in \(F_0\) and \(2d_{\text{min}}^2\) in \(F_M\) we only have to consider events that have distance less than \(2 - 2d_{\text{min}}^2\) in \(F_M\).

We collect them in a set \(M_M\). Note that it does not matter which subchannel we consider; since there is no interference in \(F_M\), the set is identical for all \(k\) and we take \(k = 1\). Then

\[
M_M = \{\Delta A_1 : d^2(F_0, \Delta A_1) < 2 - 2d_{\text{min}}^2\}.
\]

(33)

We can also easily compute the minimum distance in \(F_M\). Since we know that we will accumulate at least this amount, we can redefine the sets \(M_0^+\) and \(M_0\); \(M_0^+\) becomes \(M_0^+ = \{\Delta A_0 : d^2(F_0, \Delta A_0) < 2 - 2d_{\text{min}}^2\} \) and \(M_0^-\) is found from \(M_0^+\) by the formula

\[
M_0^- = \{\Delta A_0 \in M_0^+ : d^2(F_0, \Delta A_0) > 2 - 2d_{\text{min}}^2\}.
\]

Thus, in the the interval \(F_1^+\) we only have to consider events in \(M_0^+\) \(\times M_0\) when we find

\[
M_1^+ = \{\Delta A_0 \in M_0^+ : d^2(F_0, \Delta A_0) < 2 - 2d_{\text{min}}^2\}.
\]

We also find the set \(M_1^-\). Note that \(M_1^+\) contains events with components over subchannels 0 and 1 and that \(M_1^-\) contains events over subchannels 1 and 2. Thus we cannot treat the two sets independently, as was possible when \(k_0' = 2\). We need to exhaust all events of the form

\[
\Delta A = \begin{bmatrix}
\Delta A_0 \\
\Delta A_1 \\
\Delta A_2
\end{bmatrix},
\]

with \(\Delta A_0 \in M_0^+\), \(\Delta A_1 \in M_0\), \(\Delta A_2 \in M_0^+\).

This generates the minimum distance when there are exactly three subchannels involved in the error event.

(iii) To investigate events on four subchannels, we first compute \(d_{\text{min}}^2\) \(F_0^+\) by exhausting the events in \(M_1^+\).

Then we redefine \(M_1^+\) to \(M_1^+ = \{\Delta A_0 \Delta A_1 \in M_0^+ \times M_0^+ : d^2(F_0, \Delta A_0 \Delta A_1) < 2 - 2d_{\text{min}}^2\} \) and exhaust events in \(M_1^+ \times M_1^-\); redefine \(M_1^-\) defined from \(M_1^+\) according to (34). The procedure can be repeated up to \(k_0\) subchannels, where \(k_0\) satisfies (32).

We summarize the algorithm.

1) First compute \(d_{\text{min}}^2\) \(F_0^+\) and \(d_{\text{min}}^2\) \(F_M\); these are one dimensional searches. If \(d_{\text{min}}^2\) \(F_0^+\) > 1, then stop and go to step 4; the minimum distance for error events with 2 or more subchannels involved is larger than 2. Define the sets \(M_0^+\), \(M_0^-\) and \(M_M\) according to (31), (34) and (33). Set \(k_0' = 2\). Exhaust all events in the set \(M_2^+ \times M_0^-\). This generates the minimum distance when 2 subchannels are involved and we denote it \(d_2^2\).

2) Set \(k_0' = k_0' + 1\) \(k_0'\) is now odd). If \(k_0' > k_0\) then stop; the search over all \(k_0 \times n_0\) error events is complete. Define a temporary set \(T = M_{k_0'}\).

Then redefine \(M_{k_0'}^+\) to

\[
M_{k_0'}^+ = \{\Delta A_0 \ldots \Delta A_{k_0-1} \in T : d^2(F_0, \Delta A_0 \ldots \Delta A_{k_0-1}) < 2 - d_{\text{min}}^2\}
\]

Find \(d_{\text{min}}^2\) \(F_{k_0'}^+\) by exhausting the events in \(M_{k_0'}\). Define

\[
M_{k_0'}^- = \{\Delta A_0 \ldots \Delta A_{k_0-1} \in M_{k_0'}^+ \times M_M : d^2(F_0, \Delta A_0 \ldots \Delta A_{k_0-1}) < 2 - d_{\text{min}}^2\}
\]

If \(d_{\text{min}}^2\) \(F_{k_0'}^+\), then stop and go to step 4; the minimum distance for error events with \(k_0'\) or more subchannels involved is larger than 2. Otherwise, compute the set \(M_{k_0'}^+ \times M_{k_0'}^-\). Find \(d_{\text{min}}^2\) for error events with \(k_0'\) subchannels involved, which we denote \(d_{k_0'}^2\), by exhausting all events of the form

\[
\Delta A = \begin{bmatrix}
\Delta A_0 \\
\vdots \\
\Delta A_{k_0-1}
\end{bmatrix},
\]

with \(\Delta A_0 \ldots \Delta A_{k_0-1} \in M_{k_0'}^+ \times M_{k_0'}^-\).

3) Set \(k_0' = k_0' + 1\) \(k_0'\) is now even). If \(2d_{\text{min}}^2\) \(F_{k_0'}^+\) > 2, then stop and go to step 4; the minimum distance for error events with \(k_0'\) or more subchannels involved is larger than 2. If not, if \(k_0' > k_0\) then stop and go to step 4; the search over all \(k_0 \times n_0\) error events is complete. Otherwise, define
a temporary set $T = M_{k_0/2-1}^+$ and redefine the set

$$M_{k_0/2-1}^+ = \left\{ \left[ \Delta A_0 \ldots \Delta A_{k_0/2-1} \right]^T \in T : 
\begin{align*}
d^2 \left( \left[ \Delta A_0 \ldots \Delta A_{k_0/2-1} \right]^T \right) 
&< 2 - d_{\min}^2 \right\}. 
\end{align*}
$$

Compute the set $M_{k_0/2-1}^-$. Find $d_{\min}^2$ for error events with $k_0$ subchannels involved, which we denote $d_{k_0}^2$, by exhausting all events of the form

$$\Delta A = \begin{bmatrix}
\Delta A_0 \\
\vdots \\
\Delta A_{k_0/2} \\
\Delta A_{k_0/2} \\
\vdots \\
\Delta A_{k_0-1} \\
\end{bmatrix}$$

with

$$\begin{align*}
\left[ \Delta A_0 \ldots \Delta A_{k_0/2-1} \right]^T &\in M_{k_0/2-1}^+ \\
\left[ \Delta A_{k_0/2} \ldots \Delta A_{k_0} \right]^T &\in M_{k_0/2-1}^-.
\end{align*}$$

Go to step 2.

4) Find $d_{\min}^2 = \min\{d_2^2, d_3^2, \ldots, d_{k_0}^2\}$.

REFERENCES


