Separation of Cochannel Signals in TDMA Mobile Radio

Arvind V. Keerthi and John J. Shynk, *Senior Member, IEEE*

***Abstract*— In this paper, we propose a sequential algorithm that separates cochannel time-division multiple-access (TDMA) signals that encounter multipath interference and noise. The receiver employs a multistage architecture where each stage consists of a beamformer and an equalizer that isolates one source, compensates for intersymbol interference (ISI), and demodulates the data. A problem encountered with such bursty sources is that the beamformer/equalizer trained for a particular time slot may not be appropriate for all the data contained in that slot. This occurs because a cochannel source typically overlaps only part of the time slot of interest and may not overlap the training sequence at all. The algorithm presented here overcomes this problem by processing the data forward and backward in a sequential noncausal manner. Computer simulations using signals with the IS-54 format are presented to demonstrate the properties of the sequential algorithm.**

***Index Terms*— Adaptive arrays, adaptive equalizers, adaptive signal processing, cochannel interference, communication systems, land mobile radio cellular systems, least-squares methods, time division multiple access.**

I. Introduction

MOBILE phone service is typically provided by segmenting a geographic region into cells, each of which is allocated a portion of the frequency spectrum. Because the spectrum is limited, carrier frequencies must be reused in distant cells (e.g., the seven-cell reuse pattern) [1]. With a growing demand for cellular services, the cell size can be reduced (cell splitting) to accommodate a greater number of subscribers. However, it becomes more likely that the transmissions in one cell will interfere with those in other cells using the same carrier frequencies. Such *cochannel interference* can often be the dominant impairment in wireless networks. In this paper, we employ spatial processing to isolate sources and null cochannel interference. A multistage structure that incorporates a series of beamformers and signal cancelers allows us to capture multiple cochannel sources.

In digital cellular networks, time-division multiple access (TDMA) enables many users to operate on the same carrier frequency, thus further increasing the capacity of a frequency-reuse system [2]. In North America, TDMA was introduced to be compatible with the current analog system called the Advanced Mobile Phone Service (AMPS); this led to dual-mode AMPS (D-AMPS), which is also known as Interim Standard 54 (IS-54) [3]. The counterpart in Europe is called the Global System for Mobile (GSM) [4]. In this paper, we focus only on the frame and time slot structure of the IS-54 system; the slot structure and the greater bandwidth of GSM require that a different algorithm be employed for cochannel signal separation, as described in [5]. The North American standard IS-136 is similar to that of IS-54, but it allows for a host of new features and services [2].

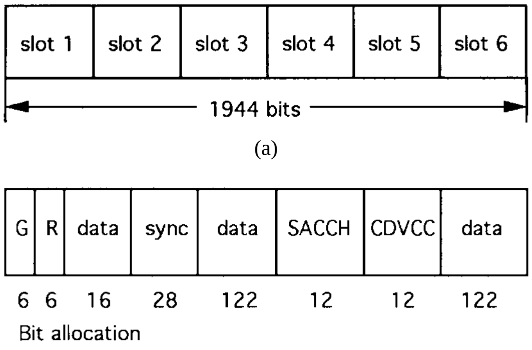


Fig. 1. IS-54 frame and slot structure (there are 2 bits per symbol). (a) One frame. (b) Time slot of the mobile to base station (reverse path or uplink).

The IS-54 frame format allows up to six users to be time multiplexed on the same carrier such that their time slots (bursts) do not overlap. A simplified illustration of the frame and slot structure is shown in Fig. 1. Note that even though these slots share a common carrier frequency, they do not interfere with each other; the guard (G) spacing in Fig. 1(b) is incorporated to prevent intracell burst collisions. Cochannel interference arises from signals transmitted in distant cells whose slots are not synchronized with those of the current cell (i.e., cochannel interference is an intercell impairment).

Fig. 1 shows the slot structure for transmission from a mobile to the base station (the reverse path or uplink). The modulation format is differential quadrature phase-shift keying (DQPSK) so that there are two bits per symbol. Each slot contains a 14-symbol sequence for frame synchronization that can also be used to compensate for channel distortion; it functions as a training sequence for the beamformer/equalizer in the receiver. The carrier power is zero during the guard time (G), which spans three symbols; three additional symbol periods (R) are provided so that the carrier can ramp up to its operating

Manuscript received January 29, 1997; revised February 27, 1998. This work was supported by Applied Signal Technology, Inc., and the University of California MICRO Program. The associate editor coordinating the review of this paper and approving it for publication was Dr. Sergios Theodoridis.

A. V. Keerthi was with the Department of Electrical and Computer Engineering, University of California, Santa Barbara, CA 93106 USA. He is now with Qualcomm, Inc., San Diego, CA 92121 USA.

J. J. Shynk is with the Department of Electrical and Computer Engineering, University of California, Santa Barbara, CA 93106 USA (e-mail: shynk@ece.ucsb.edu).

Publisher Item Identifier S 1053-587X(98)07070-6.

1053–587X/98$10.00 © 1998 IEEE

power level. There are also 12 administrative symbols for link control and network management [slow associated control channel (SACCH) and coded digital verification color code (CDVCC)]. The data are spread out over three regions of the time slot for a total of 130 symbols.

The proposed *sequential* algorithm operates on the baseband data obtained from an *array* of antenna elements. A least-squares (LS) algorithm designed for the training sequence is employed to compute the coefficients of the *beamformer/equalizer*. Frame synchronization is achieved by utilizing the known correlation properties of the IS-54 training sequence. We assume that the channel characteristics are nonstationary so that the coefficients must be recomputed for each new burst. A multipath scenario is assumed whereby the cochannel symbol streams propagate through multiple channels, giving rise to intersymbol interference (ISI).

Several authors have studied the effects of cochannel interference for continuous (nonbursty) digital transmission (see, e.g., [6] and [7]), and various receiver architectures have been developed for this scenario (see, e.g., [8] and [9]). However, the problem of cochannel signal separation for bursty (TDMA) signals has been addressed by relatively few authors. The main problem with cancelling the cochannel interference in TDMA systems is that the beamformer/equalizer coefficients for the current slot may not be suitable for demodulating all the data in that slot. This problem arises because the cochannel sources are not synchronized (i.e., bursty); thus, they may not overlap the training sequence of the current slot.

An algorithm that jointly recovers two cochannel symbol streams based on recursive least squares (RLS) channel estimation and the Viterbi algorithm (VA) was proposed in [10]. However, the authors do not explicitly mention how asynchronous cochannel bursts should be handled; it is not clear how frame synchronization is achieved in the presence of cochannel interference. The authors in [11] acknowledge the problem of asynchronous cochannel bursts; however, in their analysis and simulations, the cochannel bursts are assumed to be precisely time aligned. Their approach is a version of joint maximum likelihood sequence estimation (JMLSE) [12] that is modified for bursty digital transmission. Although they mention that network management could be used to align the bursts, this is not a practical assumption; obviously, it is preferable that the signal separation algorithm not depend on time-aligned cochannel bursts.

The method in [13] also addresses the problem of asynchronous bursts. Assuming that the starting point of one cochannel burst is known, all channels and the corresponding transmitted symbols are simultaneously estimated by maximizing a likelihood function over all possible transmitted symbols (discrete-space optimization) and channels (continuous-space optimization). The author acknowledges the large complexity of this approach and proposes an alternative method, which assumes that the starting points of all the bursts are known. This assumption makes it possible to derive the channel estimates *a priori* using simple cross-correlation techniques; however, the resulting channel estimates may not be accurate.

The antenna array in [14] nulls cochannel interference by using the training sequences of the current and previous time slots. First, the synchronization sequences are used to estimate the mobile channel. These estimates are employed to compute the correlation matrices of the desired and interfering slots from which the least-squares weights of the antenna array are derived. The approach in [14] resembles that presented in [15] because both use the fact that cochannel interference that does not intersect the training sequence of the current slot does so with the adjacent slot. The antenna array in [15] is initialized by weights derived from the synchronization sequence and the guard symbols.

In our approach, sequential processing is utilized whereby only a portion of the data is initially estimated (beamformed). After a channel estimator and a signal canceler remove these estimates from the array input, the remaining data are estimated using the same set of beamformer coefficients. In the sequel, we compare the performance of this approach to the method in [15], which we will refer to as the guard-ramp (GR) method.[1] The paper is organized as follows. Section II gives further details of the IS-54 slot structure and establishes a baseband model for symbol generation and transmission. Section III describes the architecture of the beamformer/equalizer that separates the cochannel sources and compensates for the ISI. The frame synchronization algorithm is discussed in Section IV. The proposed sequential separation (SQ) algorithm is then presented in Section V, and its performance is compared with that of the GR method in Sections VI and VII. Conclusions are given in Section VIII.

II. Signal and Channel Models

For physical reasons, an antenna array may generally be available only at the base station. Therefore, we will be interested in cochannel interference affecting the uplink (mobile to base station). Note also that the mobiles always utilize TDMA such that transmissions to the base station occur in bursts; the base station transmits to the mobiles in a continuous (nonbursty) manner and thus employs time division multiplexing (TDM) and not TDMA.

The base station receives synchronized bursts from users within its cell and randomly timed bursts from users in distant cells. In the sequel, we do not differentiate between a signal of interest and cochannel signals. Our goal is to separate and estimate all signals impinging on the antenna array. Thus, when we refer to L cochannel sources, this does not necessarily imply that all L users interfere with each other. An example of this will be presented later in the computer simulations.

A. Signal Model

Recall the frame and slot format for the IS-54 standard shown in Fig. 1. At least two modes of operation are possible: full rate, where three users are assigned two slots each in an A-B-C-A-B-C pattern for each frame, and half rate, where each of six users is assigned only one slot in a frame. There are four basic types of symbols in each slot:

[1] Actually, only the guard symbols are used in [15], but an obvious extension is to use both the guard and ramp-up symbols.

TABLE I
SYNCHRONIZATION SEQUENCES OF THE IS-54 STANDARD

Sequence	Phase Change (radians)													
1	$-\frac{\pi}{4}$	$-\frac{\pi}{4}$	$-\frac{\pi}{4}$	$\frac{3\pi}{4}$	$\frac{\pi}{4}$	$\frac{3\pi}{4}$	$-\frac{3\pi}{4}$	$\frac{3\pi}{4}$	$-\frac{3\pi}{4}$	$-\frac{\pi}{4}$	$\frac{3\pi}{4}$	$\frac{\pi}{4}$	$-\frac{\pi}{4}$	$-\frac{\pi}{4}$
2	$-\frac{\pi}{4}$	$-\frac{\pi}{4}$	$-\frac{\pi}{4}$	$\frac{3\pi}{4}$	$-\frac{3\pi}{4}$	$\frac{3\pi}{4}$	$\frac{\pi}{4}$	$\frac{3\pi}{4}$	$\frac{\pi}{4}$	$-\frac{\pi}{4}$	$\frac{3\pi}{4}$	$-\frac{3\pi}{4}$	$-\frac{\pi}{4}$	$-\frac{\pi}{4}$
3	$-\frac{3\pi}{4}$	$\frac{\pi}{4}$	$\frac{3\pi}{4}$	$-\frac{3\pi}{4}$	$-\frac{3\pi}{4}$	$-\frac{\pi}{4}$	$\frac{\pi}{4}$	$-\frac{3\pi}{4}$	$-\frac{3\pi}{4}$	$\frac{\pi}{4}$	$\frac{\pi}{4}$	$\frac{\pi}{4}$	$-\frac{3\pi}{4}$	$\frac{\pi}{4}$
4	$\frac{\pi}{4}$	$-\frac{3\pi}{4}$	$\frac{3\pi}{4}$	$\frac{\pi}{4}$	$\frac{\pi}{4}$	$-\frac{\pi}{4}$	$-\frac{3\pi}{4}$	$\frac{\pi}{4}$	$\frac{\pi}{4}$	$-\frac{3\pi}{4}$	$-\frac{3\pi}{4}$	$-\frac{3\pi}{4}$	$\frac{\pi}{4}$	$-\frac{3\pi}{4}$
5	$\frac{\pi}{4}$	$\frac{3\pi}{4}$	$\frac{\pi}{4}$	$-\frac{3\pi}{4}$	$-\frac{3\pi}{4}$	$-\frac{\pi}{4}$	$\frac{\pi}{4}$	$-\frac{\pi}{4}$	$\frac{\pi}{4}$	$-\frac{3\pi}{4}$	$-\frac{3\pi}{4}$	$\frac{3\pi}{4}$	$\frac{\pi}{4}$	$\frac{3\pi}{4}$
6	$-\frac{3\pi}{4}$	$\frac{3\pi}{4}$	$-\frac{3\pi}{4}$	$\frac{\pi}{4}$	$\frac{\pi}{4}$	$-\frac{\pi}{4}$	$-\frac{3\pi}{4}$	$-\frac{\pi}{4}$	$-\frac{3\pi}{4}$	$\frac{\pi}{4}$	$\frac{\pi}{4}$	$\frac{3\pi}{4}$	$-\frac{3\pi}{4}$	$\frac{3\pi}{4}$

1) synchronization symbols, which are known at the receiver and therefore may be used for frame synchronization and beamformer/equalizer training;
2) guard (G) and ramp-up (R) symbols, which serve to separate the bursts in time and to increase the carrier power to operating levels, respectively;
3) data symbols, which contain information generated by the subscriber;
4) administrative symbols (SACCH and CDVCC), which store network information and need to be estimated at the receiver.

For our purposes, we will group the administrative symbols together with the user data.

Each user in a half-rate system is assigned one of the six synchronization sequences listed in Table I. Because the information is encoded differentially (DQPSK), only phase changes are indicated. In a full-rate system (three users), only three sequences are needed. These sequences are approximately uncorrelated; in principle, it is possible to determine which user is occupying a given time slot by computing a series of cross-correlation sequences at the receiver and searching for the highest peak. This will be discussed further in Section IV on frame synchronization.

Denote the DQPSK symbol stream emanating from the lth cochannel user as $d_l(k)$. If the lth user is inactive at time k (i.e., between bursts), then $d_l(k) = 0$. During a burst, $d_l(k)$ is determined as follows. The first three symbols (the guard time G) are set to zero. Symbols 4–6, during which the carrier power ramps up, may be set to any unit-magnitude complex number (the actual ramp up is performed via the carrier amplitude function $A_l(t)$ and is described below). Next, the bit stream entering the modulator is split into two separate binary streams $\{a_l(k), b_l(k)\}$. The odd-numbered bits form $\{a_l(k)\}$, and the even numbered bits form $\{b_l(k)\}$. The $\{a_l(k), b_l(k)\}$ are encoded into the symbols $\{d_l(k)\}$ via $d_l(k) = d_l(k-1)e^{j\theta(k)}$, where the phase change $\theta(k)$ may take on one of four values, depending on the $\{a_l(k), b_l(k)\}$, as shown in Table II. Assuming that the ramp-up symbols are all set to $e^{j\pi/4}$, the eight DQPSK constellation points are ±1, $e^{\pm j\pi/4}$, $e^{\pm j\pi/2}$, and $e^{\pm j3\pi/4}$.

TABLE II
DQPSK PHASE CHANGES

$a_l(k)$	$b_l(k)$	$\theta(k)$
1	1	$-\frac{3\pi}{4}$
0	1	$\frac{3\pi}{4}$
0	0	$\frac{\pi}{4}$
1	0	$-\frac{\pi}{4}$

The symbol stream $\{d_l(k)\}$ excites a transmit filter $g(t)$, which has a linear phase and the square-root raised-cosine (SRRC) frequency response [16], shown in (1) at the bottom of the page, where $T = 41.15$ μs is the symbol period, and $\beta = 0.35$ is the roll-off factor. The output of the transmit filter, given by

$$s_l(t) = \sum_{n=0}^{\infty} g(t - nT)\, d_l(n) \tag{2}$$

is upconverted to the carrier frequency $\omega_c = 2\pi f_c$ and propagates through a multipath channel.

B. Channel Model

The transmitted waveform is

$$\tilde{s}_l(t) = \text{Re}[A_l(t)s_l(t)e^{j\omega_c t}] \tag{3}$$

where the carrier amplitude $A_l(t)$ is a periodic function with period equal to the IS-54 frame duration, as shown in Fig. 2. The carrier amplitudes of the users within a given cell are coordinated via the functions $\{A_l(t)\}$ so that only one user is active at any given time. The IS-54 standard specifies a

$$|G(f)| = \begin{cases} 1, & 0 \le |f| \le \dfrac{1-\beta}{2T} \\ \sqrt{\dfrac{1}{2}\left(1 - \sin\left[\dfrac{\pi(2|f|T-1)}{2\beta}\right]\right)}, & \dfrac{1-\beta}{2T} < |f| \le \dfrac{1+\beta}{2T} \\ 0, & |f| > \dfrac{1+\beta}{2T} \end{cases} \tag{1}$$

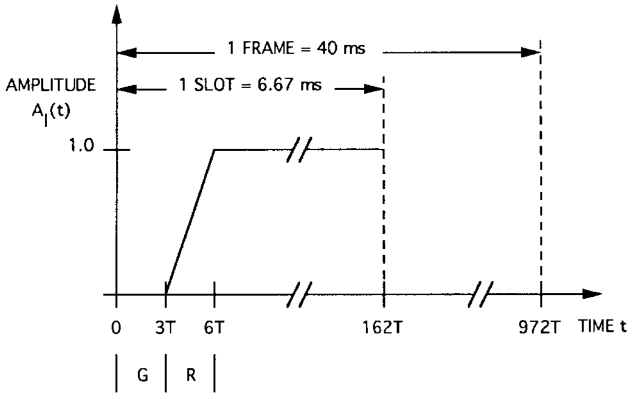


Fig. 2. Carrier amplitude function $A_l(t)$.

template for the time variation of the $\{A_l(t)\}$ [3]; for our purposes, it will be sufficient to model $A_l(t)$ as a piecewise linear function of time with a maximum amplitude equal to unity.

Denote the number of cochannel sources by L. Recall that not all of these signals interfere with each other. The transmitted waveform $\tilde{s}_l(t)$ of the lth user propagates through a multipath channel with M_l paths. The attenuation associated with the ith path is a_{li}, and the corresponding propagation delay is τ_{li}. Assuming that the array sensors are sufficiently close to each other, the signal at the mth antenna element is given by [17]

$$\tilde{r}_m(t) = \mathrm{Re}\left[\sum_{l=1}^{L}\sum_{i=1}^{M_l} a_{li}A_l(t-\tau_{li})s_l(t-\tau_{li}) \cdot e^{j\omega_c(t-\tau_{li})}e^{j\omega_c d_{mli}}\right] + \tilde{n}_m(t) \tag{4}$$

where $\tilde{n}_m(t)$ is receiver noise. The time for the ith ray of the lth source to travel from a reference point (in the vicinity of the antenna array) to the mth antenna element is denoted by d_{mli}. The phase factor $e^{j\omega_c d_{mli}}$ depends on the array geometry and the angle of arrival (AOA) θ_{li} of the ith ray. For example, if the array is linear with uniformly spaced elements and interelement distance d, then taking the reference point to be one of the end elements, it is straightforward to show that $e^{j\omega_c d_{mli}} = e^{j2\pi(d/\lambda)(m-1)\sin\theta_{li}}$, where λ is the wavelength of the cochannel waveforms, and θ_{li} is measured with respect to the perpendicular of the array.

For convenience, define the complex number $\alpha_{mli} \triangleq a_{li}e^{-j\omega_c\tau_{li}}e^{j\omega_c d_{mli}}$. After downconversion, the baseband signal at the mth antenna element can be written as

$$\begin{aligned} r_m(t) &= \sum_{l=1}^{L}\sum_{i=1}^{M_l} A_l(t-\tau_{li})s_l(t-\tau_{li})\alpha_{mli} + n_m(t) \\ &= \sum_{l=1}^{L} A_l(t)s_l(t) * \left(\sum_{i=1}^{M_l}\alpha_{mli}\delta(t-\tau_{li})\right) + n_m(t) \end{aligned} \tag{5}$$

where the operator $*$ denotes convolution, $\delta(t)$ is the Dirac delta function, and $n_m(t)$ is baseband noise. The quantity in parentheses is the impulse response of the channel between the lth source and the mth antenna element.

The downconverted signal is sampled by an A/D converter operating at twice the symbol rate (i.e., $T/2$-spaced fractional sampling). Substituting (2) into (5) and sampling the result yields

$$\begin{aligned} r_m(kT/2) = \sum_{l=1}^{L} A_l(kT/2-\tau_{li}) \sum_{i=1}^{M_l}\alpha_{mli}\sum_{n=0}^{\infty} \\ \cdot\, g(kT/2-nT-\tau_{li})\, d_l(n) + n_m(kT/2). \end{aligned} \tag{6}$$

Because the period of variation of the $\{A_l(t)\}$ is large (40 ms) compared with the delays $\{\tau_{li}\}$ (about 40 μs), we may employ the approximation $A_l(kT/2-\tau_{li}) \approx A_l(kT/2)$. The model may be simplified further by combining the transmit filter and the propagation channels to generate a set of "effective" channels. In order to perform this, we truncate the SRRC pulse $g(t)$ so that its impulse response spans $[-Q_l, Q_l]$. Thus, we may rewrite (6) as

$$r_m(kT/2) = \begin{cases} \displaystyle\sum_{l=1}^{L} A_l(kT/2)\sum_{i=1}^{M_l}\alpha_{mli}\sum_{r=-Q_l}^{Q_l} \\ \quad \cdot\, g(rT-\tau_{li})d_l(k/2-r) \\ \quad +\, n_m(kT/2), & k \text{ even} \\ \displaystyle\sum_{l=1}^{L} A_l(kT/2)\sum_{i=1}^{M_l}\alpha_{mli}\sum_{r=-Q_l}^{Q_l} \\ \quad \cdot\, g(rT+T/2-\tau_{li}) \\ \quad \cdot\, d_l((k-1)/2-r) \\ \quad +\, n_m(kT/2), & k \text{ odd.} \end{cases} \tag{7}$$

Finally, defining the composite channel coefficients

$$f^e_{ml}(r) \triangleq \sum_{i=1}^{M_l}\alpha_{mli}g(rT-\tau_{li}) \tag{8}$$

$$f^o_{ml}(r) \triangleq \sum_{i=1}^{M_l}\alpha_{mli}g((2r+1)T/2-\tau_{li}) \tag{9}$$

where the superscripts e and o denote the even and odd components, respectively, yields

$$r_m(kT/2) = \begin{cases} \displaystyle\sum_{l=1}^{L} A_l(kT/2)\sum_{r=-Q_l}^{Q_l} f^e_{ml}(r) \\ \quad \cdot\, d_l(k/2-r) + n_m(kT/2), & k \text{ even} \\ \displaystyle\sum_{l=1}^{L} A_l(kT/2)\sum_{r=-Q_l}^{Q_l} f^o_{ml}(r) \\ \quad \cdot\, d_l((k-1)/2-r) \\ \quad +\, n_m(kT/2), & k \text{ odd.} \end{cases} \tag{10}$$

The output of the A/D converter is thus a $T/2$-spaced discrete-time sequence. For convenience, we will denote it by $x_m(k) \equiv r_m(kT/2)$ (where T has been suppressed). Assuming that there are N antenna elements, these samples may be collected into the following array input vector $\mathbf{x}(k) \triangleq [x_1(k), \cdots, x_N(k)]^T$.

An example of the type of overlap that can occur for cochannel TDMA signals is shown in Fig. 3. Note that we have not distinguished among the different training sequences in the figure (refer to Table III for this information and other

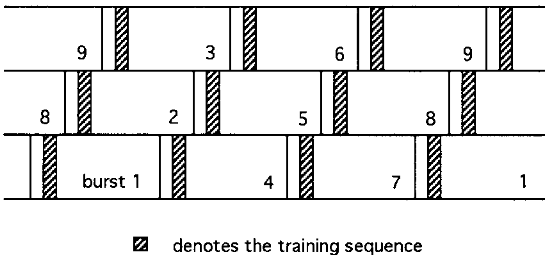


Fig. 3. Example of cochannel TDMA bursts. (The signal parameters are specified in Table III.)

TABLE III
Channel Parameters for Scenario 1

Source	Relative delay	Synchronized with sources	Training sequence	Angle of arrival
1	0	4 and 7	1	0.0°
2	35	5 and 8	4	10.0°
3	70	6 and 9	4	−45.0°
4	162	1 and 7	2	60.0°
5	197	2 and 8	5	−15.0°
6	232	3 and 9	5	−75.0°
7	324	1 and 4	3	15.0°
8	359	2 and 5	6	25.0°
9	394	3 and 6	6	−60.0°

signal parameters). Suppose we are trying to demodulate burst 4. Observe that bursts 2, 3, 5, and 6 are cochannel interference because they overlap burst 4. A beamformer/equalizer that is initialized by the synchronization sequence in burst 4 will perform satisfactorily during bursts 2 and 3 but may not be acceptable during bursts 5 and 6 because its training sequence does not overlap those bursts. A similar problem exists for the other bursts shown in the figure.

III. Beamforming and Equalization

The N-element antenna array provides $2N$ baseband samples $\{x_m(k)\}$ for every symbol interval, which contain a mixture of the L cochannel signals and white Gaussian noise. Because of the fractional sampling ($T/2$-spaced), the array input signals are processed by two weight-and-sum beamformers: one for the even samples and one for the odd samples. The weights of these two beamformers, which are denoted by $\mathbf{w}_e$ and $\mathbf{w}_o$ (each an N-vector), will be optimized separately on the two streams of data because they arise from different statistics. The overall output of the array is

$$y(k) = \begin{cases} \mathbf{w}_e^H \mathbf{x}(k), & k - k_j \text{ even} \\ \mathbf{w}_o^H \mathbf{x}(k), & k - k_j \text{ odd} \end{cases} \tag{11}$$

where $\mathbf{x}(k)$ was previously defined. The starting point of the training sequence for the jth slot is denoted by k_j (see Section V for further discussion of k_j).

The beamformer is designed to cancel (null) cochannel signals arriving from different AOA's, including multipath interference but not the ISI induced by the transmit filter. This ISI is remedied by interleaving the beamformer output samples and processing them with a $T/2$-spaced linear equalizer with $2P$ coefficients. Denoting the equalizer weight vector by $\mathbf{h} \triangleq [h_1, \cdots, h_{2P}]^T$, the output of the equalizer, which appears once every symbol (T-spaced), is given by

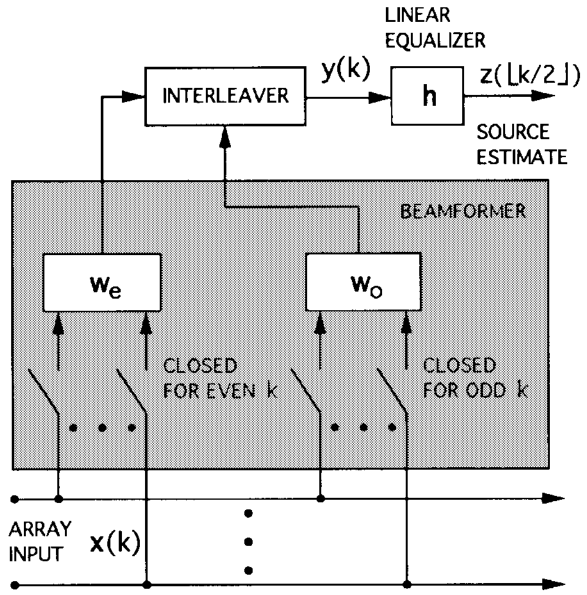


Fig. 4. Architecture of the beamformer/equalizer.

$$z(\lfloor k/2 \rfloor) = \mathbf{h}^H \mathbf{y}(k), \qquad k - k_j \text{ even} \tag{12}$$

where $\mathbf{y}(k) \triangleq [y(k), \cdots, y(k - 2P + 1)]^T$, and $\lfloor a \rfloor$ denotes the greatest integer $\leq a$. A block diagram of the architecture is shown in Fig. 4. Note that a decision-feedback equalizer (DFE) could be incorporated at the output of the linear equalizer (although we do not consider it in this paper).[2]

In order to separate and estimate one of the cochannel sources from the mixture in $\mathbf{x}(k)$, we need to appropriately initialize the weight vectors $\mathbf{w}_e$, $\mathbf{w}_o$, and $\mathbf{h}$ from the training sequence. This is performed after frame synchronization is achieved (i.e., when the starting point k_j of the training sequence in the jth slot is identified).

IV. Frame Synchronization Algorithm

Recall that the training sequences of the IS-54 standard are designed to be approximately uncorrelated. Thus, in principle, a training sequence may be located by cross-correlating the antenna array samples with the training sequence for each user and identifying the largest peak. However, this method will not work when there is strong cochannel interference, which cannot be removed until the training sequence has been located. A solution to this problem proceeds as follows [15]. Assume that a training sequence starts at an arbitrary point in time, and compute the beamformer weights based on this assumption. Pass the antenna array data through the beamformer, cross-correlate the beamformer output and the training sequence, and store the result. Repeat this process for every

[2] The reader is referred to the sequential beamforming algorithm for GSM signals in [5], where a DFE is employed instead of a linear equalizer.

sampling instant over a specified search window, yielding a cross-correlation sequence for that duration of samples. The peaks in the cross-correlation function indicate the sampling instants at which the training sequences are located.

Only the beamformers are needed in the frame synchronization algorithm (i.e., the follow-on linear equalizer is not used in this step). Because the beamformers are not designed to remove the ISI induced by the transmit filter, the appropriate training sequences for the beamformer (even and odd parts) are derived by convolving the IS-54 training sequences with the transmit filter and sampling the output at twice the symbol rate. With these modified training sequences, the beamformer weights are computed using the method of least squares [18].

Denote the ith training sequence as $\{t_i(0), \cdots, t_i(N_T-1)\}$, where $N_T = 14$ is its length. The training data for the beamformer comprise the following $T/2$-spaced sequence

$$t_{b,\,i}(k) = \sum_{n=0}^{N_T-1} t_i(n)g(kT/2-nT), \qquad k=0, \cdots, 2N_T-1 \tag{13}$$

where we have implicitly used the fact that the carrier power equals unity during the training sequence. Recall that the transmit filter $g(t)$ has the SRRC frequency response in (1). The subscript b denotes that the modified training sequence is designed for the beamformer.

Suppose that we are searching for training sequence i. Let the search window be equal to the length of the IS-54 frame (i.e., 972 symbols or 1944 $T/2$-spaced samples). For an arbitrary time instant k, the beamformer weight vectors $\mathbf{w}_e$ and $\mathbf{w}_o$ are computed to minimize the least-squares (LS) cost function [18]

$$\begin{aligned} J(\mathbf{w}_e, \mathbf{w}_o) = \sum_{j=0}^{N_T-1} & (|\mathbf{w}_e^H \mathbf{x}(k+2j) - t_{b,\,i}(2j)|^2 \\ & + |\mathbf{w}_o^H \mathbf{x}(k+2j+1) - t_{b,\,i}(2j+1)|^2). \end{aligned} \tag{14}$$

This cost function is motivated by the fact that alternate samples arise from different input statistics, as mentioned previously. The solution to this minimization problem is given by

$$\mathbf{w}_e = \mathbf{R}^{-1}(k)\mathbf{p}_e(k), \mathbf{w}_o = \mathbf{R}^{-1}(k+1)\mathbf{p}_o(k+1) \tag{15}$$

where $\mathbf{R}(k)$ is an autocorrelation matrix, and $\{\mathbf{p}_e(k),\, \mathbf{p}_o(k)\}$ are cross-correlation vectors computed as

$$\mathbf{R}(k) \triangleq \sum_{j=0}^{N_T-1} \mathbf{x}(k+2j)\mathbf{x}^H(k+2j) \tag{16}$$

$$\mathbf{p}_e(k) \triangleq \sum_{j=0}^{N_T-1} \mathbf{x}(k+2j)t_{b,\,i}^*(2j) \tag{17}$$

$$\mathbf{p}_o(k+1) \triangleq \sum_{j=0}^{N_T-1} \mathbf{x}(k+2j+1)t_{b,\,i}^*(2j+1) \tag{18}$$

where the superscript * denotes complex conjugation.

The data $\{\mathbf{x}(k), \cdots, \mathbf{x}(k+2N_T-1)\}$ are processed by the beamformer weights according to (11), i.e.,

$$y(k+j) = \begin{cases} \mathbf{w}_e^H \mathbf{x}(k+j), & j=0, 2, \cdots, 2N_T-2 \\ \mathbf{w}_o^H \mathbf{x}(k+j), & j=1, 3, \cdots, 2N_T-1. \end{cases} \tag{19}$$

The cross-correlation coefficient at time instant k may be computed as

$$C(k) = \frac{\left| \displaystyle\sum_{j=0}^{2N_T-1} t_{b,\,i}^*(j)y(k+j) \right|}{\sqrt{\left(\displaystyle\sum_{j=0}^{2N_T-1} |t_{b,i}(j)|^2 \right) \left(\displaystyle\sum_{j=0}^{2N_T-1} |y(k+j)|^2 \right)}}. \tag{20}$$

This procedure is repeated for every sample point in the window, and the beamformer training sequences are located at those time instants where $C(k)$ exceeds a prespecified threshold.

The step requiring the greatest number of computations involves the autocorrelation matrix inverses in (15). This complexity may be reduced, however, by noting that

$$\begin{aligned} \mathbf{R}(k+2) = & \mathbf{R}(k) - \mathbf{x}(k)\mathbf{x}^H(k) \\ & + \mathbf{x}(k+2N_T)\mathbf{x}^H(k+2N_T). \end{aligned} \tag{21}$$

Applying the matrix inversion lemma [18] twice to obtain $\mathbf{R}^{-1}(k+2)$ directly from the inverse $\mathbf{R}^{-1}(k)$ yields

$$\begin{aligned} & \mathbf{R}_{tmp}^{-1} \\ & \quad = \mathbf{R}^{-1}(k) + \frac{\mathbf{R}^{-1}(k)\mathbf{x}(k)\mathbf{x}^H(k)\mathbf{R}^{-1}(k)}{1 - \mathbf{x}^H(k)\mathbf{R}^{-1}(k)\mathbf{x}(k)} \end{aligned} \tag{22}$$

$$\begin{aligned} & \mathbf{R}^{-1}(k+2) \\ & \quad = \mathbf{R}_{tmp}^{-1} - \frac{\mathbf{R}_{tmp}^{-1}\mathbf{x}(k+2N_T)\mathbf{x}^H(k+2N_T)\mathbf{R}_{tmp}^{-1}}{1 + \mathbf{x}^H(k+2N_T)\mathbf{R}_{tmp}^{-1}\mathbf{x}(k+2N_T)} \end{aligned} \tag{23}$$

where $\mathbf{R}_{tmp}$ is an intermediate matrix used only to compute the final result in (23).

The steps in the frame synchronization algorithm are summarized below.

- Choose a search window (e.g., equal to the IS-54 frame duration).
- For each sampling instant k in the search interval, perform the following three steps.
  a) If the sampling point is first in the search window, compute the beamformer weights using (15)–(18). Otherwise, compute the beamformer weights using (15), (17), (18), (22), and (23) (via the matrix inversion lemma).
  b) Pass the antenna array samples $\{\mathbf{x}(k), \cdots, \mathbf{x}(k+2N_T-1)\}$ through the beamformers (even and odd) as in (19).
  c) Cross-correlate the beamformer output with each (modified) training sequence as in (20).

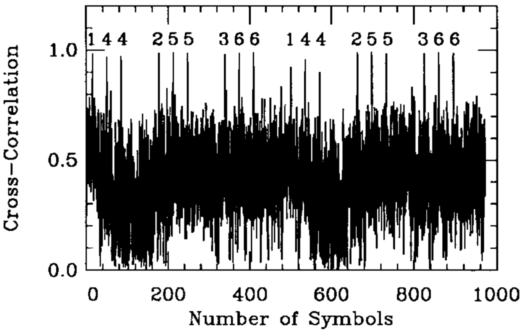


Fig. 5. Superimposed cross-correlation functions for the beamformer output. The training sequence for each peak is labeled (refer to Fig. 3 and Table III).

- The training sequences start at those sampling instants where the cross-correlation coefficients exceed a prespecified threshold.

It should be mentioned that the previous expressions illustrate the basic approach of the frame synchronization algorithm and its various steps. Of course, computationally efficient implementations of the algorithm should be employed, which are analogous to the matrix inversion lemma used above. In this paper, we have not focused on the computational complexity (although we discuss it briefly in Section VII in connection with the GR algorithm).

As specified in the IS-54 standard, we assume that the multipath is limited to one symbol; thus, the maximum number of coefficients in each channel is two. A beamformer is optimized so that its output approximates one of the modified training sequences. In this way, the short-delay multipath is coherently combined, yielding a dominant peak for each source in the cross-correlation functions. Fig. 5 shows the six superimposed cross-correlation functions produced by the frame synchronization algorithm for each of the six synchronization sequences of the example in Fig. 3 (the channel parameters are presented later in Table III and Section VI when the computer simulations are discussed).

V. Sequential Separation (SQ) Algorithm

For convenience in the following discussion, we assume that no more than two of the L cochannel signals overlap at any time instant. The arguments remain unchanged, however, for any number of overlapping signals. The sequential separation (SQ) algorithm processes the data in two passes, which are referred to as the *forward pass* and the *reverse pass*. The forward pass consists of two steps: *frame synchronization* and *beamforming*. Frame synchronization (i.e., determining the starting points of the training sequences) is performed using the algorithm described in the previous section. Once this is done, the corresponding beamformer weights [which are computed using the modified training sequences in (13)] are employed to estimate a subset of the data. Cochannel interference that overlaps the current training sequence can be nulled by these beamformer weights. On the other hand, cochannel interference that appears in the same time slot but does not overlap the current training sequence cannot be adequately nulled by the same beamformer weights; these samples will be handled during the reverse pass. At the end of the forward pass, the beamformer output is free (ideally) of cochannel interference only in the vicinity of the training sequences (the limits of each vicinity will be established later in this section).

The reverse pass consists of three steps: *channel estimation*, *signal cancellation*, and *beamforming*. During channel estimation, the channel for each slot in the window is estimated. During signal cancellation, the estimated channels are used to cancel the beamformer output (which is obtained in the forward pass) from the antenna array samples.[3] Over the interval where signal cancellation is performed, the antenna samples are now free of cochannel interference (because we have assumed that at most two sources overlap). Thus, the beamformer weights used during the forward pass can also be used here. The reverse pass completes the process of signal separation by beamforming over those regions of the window that were not estimated in the forward pass. Finally, after the reverse pass has terminated, the follow-on linear equalizer removes the ISI associated with each captured source in the window.

These steps will now be discussed in greater detail.

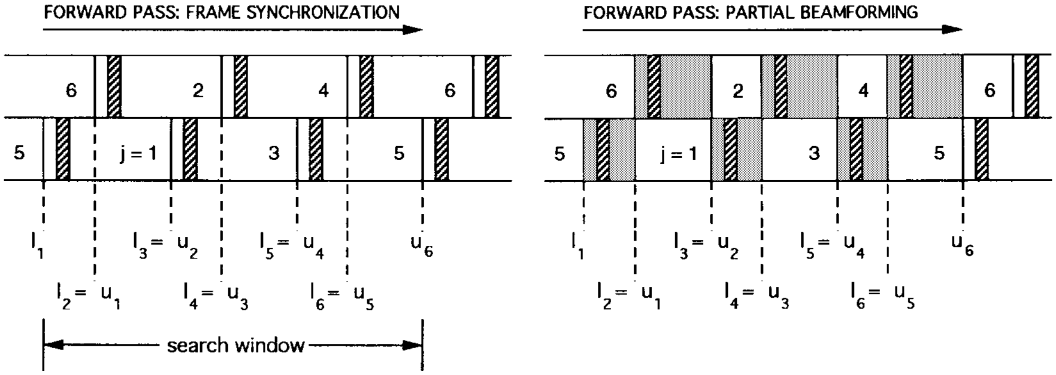
Forward Pass: During the forward pass, the following two steps are performed.

Frame Synchronization: For convenience, assume that we have chosen the search window of the frame synchronization algorithm to be the same duration as the IS-54 frame. For each of the six possible training sequences (three training sequences for a full-rate system), the frame synchronization algorithm will generate a cross-correlation function with several peaks, which indicate the starting points of the training sequences present in that window. Let the time instants of those peaks be denoted by $\{k_1, k_2, \cdots, k_M\}$, arranged in ascending order.[4] The total number of peaks M in a frame depends on whether or not the system is operating in half- or full-rate mode. We will refer to the slot corresponding to k_j as the jth slot.

Beamforming: Consider the jth peak located at time instant k_j corresponding to the ith training sequence, and suppose that it belongs to the lth user. Since there are 14 symbols (28 samples) before the beginning of a training sequence, the beamformer weights computed from the training sequence starting at time k_j will be optimal for data in the jth slot for the interval $[k_j - 28, \min(k_j + 296, k_{j+1} - 28)) \triangleq [l_j, u_j)$. (Recall that there are 162 symbols—and thus 324 $T/2$-spaced samples—in an IS-54 slot.) Beyond the upper limit u_j, the jth slot either ends or the beamformer faces cochannel interference for which it was not trained. When $j = M$ (corresponding to the last peak), the upper limit u_M is set to the last time point of the search window. For each of the M intervals $\{[l_j, u_j)\}$, the beamformer weights $\mathbf{w}_e$ and $\mathbf{w}_o$ are computed using (15)–(18) (with k_j substituted for k in those equations).

[3] This step is reminiscent of the signal cancellation employed in the multistage constant modulus (CM) array, which was designed to separate cochannel analog signals (i.e., AMPS) [19] and continuous (nonbursty) digital sources [20].

[4] Note that k_j does not indicate which training sequence starts at time k_j: only that some training sequence begins there. Although we do not require special notation to indicate which training sequence starts at time k_j, the algorithm does have this information.



(a) (b)

Fig. 6. Forward processing of the data. (a) Frame synchronization (entire search interval). (b) Beamforming (partial).

The resulting beamformer output $y(k)$ over those intervals is now (ideally) free of cochannel interference. Let the output during the jth slot be denoted by $\hat{s}_l(k)$ to emphasize that it represents a discrete-time estimate of the lth baseband transmitted signal. For each slot, the beamformer output samples obtained during the forward pass are stored in a buffer; the remaining output samples will be estimated during the reverse pass.

Reverse Pass: During the reverse pass, the following three steps are performed recursively for $j = M, \cdots, 1$ (in descending order).

Channel Estimation: Using the beamformer output obtained in the forward pass, a least-squares estimate of the channel for every source/antenna combination is computed for the jth slot. Denote the channel estimate from the lth source to the mth antenna element by $\hat{\mathbf{a}}_{ml}$, which is a column vector of length $\hat{M}_l$ equal to some assumed number of propagation paths.[5] Consider a cost function of the form

$$J(\hat{\mathbf{a}}_{ml}) = \sum_{m=1}^{N} \sum_{n=0}^{N_j} |x_m(l_j + n) - \hat{\mathbf{a}}_{ml}^H \hat{\mathbf{s}}_l(l_j + n)|^2 \tag{24}$$

where $\hat{\mathbf{s}}_l(k)$ is a vector of $T/2$-spaced delayed beamformer outputs, i.e., $\hat{\mathbf{s}}_l(k) \triangleq [\hat{s}_l(k), \hat{s}_l(k-1), \cdots, \hat{s}_l(k - \hat{M}_l + 1)]^T$, and N_j is specified below.

The vectors $\{\hat{\mathbf{a}}_{ml}\}$ that minimize (24) optimally cancel $\hat{\mathbf{s}}_l(l_j + n)$ from the antenna samples $\{x_m(l_j + n)\}$. The limits of the second summation would, in general, include as many samples as possible to estimate the channel. For the last few slots in the search interval, there may exist only $u_M - l_j < 324$ beamformer output samples generated in the forward and reverse passes. In Fig. 6(a) (which is discussed below), slot $M = 6$ has only $u_6 - l_6$ beamformer output samples available thus far; since it is the last slot in the search window, these arise from the forward pass alone. For those slots that end completely before the end of the search interval, 324 beamformer output samples from the forward and reverse passes are available. Minimizing (24), we obtain the channel estimate

$$\hat{\mathbf{a}}_{ml} = \left(\sum_{n=0}^{N_j} \hat{\mathbf{s}}_l(l_j + n) \hat{\mathbf{s}}_l^H(l_j + n) \right)^{-1} \cdot \left(\sum_{n=0}^{N_j} \hat{\mathbf{s}}_l(l_j + n) x_m^*(l_j + n) \right) \tag{25}$$

for $m = 1, \cdots, N$, where $N_j = 323$ if the slot ends before the search interval does, and $N_j = u_M - l_j - 1$ otherwise.

Signal Cancellation: The jth channel estimate is used to cancel the beamformer output of slot j (obtained during the forward pass) from the antenna array input over the interval where slots j and $j - 1$ intersect,[6] assuming that this interval is nonempty. Thus, if $u_{j-1} < l_{j-1} + 324$, then for $k \in [u_{j-1}, l_{j-1} + 324)$, the following signals are free of the interfering slot j

$$e_m(k) = x_m(k) - \hat{\mathbf{a}}_{ml}^H \hat{\mathbf{s}}_l(k), \qquad m = 1, \cdots, N. \tag{26}$$

These signals now contain data only from slot $j - 1$.

Beamforming: After the cancellation in slot j has been performed over the interval that it intersects slot $j - 1$, we may process $e_m(k)$ for slot $j - 1$ using the same beamformer weights as those employed during the forward pass. The buffer for slot $j - 1$ is filled for the interval $k \in [u_{j-1}, l_{j-1} + 324)$ using

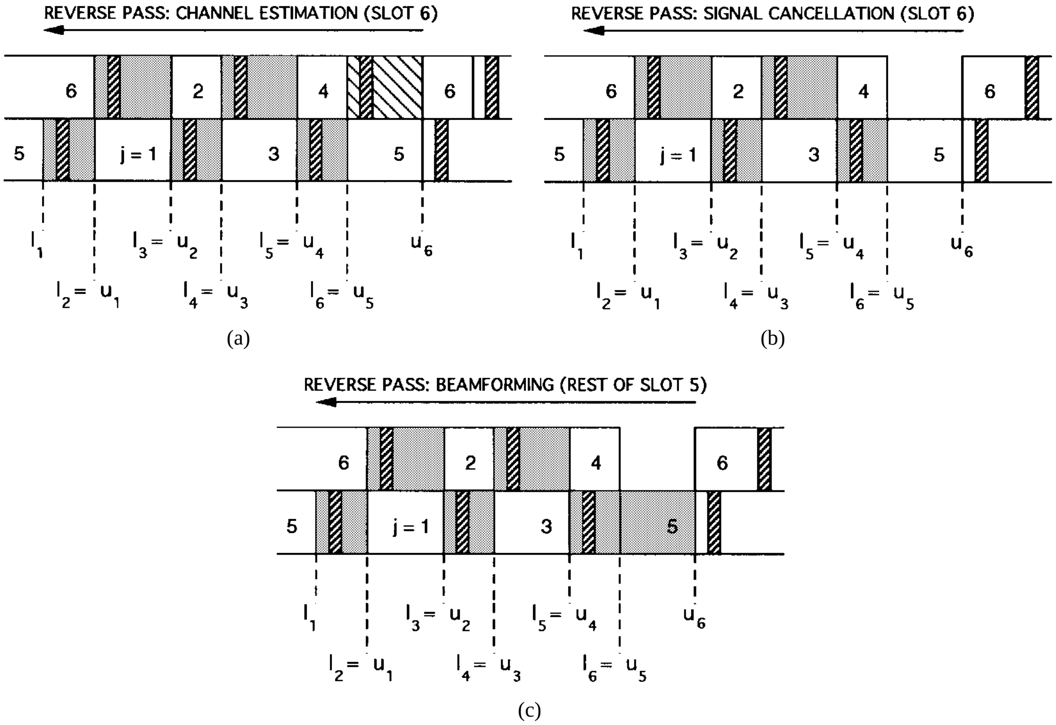
$$y(k) = \begin{cases} \mathbf{w}_e^H \mathbf{e}(k), & k - l_{j-1} \text{ even} \\ \mathbf{w}_o^H \mathbf{e}(k), & k - l_{j-1} \text{ odd} \end{cases} \tag{27}$$

where $\mathbf{e}(k) \triangleq [e_1(k), \cdots, e_N(k)]^T$. Recall that $\mathbf{w}_e$ and $\mathbf{w}_o$ are derived from the training sequence of slot $j - 1$.

ISI Equalization: After the forward and reverse passes have been completed, the beamformer outputs (even and odd) are free of cochannel interference. The linear equalizer then removes the ISI introduced by the transmit filter; the equalizer coefficients are computed using a least-squares method operating on the original training sequence (i.e., the training sequence is not processed by the transmit filter as it was for

[5] Recall from the channel model that α_{mli} is a complex channel coefficient for the ith path of the lth source and the mth antenna element. The vector $\hat{\mathbf{a}}_{ml}$ contains estimates of these coefficients.

[6] If $j = 1$, the cancellation is performed where the first slot of the current frame and the last slot of the previous frame intersect.


REVERSE PASS: CHANNEL ESTIMATION (SLOT 6)
REVERSE PASS: SIGNAL CANCELLATION (SLOT 6)
REVERSE PASS: BEAMFORMING (REST OF SLOT 5)
6
2
4
6
5
j = 1
3
5
l_1
$l_3 = u_2$
$l_5 = u_4$
u_6
$l_2 = u_1$
$l_4 = u_3$
$l_6 = u_5$

(a)

(b)

(c)

Fig. 7. Reverse processing of the data. (a) Channel estimation (slot 6). (b) Signal cancellation (slot 6). (c) Beamforming (rest of slot 5). This procedure continues for the other slots in the frame.

the beamformer weights). The weights of the linear equalizer for the jth slot are computed using

$$\mathbf{h} = \left(\sum_{n=0}^{N_T-1} \mathbf{y}(k_j + 2n)\mathbf{y}^H(k_j + 2n) \right)^{-1} \cdot \left(\sum_{n=0}^{N_T-1} \mathbf{y}(k_j + 2n)t_i^*(n) \right) \tag{28}$$

and the beamformer outputs are combined using (12). The output of the linear equalizer corresponds to an estimate of the transmitted DQPSK symbols.

The steps of the SQ algorithm are summarized below.

- **Forward Pass**: Perform the following two steps:
  a) *Frame Synchronization*: Find the starting points of the training sequences in the search window using the frame synchronization algorithm described in Section IV. Denote these starting time instants by $\{k_1, \cdots, k_M\}$.
  b) *Beamforming*: For $j = 1, \cdots, M$, beamform over the intervals $[l_j, u_j)$ using (15)–(18) (with k_j substituted for k in those equations).
- **Reverse Pass**: For $j = M, \cdots, 1$, perform the following three steps, recursively:
  a) *Channel Estimation*: Estimate the channel of slot j using (25).
  b) *Signal Cancellation*: Cancel slot j over the interval where it intersects slot $j - 1$ (if that interval is nonempty) using (26).
  c) *Beamforming*: Beamform over the interval where slots j and $j - 1$ intersect using the optimal weights previously calculated for slot $j - 1$.
- **ISI Equalization**: Equalize the residual ISI using (12) and (28).

An example of these steps for $M = 6$ cochannel sources is shown in Figs. 6 and 7. During the forward pass of the data, all the intervals $\{[l_j, u_j)\}$ are identified [Fig. 6(a)]. Note that $l_j = u_{j-1}$ (by design) for $M = 2, \cdots, 6$. In this example, l_1 is located at the beginning of slot 1, but u_6 occurs before the end of slot 6. Observe in Fig. 6(b) that although we beamform over the entire frame, only portions of each slot are beamformed (because of the cochannel interference), i.e., interval $[l_j, u_j)$ of slot j is beamformed during the forward pass. This procedure ensures that the same cochannel interference is present over an interval of beamforming (e.g., only source 1 appears over the interval where slot 2 is beamformed).

During the reverse pass in Fig. 7(a), the channel associated with slot 6 is estimated first (the algorithm proceeds from right to left). Because slot 6 extends beyond the search window, only a portion of the data in slot 6 can be used for channel estimation (whereas the entire slot is available for each of the other sources). Once the channel is estimated, we may cancel the beamformed data of slot 6 from the overlapping data in slot 5 [Fig. 7(b)]. As a result, the remaining data in slot 5 are now free of cochannel interference. This data can be beamformed using the weights that were employed for the first part of slot 5 during the forward pass [Fig. 7(c)]. This procedure is then repeated for the other bursts in the search interval.

VI. Computer Simulations

The SQ algorithm has been tested successfully over a wide range of multipath and cochannel interference conditions. In this section, we present two representative examples. In the

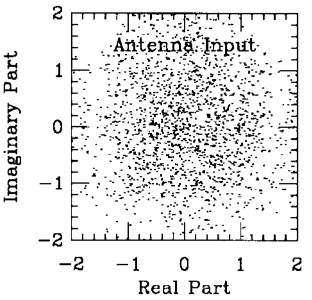


Fig. 8. Data received at one of the antenna elements (before processing).

first scenario, relatively few simultaneous cochannel interferers impinge on a four-element array, whereas in the second scenario, both the number of antenna elements and the number of *simultaneous* cochannel interferers are increased.

We compare the performance of the proposed algorithm with the GR method described in [15], which augments the training sequences with the guard and ramp-up symbols. Since these symbols are invariant for all users, they may be utilized as known training symbols even though they do not have any special correlation properties. The augmented training sequences are given by $\{0, 0, 0, (1/3)e^{j\pi/4}, (2/3)e^{j\pi/4}, e^{j\pi/4}, t_i(0), \cdots, t_i(N_T-1), 0, 0, 0, (1/3)e^{j\pi/4}, (2/3)e^{j\pi/4}, e^{j\pi/4}\}$, where $\{t_i(\cdot)\}$ are the original training sequences.[7] The GR algorithm performs only a forward pass of the data so that all the symbols are demodulated at that time (i.e., partial beamforming, channel estimation, and signal cancellation are not employed). The beamformer weights are computed as in (15), except that the augmented training sequences are employed. The linear equalizer weights are computed using (28).

A. Scenario 1

Table III lists the parameters of the cochannel signals arriving at a four-element linearly spaced antenna array with an interelement spacing of $d = \lambda/2$. The column labeled "Relative delay" indicates the number of symbol durations between the first transmitted symbol of a source and that of source 1. This scenario was presented earlier in Fig. 3. Each source is operating in the full-rate mode and has a single propagation path (ray). The signal-to-noise ratio (SNR) is 20 dB (i.e., $\sigma_n^2 = 0.01$). Since each cochannel signal has unit power, the signal-to-interference-plus-noise-ratio (SINR) for any slot is −3 dB. The SQ and GR algorithms each processed eight frames of data.

The baseband samples received at an antenna element are shown in Fig. 8. Since this result does not resemble the usual DQPSK symbol constellation, it is clear that the level of cochannel interference is high. Both algorithms employ the frame synchronization procedure described in Section IV; the results were shown previously in Fig. 5, where the cross-correlation coefficient $C(k)$ in (20) is plotted as a function of k for each of the six training sequences. A correlation threshold of about 0.9 is sufficient to locate all the bursts. A search interval of three frame durations (120 ms) is used (even though Fig. 5 shows only two frame durations).

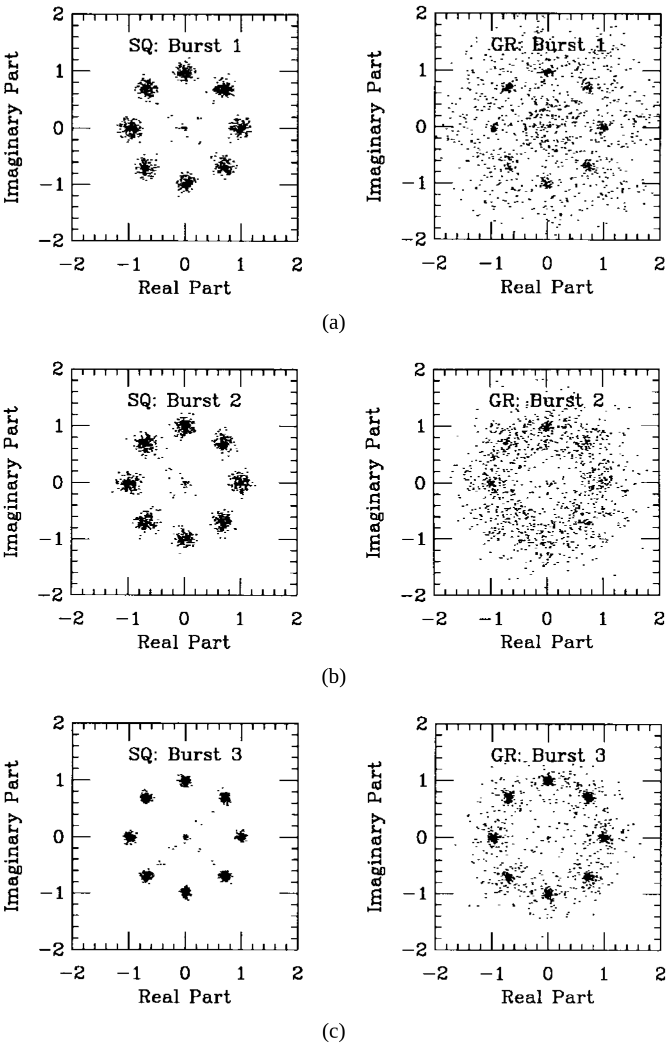


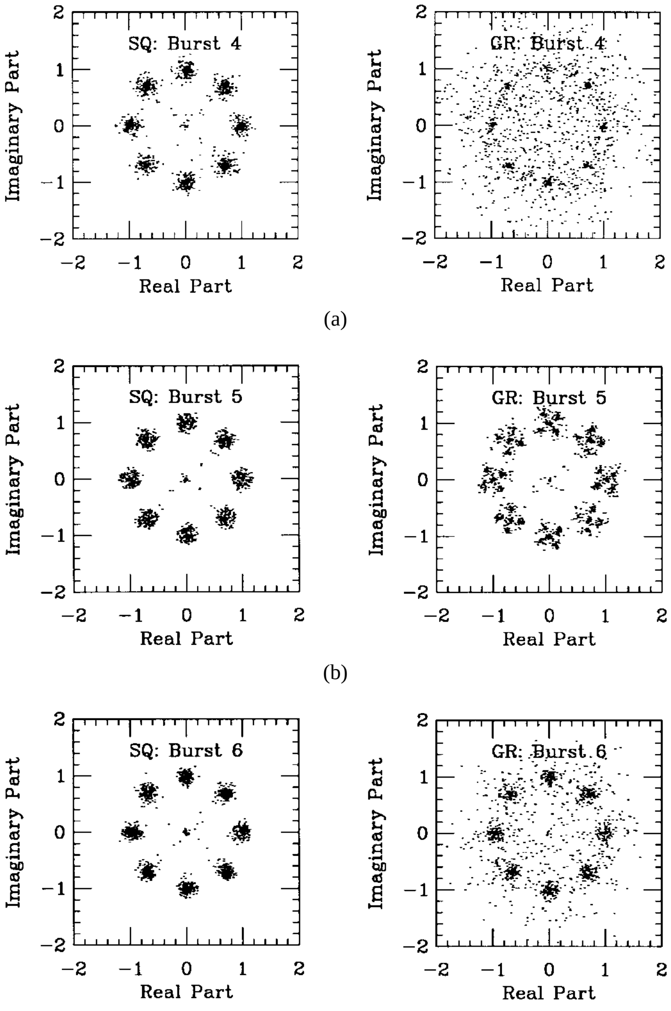
Fig. 9. Signal constellations for the SQ and GR algorithms: Scenario 1. (a) Burst 1. (b) Burst 2. (c) Burst 3.

Figs. 9–11 present the output constellations for both algorithms for all nine signals in Table III. Note that due to the phase uncertainty inherent in DQPSK signals (the information is encoded as the difference in the phases of consecutive symbols), the training sequence stored at the receiver may be offset from the received training sequence by an unknown phase factor. This results in a phase-rotated set of beamformer/equalizer weights, which in turn rotate the signal constellation at the output of the equalizer by an angle equal to an unknown multiple of $\pi/2$ rad. This is manifested in the ramp-up symbols (identified as radial points) of some of the (less noisy) plots in Figs. 9–11.

B. Scenario 2

In the second scenario, the number of simultaneous interferers is increased from three to five, and the number of antenna elements is increased from four to eight. The noise

[7] Note that the ramp-up symbols are defined to have unit amplitude (i.e., $e^{j\pi/4}$); the coefficients {1/3, 2/3, 1} are generated by the carrier amplitude function $A_l(t)$ in Fig. 2.



(a)

(b)

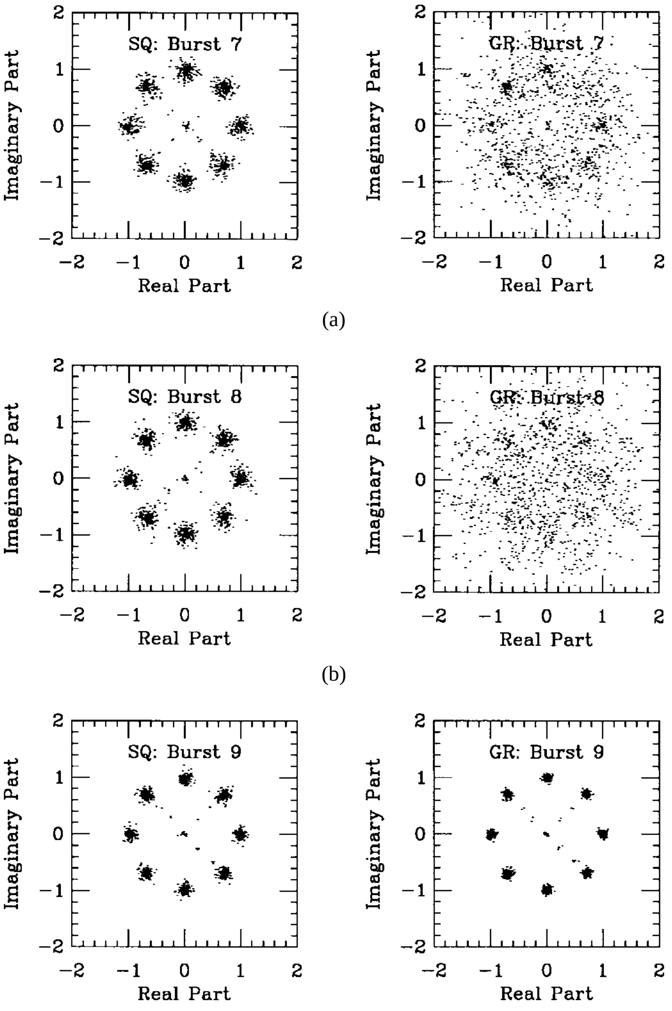
(c)

Fig. 10. Signal constellations for the SQ and GR algorithms: Scenario 1 (*continued*). (a) Burst 4. (b) Burst 5. (c) Burst 6.

power remains the same so that the SINR becomes −6 dB. The relative timings of the five cochannel signals are listed in Table IV. Note that the cochannel interference for a given slot may change even during the training sequence. After successful frame synchronization, the output constellations obtained by the two algorithms are shown in Figs. 12 and 13.

VII. Discussion

From these simulations, it is evident that when the number of antenna elements is relatively low, the SQ algorithm outperforms the GR method. The reasons for this may be explained as follows. In general, it is not possible for an N-element antenna array to simultaneously null more than $N-1$ interferers. This follows from the fact that an N vector can be orthogonal to at most $N-1$ linearly independent vectors. In Scenario 1 (see Fig. 3), each slot is faced with four interferers. The GR method attempts to simultaneously null all four sources by using an augmented training sequence. However, a four-element vector can be orthogonal only to three direction vectors, regardless of the amount of augmentation.



(a)

(b)

(c)

Fig. 11. Signal constellations for the SQ and GR algorithms: Scenario 1 (*continued*). (a) Burst 7. (b) Burst 8. (c) Burst 9.

TABLE IV
Channel Parameters for Scenario 2

Source	Relative delay	Training sequence	Angle of arrival
1	0	1	−75.0°
2	20	2	−45.0°
3	40	3	−15.0°
4	60	4	15.0°
5	80	5	45.0°

The SQ algorithm circumvents this constraint by sequentially removing the interferers. For any given slot, the forward pass nulls two interferers, whereas the reverse pass cancels the remaining two interferers. Thus, the resolution of the array in the “look direction” is improved because the interferers are handled in a sequential manner. The inability of the GR method to cancel all four interferers is confirmed by observing the beampattern for the first slot in Fig. 14(a). Observe that the GR beamformer nulls only two (8 and 9) of the four interferers. The nulls associated with the last slot beamformed by the GR algorithm are well defined only because the source AOA’s

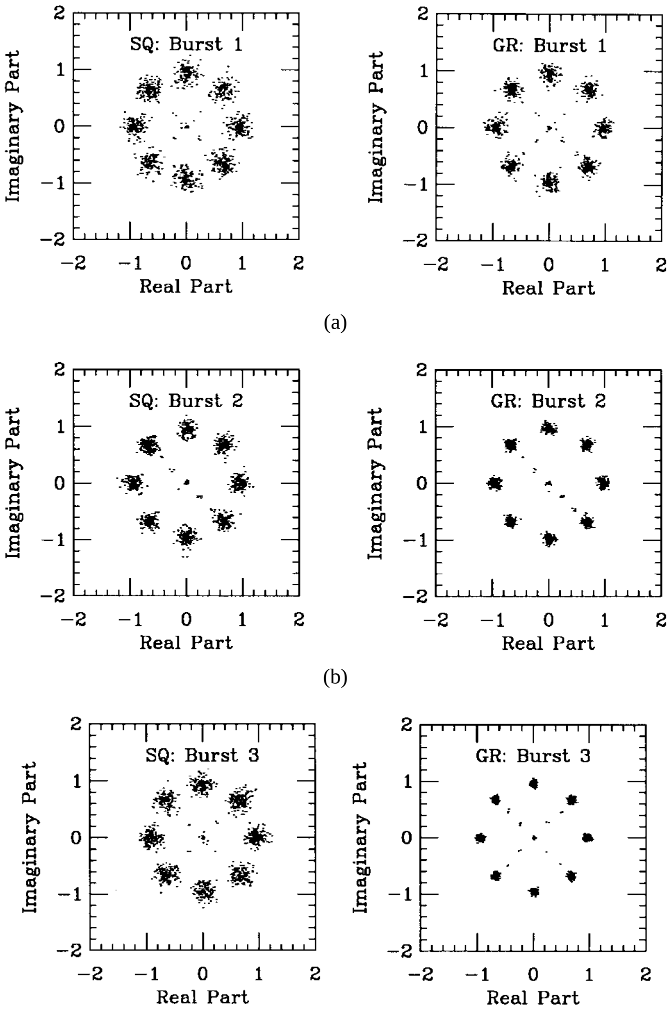


Fig. 12. Signal constellations for the SQ and GR algorithms: Scenario 2. (a) Burst 1. (b) Burst 2. (c) Burst 3.

happen to be in regions of the beampattern that have a low gain [see Fig. 14(b)].

For Scenario 2, there are more antenna elements (eight) than cochannel interferers (five). The GR method demodulates the cochannel signals with a uniform (and satisfactory) performance. The beampattern of slot 4 for the GR algorithm is shown in Fig. 15. Observe that the array strongly attenuates not only in the directions of the interferers but in other directions (because of the larger number of antenna elements).

The performance of the SQ algorithm in Scenario 2 suffers when it attempts to demodulate cochannel signals "buried deep" under the remaining signals. The performance is best for signals 4 and 5 (see Fig. 13), and progressively deteriorates as it reaches signal 1 [Fig. 12(a)]. Note that the output constellations are still well defined, although they are not as "clean" as those of the GR method. This probably occurs because the lower "layers" of the cochannel signals (signals 1 and 2) experience several cancellation steps from the signals above (signals 3, 4, and 5). This repeated signal processing over the samples of the received signals may introduce a type of artificial noise that degrades the performance for signals in the lower cochannel layers, as is evident from the constellation plots.

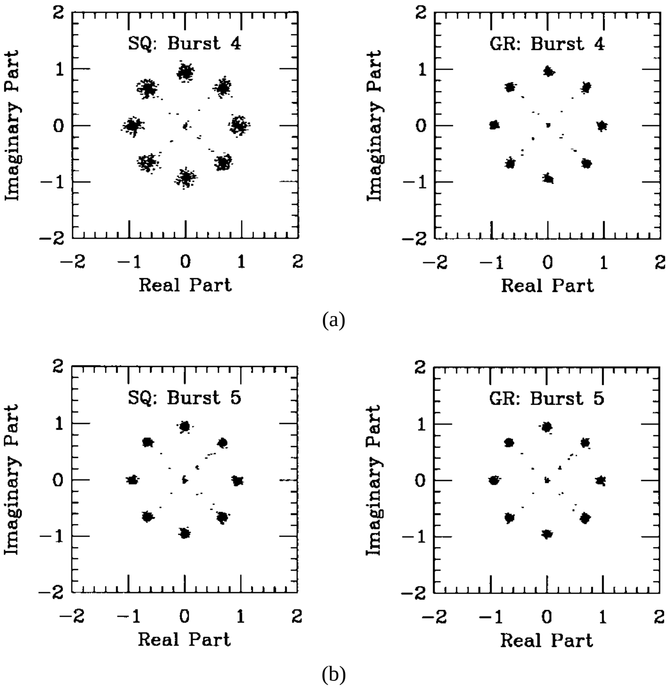


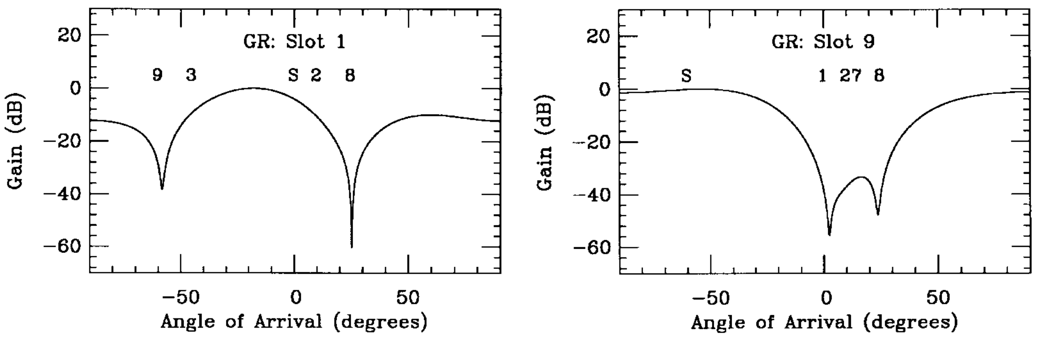
Fig. 13. Signal constellations for the SQ and GR algorithms: Scenario 2 (*continued*). (a) Burst 4. (b) Burst 5.

The number of operations required by both algorithms is dependent on the number of antenna elements N, the number of peaks M detected in the search interval by the frame synchronization algorithm, and the number of feed-forward filter coefficients $2P$. For the SQ algorithm, the complexity also depends on the estimated length $\hat{M}_l$ of the multipath. The actual expressions for the operations count in terms of these variables are fairly cumbersome; however, for practical sizes of the parameters (e.g., $N \leq 8$, $P \leq 3$, and $\hat{M}_l \leq 3$), it turns out that the most dominant terms for both algorithms are linear and (to a lesser extent) of second-degree in N, P, and $\hat{M}_l$. For $N = 8$ antenna elements, $2P = 4$ equalizer weights, and $\hat{M}_l = 2$, the operations count of the SQ algorithm is greater than that of the GR method by a factor of about 1.8. This increased complexity is due to the channel estimation and signal cancellation steps.

Finally, we should mention that the SQ algorithm can experience a loss in performance when there exists a pair of interfering slots such that one of the slots ends at a time point before the training sequence of the other slot. Recall that there is a gap of 14 symbols between the beginning of a slot and the beginning of the training sequence. An extension of the SQ algorithm that handles this type of overlap scenario is described in [21]. This modified algorithm also processes the data in two passes, but it incorporates previously beamformed data from adjacent slots (generated in the forward pass) in the least-squares cost function.

VIII. Conclusion

In this paper, we considered the problem of separating cochannel TDMA signals in a mobile radio environment using a beamformer and an equalizer in the receiver. Training



(a) (b)

Fig. 14. Beampatterns of the GR algorithm for Scenario 1. The signal of interest is denoted by S, whereas the interferers are labeled using the identifying numbers in Table III. (a) Slot 1. (b) Slot 9.

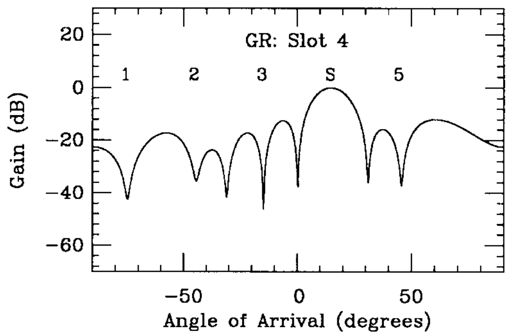


Fig. 15. Beampattern of the GR algorithm for slot 4 of Scenario 2. The signal of interest is denoted by S, whereas the interferers are labeled using the identifying numbers in Table IV.

is employed to compute the weights for each slot using the method of least squares. The main problem with this bursty data format is that the interference may not overlap the training sequence of the current slot, thus yielding a suboptimal beamformer/equalizer. To overcome this problem, we proposed a sequential separation (SQ) algorithm that estimates the data in a noncausal manner using forward and backward processing.

Computer simulations were presented to illustrate the accuracy of the estimates generated by the SQ and GR (guard-ramp) algorithms. The two scenarios illustrate that the GR method fundamentally derives its effectiveness from the beamforming capability of the array and, thus, the corresponding number of antenna elements. On the other hand, the SQ algorithm is capable of operating with fewer antenna elements by using a sequential approach. However, the accumulation of errors due to the channel estimation and signal cancellation steps becomes more noticeable with an increasing number of cochannel sources. Thus, we conclude that there is a tradeoff between the number of antenna elements (hardware) and the number of DSP operations (signal processing). The SQ algorithm is more complex, but it can operate with fewer antenna elements because the sources are captured and removed sequentially.

Currently, we are investigating variants of the basic algorithm and extending it to the frame and time slot structure of GSM. The modulation scheme and the greater bandwidth of GSM require that the sequential algorithm be modified as described in [5]. For fading channels, it may be necessary to incorporate a decision-directed algorithm to adapt the beamformer/equalizer weights and track channel variations.

References

[1] W. C. Y. Lee, *Mobile Cellular Telecommunications Systems.* New York: McGraw-Hill, 1989.

[2] T. S. Rappaport, *Wireless Communications: Principles and Practice.* Upper Saddle River, NJ: Prentice-Hall, 1996.

[3] Electron. Ind. Assoc., *EIA/TIA IS-54: Dual-Mode Mobile Station—Base Station Compatibility Std.*, Oct. 1990.

[4] S. M. Redl, M. K. Weber, and M. W. Oliphant, *An Introduction to GSM.* Norwood, MA: Artech House, 1995.

[5] Y. K. Lee, R. Chandrasekaran, J. J. Shynk, "Separation of cochannel GSM signals using an adaptive array," *IEEE Trans. Signal Processing*, to be published.

[6] M. Kocatürk and S. C. Gupta, "Simulation of cochannel interference in coexisting cellular TDMA networks," *IEEE Trans. Veh. Technol.*, vol. 43, pp. 753–761, Aug. 1994.

[7] I. Korn, "GMSK with frequency-selective Rayleigh fading and cochannel interference," *IEEE J. Select. Areas Commun.*, vol. 10, pp. 506–515, Apr. 1992.

[8] S. S. Shin and P. T. Mathiopoulos, "Differentially detected GMSK signals in CCI channels for mobile cellular telecommunication systems," *IEEE Trans. Veh. Technol.*, vol. 42, pp. 289–293, Aug. 1993.

[9] A. L. Swindlehurst, S. Daas, and J. Yang, "Analysis of a decision directed beamformer," *IEEE Trans. Signal Processing*, vol. 43, pp. 2920–2927, Dec. 1995.

[10] H. Yoshino, K. Fukawa, and H. Suzuki, "Interference cancelling equalizer (ICE) for mobile radio communications," in *Proc. IEEE Int. Conf. Commun.*, New Orleans, LA, May 1994, pp. 1427–1432.

[11] P. A. Ranta, A. Hottinen, and Z.-C. Honkasalo, "Cochannel interference cancelling receiver for TDMA mobile systems," in *Proc. IEEE Int. Conf. Commun.*, Seattle, WA, June 1995, pp. 17–21.

[12] K. Giridhar, J. J. Shynk, A. Mathur, S. Chari, and R. P. Gooch, "Nonlinear techniques for the joint estimation of cochannel signals," *IEEE Trans. Commun.*, vol. 45, pp. 473–484, Apr. 1997.

[13] S. W. Wales, "Techniques for cochannel interference suppression in TDMA mobile radio systems," *Proc. Inst. Elect. Eng. Commun.*, vol. 142, pp. 106–114, Apr. 1995.

[14] E. Lindskog, "Combatting cochannel interferers in a TDMA system using interference estimates from adjacent frames," in *Proc. Twenty-Ninth Asilomar Conf. Signals, Syst., Comput.*, Pacific Grove, CA, Nov. 1995, pp. 367–371.

[15] J. Leary and R. P. Gooch, "Adaptive beamforming for TDMA signals," in *Proc. Twenty-Ninth Asilomar Conf. Signals, Syst., Comput.*, Pacific Grove, CA, Nov. 1995, pp. 1378–1382.

[16] J. G. Proakis, *Digital Communications*, 2nd ed. New York: McGraw-Hill, 1989.

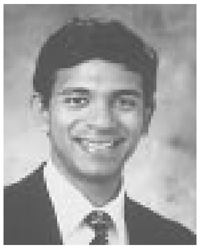
[17] W. C. Jakes, "Multipath interference," in *Microwave Mobile Communications*, W. C. Jakes, Ed. New York: Wiley, 1974, pp. 67–68.

[18] S. Haykin, *Adaptive Filter Theory*, 3rd ed. Upper Saddle River, NJ: Prentice-Hall, 1996.

[19] J. J. Shynk, A. V. Keerthi, and A. Mathur, "Steady-state analysis of the multistage constant modulus array," *IEEE Trans. Signal Processing*, vol. 44, pp. 948–962, Apr. 1996.

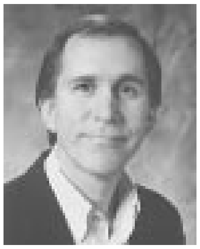
[20] A. V. Keerthi, A. Mathur, and J. J. Shynk, "A multistage CM array for digital communication signals," in *Proc. Thirtieth Conf. Inform. Sci. Syst.*, Princeton, NJ, Mar. 1996, pp. 13–18.

[21] R. Chandrasekaran, A. V. Keerthi, and J. J. Shynk, "A modified sequential beamforming algorithm for IS-136 signals," in *Proc. Thirty-First Asilomar Conf. Signals, Syst., Comput.*, Pacific Grove, CA, Nov. 1997, pp. 1564–1569.



Arvind V. Keerthi was born in Madras, India, on May 10, 1971. He received the B.Tech. degree in electrical engineering from the Indian Institute of Technology, Bombay, in 1992 and the M.S. and Ph.D. degrees in electrical and computer engineering from the University of California, Santa Barbara (UCSB), in 1993 and 1997, respectively.

Since September 1997, he has been employed with QUALCOMM, Inc., San Diego, CA. While at UCSB, his research interests focused on the application of digital signal processing techniques for problems in wireless communications. He was also involved in the design and implementation of radio frequency hardware for a synthetic aperture radar project. His current interests are adaptive filtering, communication theory, electromagnetics, and the design of telecommunications infrastructure in developing countries.



John J. Shynk (S'78–M'86–SM'91) received the B.S. degree in systems engineering from Boston University, Boston, MA, in 1979 and the M.S. degree in electrical engineering and statistics and the Ph.D. degree in electrical engineering from Stanford University, Stanford, CA, in 1980, 1985, and 1987, respectively.

From 1979 to 1982, he was a Member of Technical Staff with the Data Communications Performance Group, AT&T Bell Laboratories, Holmdel, NJ, where he formulated performance models for voiceband data communications. He was a Research Assistant from 1982 to 1986 with the Department of Electrical Engineering, Stanford University, where he worked on frequency-domain implementations of adaptive IIR filter algorithms. From 1985 to 1986, he was also an Instructor at Stanford, teaching courses on digital signal processing and adaptive systems. Since 1987, he has been with the Department of Electrical and Computer Engineering, University of California, Santa Barbara, where he is currently a Professor and Vice Chair. His research interests include adaptive signal processing, adaptive beamforming, wireless communications, direction-of-arrival estimation, and neural networks.

Dr. Shynk has served as an Associate Editor for adaptive filtering for the IEEE TRANSACTIONS ON SIGNAL PROCESSING and is currently an Editor for adaptive signal processing for the *International Journal of Adaptive Control and Signal Processing* and an Associate Editor for the *IEEE Signal Processing Letters*. He was Technical Program Chair of the 1992 International Joint Conference on Neural Networks. He is a member of Eta Kappa Nu, Tau Beta Pi, and Sigma Xi.