

MULTI-STAGE RECEIVER FOR WIRELESS COMMUNICATION

Claim of Priority under 35 U.S.C. §119

[0001] The present Application for Patent claims priority to Provisional Application Serial No. 60/741,159, entitled "Bipartite Receiver Structure for MIMO CDMA Downlinks," filed November 30, 2005, assigned to the assignee hereof, and expressly incorporated herein by reference.

BACKGROUND

Field

[0002] The present disclosure relates generally to communication, and more specifically to techniques for receiving a multiple-input multiple-output (MIMO) transmission.

Background

[0003] A MIMO transmission is a transmission from multiple (M) transmit antennas to multiple (N) receive antennas. For example, a transmitter may simultaneously transmit M data streams from the M transmit antennas. These data streams are distorted by the wireless environment and further degraded by noise and interference. A receiver receives the transmitted data streams via the N receive antennas. The received signal from each receive antenna contains scaled and delayed versions of the transmitted data streams. The transmitted data streams are thus dispersed among the N received signals from the N receive antennas. The receiver may then process the N received signals with a space-time equalizer to recover the transmitted data streams.

[0004] The receiver may dynamically derive coefficients for the space-time equalizer to account for variations in signal properties. These signal properties may relate to channel and interference statistics, spatio-temporal processing of the transmitted data streams, etc. The derivation of the equalizer coefficients is computationally intensive. Updating these equalizer coefficients to match the fastest changes in the signal properties may result in a very complex receiver. Updating these equalizer coefficients at a slower rate may result in performance degradation.

[0005] There is therefore a need in the art for techniques to efficiently receive a MIMO transmission.

SUMMARY

[0006] Techniques for efficiently receiving a MIMO transmission are described herein. A receiver receives multiple signals transmitted by a transmitter and processes multiple received signals to obtain received data. The receiver may process the received data in multiple stages. A first stage may perform filtering/equalization on the received data with a front-end filter to process (e.g., compensate for, suppress, or mitigate) non on-time signal components. A second stage may process filtered data from the first stage with one or more combiner matrices to combine on-time signal components. The on-time and non on-time signal components may be distinguished based on the transmit time. At the receiver, the on-time signal components may comprise signal components tracing back to a desired symbol to be recovered as well as other symbols transmitted at the same time as the desired symbol. The non on-time signal components may comprise signal components that are not on-time signal components, such as signal components tracing back to other symbols transmitted before and after the desired symbol.

[0007] The front-end filter is not dependent on data-specific processing at the transmitter whereas the combiner matrices may be functions of data-specific processing at the transmitter. The data-specific processing may correspond to, e.g., channelization codes, transmit matrices, gains, etc., used to send data. For example, in a MIMO transmission that utilizes code division multiplexing (CDM), a single front-end filter may be derived and used for all channelization codes, and a different combiner matrix may be derived for each channelization code. Partitioning the receiver processing into multiple stages simplifies the derivation of the front-end filter and the combiner matrices while achieving good performance. The front-end filter and combiner matrices may be updated separately at the same or different rates.

[0008] According to an aspect, an apparatus is described which includes at least one processor and a memory. The processor(s) filter received data to process non on-time signal components in multiple received signals and obtain filtered data. The processor(s) also process the filtered data to combine on-time signal components for multiple transmitted signals.

[0009] According to another aspect, an apparatus is described which includes at least one processor and a memory. The processor(s) derive a front-end filter for processing non on-time signal components in multiple received signals and also derive

multiple combiner matrices for combining on-time signal components for multiple transmitted signals sent with multiple channelization codes. The processor(s) filter received samples with the front-end filter and obtain filtered symbols for the multiple channelization codes. The processor(s) further process the filtered symbols for each channelization code with a combiner matrix for the channelization code to obtain output symbols for the channelization code.

[0010] Various aspects and features of the disclosure are described in further detail below.

BRIEF DESCRIPTION OF THE DRAWINGS

- [0011] FIG. 1 shows a block diagram of a transmitter and a receiver.
- [0012] FIG. 2 illustrates a MIMO-CDM transmission.
- [0013] FIG. 3 shows a block diagram of a CDMA modulator.
- [0014] FIG. 4 shows a design of various blocks within the receiver.
- [0015] FIG. 5 shows a process for recovering a MIMO transmission at the receiver.

DETAILED DESCRIPTION

[0016] The receiver processing techniques described herein may be used for various communication systems such as Code Division Multiple Access (CDMA) systems, Time Division Multiple Access (TDMA) systems, Frequency Division Multiple Access (FDMA) systems, Orthogonal FDMA (OFDMA) systems, Single-Carrier FDMA (SC-FDMA) systems, etc. A CDMA system utilizes code division multiplexing (CDM) and transmits modulation symbols in parallel using different channelization codes. A CDMA system may utilize a radio technology such as Wideband-CDMA (W-CDMA), cdma2000, etc. cdma2000 covers IS-2000, IS-856, and IS-95 standards. A TDMA system may utilize a radio technology such as Global System for Mobile Communications (GSM). W-CDMA and GSM are described in documents from an organization named "3rd Generation Partnership Project" (3GPP). cdma2000 is described in documents from an organization named "3rd Generation Partnership Project 2" (3GPP2). 3GPP and 3GPP2 documents are publicly available. An OFDMA system utilizes orthogonal frequency division multiplexing (OFDM) and transmits modulation symbols in the frequency domain on orthogonal subcarriers. An SC-FDMA system utilizes single-carrier frequency division multiplexing (SC-FDM) and transmits modulation symbols in the time domain on orthogonal subcarriers.

[0017] The techniques described herein may also be used for MIMO transmissions on the downlink as well as the uplink. The downlink (or forward link) refers to the communication link from base stations to wireless devices, and the uplink (or reverse link) refers to the communication link from the wireless devices to the base stations. For clarity, the techniques are described below for a MIMO transmission in a CDMA system, which may utilize W-CDMA, cdma2000, or some other CDMA radio technology.

[0018] **FIG. 1** shows a block diagram of a transmitter 110 and a receiver 150 for a MIMO transmission. For downlink transmission, transmitter 110 is part of a base station, and receiver 150 is part of a wireless device. For uplink transmission, transmitter 110 is part of a wireless device, and receiver 150 is part of a base station. A base station is typically a fixed station that communicates with the wireless devices and may also be called a Node B, an access point, etc. A wireless device may be stationary or mobile and may also be called a user equipment (UE), a mobile station, a terminal, a station, a subscriber unit, etc. A wireless device may be a cellular phone, a personal digital assistant (PDA), a wireless modem, a laptop computer, a handheld device, etc.

[0019] At transmitter 110, a transmit data processor (TX Data Proc) 112 receives traffic data and signaling, processes (e.g., encodes, interleaves, and symbol maps) the received data, and provides data symbols. Processor 112 also generates and multiplexes pilot symbols with the data symbols. As used herein, a data symbol is a symbol for traffic data or signaling, a pilot symbol is a symbol for pilot, and a symbol is typically a complex value. The data symbols and pilot symbols may be modulation symbols from a modulation scheme such as PSK or QAM. Pilot is data that is known *a priori* by both the transmitter and receiver. A TX MIMO processor 114 performs spatial or spatio-temporal processing on the data and pilot symbols as described below and provides output symbols to multiple (M) CDMA modulators 116a through 116m. Each CDMA modulator 116 processes its output symbols as described below and provides output chips to an associated transmitter unit (TMTR) 118. Each transmitter unit 118 processes (e.g., converts to analog, amplifies, filters, and frequency upconverts) its output chips and generates a modulated signal. M modulated signals from M transmitter units 118a through 118m are transmitted from M antennas 120a through 120m, respectively.

[0020] At receiver 150, multiple (N) antennas 152a through 152n receive the transmitted signals via various propagation paths in the wireless environment and

provide N received signals to N receiver units (RCVR) 154a through 154n, respectively. Each receiver unit 154 processes (e.g., filters, amplifies, frequency downconverts, and digitizes) its received signal and provides received samples to a channel processor 156 and an equalizer/CDMA demodulator 160. Processor 156 derives coefficients for a front-end filter/equalizer and coefficients for one or more combiner matrices as described below. Unit 160 performs equalization with the front-end filter and CDMA demodulation and provides filtered symbols. A receive (RX) MIMO processor 170 combines the filtered symbols across spatial dimension and provides detected symbols, which are estimates of the transmitted data symbols. An RX data processor 172 processes (e.g., symbol demaps, deinterleaves, and decodes) the detected symbols and provides decoded data. In general, the processing by equalizer/CDMA demodulator 160, RX MIMO processor 170, and RX data processor 172 is complementary to the processing by CDMA modulator 116, TX MIMO processor 114, and TX data processor 112, respectively, at transmitter 110.

[0021] Controllers/processors 130 and 180 direct operation of various processing units at transmitter 110 and receiver 150, respectively. Memories 132 and 182 store data and program codes for transmitter 110 and receiver 150, respectively.

[0022] **FIG. 2** illustrates a MIMO-CDM transmission. For CDM, up to C symbols may be sent simultaneously via one transmit antenna with C channelization codes, where in general $C \geq 1$. These channelization codes may be orthogonal variable spreading factor (OVSF) codes in W-CDMA, Walsh codes in cdma2000, other orthogonal codes or quasi-orthogonal codes, pseudo-random codes, etc. Each channelization code is a specific sequence of chips. The number of chips in the sequence is the length or spreading factor of the channelization code. In general, any set of one or more channelization codes may be used for each transmit antenna, and the channelization codes may have the same or different spreading factors. For simplicity, the following description assumes that the channelization codes have the same spreading factor. The same set of C channelization codes may be reused for each of the M transmit antennas. For MIMO, up to M symbols may be sent simultaneously via M transmit antennas. For MIMO-CDM, up to $C \cdot M$ symbols may be sent simultaneously via M transmit antennas with C channelization codes. MIMO processing may be performed separately for each of the C channelization codes. MIMO processing is performed across all M transmit antennas for each channelization code. CDM

processing may be performed separately for each of the M transmit antennas. CDM processing is performed for all C channelization codes for each transmit antenna.

[0023] FIG. 3 shows a block diagram of a CDMA modulator 116 for one transmit antenna m , where $m \in \{1, \dots, M\}$. CDMA modulator 116 may be used for each of CDMA modulators 116a through 116m in FIG. 1. CDMA modulator 116 includes a data processor 310 for each channelization code used for traffic data and/or signaling and a pilot processor 320 for pilot.

[0024] Within data processor 310, a spreader 312 spreads output symbols $d_{m,c}(s)$ for data with channelization code c having a chip sequence of $v_c(k)$, where s is symbol index and k is chip index. A multiplier 314 scales the output of spreader 312 with a gain $g_{m,c}$ and provides data chips for channelization code c . Within pilot processor 320, a spreader 322 spreads output symbols $d_{m,p}(s)$ for pilot with channelization code p for pilot. A multiplier 324 scales the output of spreader 322 with a gain $g_{m,p}$ and provides pilot chips. The gains $g_{m,c}$ and $g_{m,p}$ determine the amount of transmit power used for channelization code c and pilot, respectively. A summer 330 sums the data and pilot chips for all channelization codes. A scrambler 332 multiplies the output of summer 330 with a scrambling sequence $p(k)$ for transmitter 110 and provides output chips $y_m(k)$ for transmit antenna m .

[0025] In general, any number and any ones of the C channelization codes may be used for each of the M transmit antennas. In one design, the same channelization code is used for pilot for all M transmit antennas. In another design, M channelization codes are used for pilot for the M transmit antennas, and the remaining $C - M$ channelization codes may be reused for each of the M transmit antennas. The same scrambling sequence may be used for all M transmit antennas, as shown in FIG. 3. Alternatively, a different scrambling sequence may be used for each transmit antenna. The spreading and scrambling may also be performed in other manners.

[0026] A MIMO channel is formed by the propagation environment between the M transmit antennas at transmitter 110 and the N receive antennas at receiver 150. L data symbols may be sent in parallel from the M transmit antennas for each channelization code, where $1 \leq L \leq \min\{M, N\}$. Receiver 150 may analyze the performance (e.g., throughput) of the MIMO channel for different values of L (and possibly different

transmit matrices/vectors) and may select the L value (and transmit matrix/vector) that achieves the best performance.

[0027] Transmitter 110 may perform transmitter spatial processing for each channelization code c in each symbol period s , as follows:

$$\underline{\mathbf{d}}_c(s) = \underline{\mathbf{B}}_c \underline{\mathbf{b}}_c(s), \quad \text{for } c = 1, \dots, C, \quad \text{Eq (1)}$$

where $\underline{\mathbf{b}}_c(s) = [b_{1,c}(s) \ b_{2,c}(s) \ \dots \ b_{L,c}(s)]^T$ is an $L \times 1$ vector of data symbols,

$\underline{\mathbf{B}}_c$ is an $L \times M$ transmit matrix for channelization code c ,

$\underline{\mathbf{d}}_c(s) = [d_{1,c}(s) \ d_{2,c}(s) \ \dots \ d_{M,c}(s)]^T$ is an $M \times 1$ vector of output symbols, and

“ T ” denotes a transpose.

[0028] Each element of $\underline{\mathbf{b}}_c(s)$ may correspond to a different data stream. The data streams may have different gains, in which case matrix $\underline{\mathbf{B}}_c$ may have different column norms for different data streams. Equation (1) shows spatial encoding with $\underline{\mathbf{B}}_c$. Spatio-temporal encoding such as, e.g., space-time transmit diversity (STTD) may also be performed but is not shown in equation (1).

[0029] Different transmit matrices may be used for different MIMO modes such as closed loop transmit diversity (CLTD), per antenna rate control (PARC), code reuse Bell Labs layered space-time (CRBLAST), double-transmit adaptive array (D-TXAA), etc. Table 1 lists some MIMO modes and, for each mode, gives L, M, $\underline{\mathbf{B}}_c$ and the source of the data symbols. In Table 1, $\underline{\mathbf{B}}_{cltd}$ may be a 2×1 vector selected from the set $\{[1 \ e^{j\pi/4}]^T, [1 \ e^{-j3\pi/4}]^T, [1 \ e^{j3\pi/4}]^T, [1 \ e^{-j\pi/4}]^T\}$. $\underline{\mathbf{B}}_{d-txaa}$ may be a 2×2 matrix selected from the set $\left\{ \begin{bmatrix} 1 & 1 \\ e^{j\pi/4} & e^{-j3\pi/4} \end{bmatrix}, \begin{bmatrix} 1 & 1 \\ e^{j3\pi/4} & e^{-j\pi/4} \end{bmatrix} \right\}$. $\underline{\mathbf{I}}$ is an identity matrix with ones along the diagonal and zeros elsewhere.

Table 1

MIMO Mode	L	M	$\underline{\mathbf{B}}_c$	Source of Data Symbols
CLTD	1	2	$\underline{\mathbf{B}}_c = \underline{\mathbf{B}}_{cltd}$	From a single encoded frame.
PARC	$L = M$	≥ 2	$\underline{\mathbf{B}}_c = \underline{\mathbf{I}}$	From L different encoded frames.
CRBLAST	$L = M$	≥ 2	$\underline{\mathbf{B}}_c = \underline{\mathbf{I}}$	From a single encoded frame.

D-TXAA	L = M	= 2	$\underline{\mathbf{B}}_c = \underline{\mathbf{B}}_{d-txaa}$	From up to L encoded frames.
--------	-------	-----	--	------------------------------

[0030] Transmitter 110 may perform CDMA processing for each transmit antenna m in each symbol period s , as follows:

$$y_m(k) = \left(\sum_{c=1}^C g_{m,c} \cdot v_c(k \bmod C) \cdot d_{m,c}(k \div C) \right) \cdot p(k), \quad \text{for } m=1, \dots, M, \quad \text{Eq (2)}$$

where symbol period s corresponding to chip period k is given by $s = k \div C = \lfloor k / C \rfloor$.

The gain $g_{m,c}$ may be set equal to zero for each channelization code that is not used.

[0031] If channelization codes with different spreading factors are used, then the CDMA processing for transmit antenna m may be expressed as:

$$y_m(k) = \left(\sum_{c=1}^{N_{pc}} g_{m,c} \cdot v_c(k \bmod C_c) \cdot d_{m,c}(k \div C_c) \right) \cdot p(k),$$

where C_c is the spreading factor of channelization code c , and

N_{pc} is the number of channelization codes used for transmit antenna m .

[0032] For simplicity, the following description assumes that channelization codes with spreading factor of C is used for each transmit antenna. In equation (2), output symbol $d_{m,c}(s)$ is spread with channelization code c with spreading factor C and scaled by gain $g_{m,c}$ to obtain data chips. The spreading is achieved by replicating output symbol $d_{m,c}(s)$ a number of C times and multiplying the C output symbol copies with the $C^{(c)}$ chips $v_c(k)$ of channelization code c . The data and pilot chips for all C channelization codes are summed and further scrambled with scrambling sequence $p(k)$ to obtain output chips $y_m(k)$ for transmit antenna m . The same CDMA processing is performed for each of the M transmit antennas.

[0033] The received samples at receiver 150 in each chip period k may be expressed as:

$$\underline{\mathbf{x}}(k) = \underline{\mathbf{H}} \underline{\mathbf{y}}(k) + \underline{\mathbf{n}}(k), \quad \text{Eq (3)}$$

where $\underline{\mathbf{y}}(k)$ is a $T \times 1$ vector of output chips, where T is described below,

$\underline{\mathbf{H}}$ is an $R \times T$ channel response matrix, where R is described below,

$\underline{\mathbf{x}}(k)$ is an $R \times 1$ vector of received samples, and

$\underline{\mathbf{n}}(k)$ is an $R \times 1$ noise vector.

[0034] Receiver 150 may digitize the received signal from each receive antenna at K times the chip rate, where K is an oversampling ratio and in general $K \geq 1$. In each chip period k , receiver 150 may obtain $E \cdot K$ samples from each receiver 154 and form $\underline{\mathbf{x}}(k)$ by stacking $N \cdot E \cdot K$ samples from N receivers 154a through 154n. E is the length of the front-end equalizer at receiver 150, in number of chips. In general, $E \geq 1$ and may be selected based on a tradeoff between receiver complexity and performance. $\underline{\mathbf{x}}(k)$ includes R received samples from N receive antennas for E chip periods, where $R = N \cdot E \cdot K$.

[0035] Matrix $\underline{\mathbf{H}}$ contains time-domain channel impulse responses for all transmit and receive antenna pairs. As shown in FIG. 1, there is a propagation channel between each transmit antenna and each receive antenna, or a total of $M \cdot N$ propagation channels between the M transmit antennas and N receive antennas. Each propagation channel has a particular impulse response determined by the wireless environment. The response of a single-input multiple-output (SIMO) channel between each transmit antenna m and the N receive antennas may be given by an $R \times T_m$ submatrix $\underline{\mathbf{H}}_m$. The number of rows in $\underline{\mathbf{H}}_m$ is determined by the number of entries in $\underline{\mathbf{x}}(k)$. The number of columns in $\underline{\mathbf{H}}_m$ is determined by the equalizer length E as well as the time span of the impulse responses between transmit antenna m and the N receive antennas. T_m may be given as follows:

$$T_m = \left\lceil E + \max_n \{ \ell_{m,n} \} \right\rceil, \quad \text{Eq (4)}$$

where $\ell_{m,n}$ is the time span of the impulse response between transmit antenna m and receive antenna n , in number of chips.

[0036] Matrix $\underline{\mathbf{H}}$ is composed of M submatrices $\underline{\mathbf{H}}_m$, for $m = 1, \dots, M$, as follows:

$$\underline{\mathbf{H}} = [\underline{\mathbf{H}}_1 \ \underline{\mathbf{H}}_2 \ \dots \ \underline{\mathbf{H}}_M]. \quad \text{Eq (5)}$$

$\underline{\mathbf{H}}$ has a dimension of $R \times T$, where $T = T_1 + T_2 + \dots + T_M$.

[0037] Vector $\underline{\mathbf{y}}(k)$ is composed of M subvectors $\underline{\mathbf{y}}_m(k)$, for $m = 1, \dots, M$, for the M transmit antennas. Each subvector $\underline{\mathbf{y}}_m(k)$ includes T_m output chips from one transmit antenna m centered at chip period k . Vector $\underline{\mathbf{y}}(k)$ and subvector $\underline{\mathbf{y}}_m(k)$ may be expressed as:

$$\underline{\mathbf{y}}(k) = \begin{bmatrix} \underline{\mathbf{y}}_1(k) \\ \underline{\mathbf{y}}_2(k) \\ \vdots \\ \underline{\mathbf{y}}_M(k) \end{bmatrix} \quad \text{and} \quad \underline{\mathbf{y}}_m(k) = \begin{bmatrix} y_m(k - \lfloor T_m/2 - 1 \rfloor) \\ \vdots \\ y_m(k) \\ \vdots \\ y_m(k + \lceil T_m/2 - 1 \rceil) \end{bmatrix}. \quad \text{Eq (6)}$$

[0038] Equation (3) may also be expressed as:

$$\underline{\mathbf{x}}(k) = \sum_{m=1}^M \underline{\mathbf{H}}_m \underline{\mathbf{y}}_m(k) + \underline{\mathbf{n}}(k). \quad \text{Eq (7)}$$

[0039] For the model shown in equation (7), in each chip period k , T_m output chips are sent from each transmit antenna m and via a SIMO channel with a response of $\underline{\mathbf{H}}_m$ to the N receive antennas. The received samples in $\underline{\mathbf{x}}(k)$ include contributions from all M transmit antennas. $\underline{\mathbf{x}}(k)$, $\underline{\mathbf{y}}(k)$, and $\underline{\mathbf{H}}$ may be relatively large. As an example, with $M = 2$, $N = 2$, $K = 2$, $E = 20$, $T = 48$, and $R = 80$, $\underline{\mathbf{y}}(k)$ would be a 48×1 vector, $\underline{\mathbf{H}}$ would be an 80×48 matrix, and $\underline{\mathbf{x}}(k)$ would be an 80×1 vector.

[0040] The noise may be assumed to be stationary complex random vector with

$$E \{ \underline{\mathbf{n}}(k) \} = \underline{\mathbf{0}}, \quad \text{and} \quad \text{Eq (8)}$$

$$E \{ \underline{\mathbf{n}}(k) \underline{\mathbf{n}}^H(k) \} = \underline{\mathbf{R}}_{nn}, \quad \text{Eq (9)}$$

where $E \{ \}$ is an expectation operation, $\underline{\mathbf{0}}$ is a vector of all zeros, $\underline{\mathbf{R}}_{nn}$ is an $R \times R$ noise covariance matrix, and “ ^{H} ” denotes a conjugate transpose. Equations (8) and (9) indicate that the noise has zero mean and a covariance matrix of $\underline{\mathbf{R}}_{nn}$.

[0041] Receiver 150 may recover the data symbols in $\underline{\mathbf{h}}_c(s)$ for each channelization code c by filtering the received samples in $\underline{\mathbf{x}}(k)$ with a bank of L filters for

channelization code c and then despreading and descrambling the filtered samples, as follows:

$$\begin{aligned}
 \hat{\mathbf{b}}_c(s) &= \frac{1}{\sqrt{C}} \cdot \sum_{k=sC}^{(s+1)C-1} \mathbf{W}_c^H \mathbf{x}(k) \cdot [v_c(k \bmod C) \cdot p(k)]^* \\
 &= \mathbf{W}_c^H \left(\frac{1}{\sqrt{C}} \cdot \sum_{k=sC}^{(s+1)C-1} [\mathbf{H} \mathbf{y}(k) + \mathbf{n}(k)] \cdot [v_c(k \bmod C) \cdot p(k)]^* \right) \\
 &= \mathbf{W}_c^H [\mathbf{H} \mathbf{\Theta}_c(s) + \mathbf{n}_c(s)] \\
 &= \mathbf{W}_c^H \mathbf{\chi}_c(s)
 \end{aligned} \tag{Eq 10}$$

$$\text{where } \mathbf{\Theta}_c(s) = \left(\frac{1}{\sqrt{C}} \cdot \sum_{k=sC}^{(s+1)C-1} \mathbf{y}(k) \cdot [v_c(k \bmod C) \cdot p(k)]^* \right), \tag{Eq 11}$$

$$\mathbf{n}_c(s) = \left(\frac{1}{\sqrt{C}} \cdot \sum_{k=sC}^{(s+1)C-1} \mathbf{n}(k) \cdot [v_c(k \bmod C) \cdot p(k)]^* \right), \tag{Eq 12}$$

$$\mathbf{\chi}_c(s) = \left(\frac{1}{\sqrt{C}} \cdot \sum_{k=sC}^{(s+1)C-1} \mathbf{x}(k) \cdot [v_c(k \bmod C) \cdot p(k)]^* \right) = \mathbf{H} \mathbf{\Theta}_c(s) + \mathbf{n}_c(s), \tag{Eq 13}$$

$\mathbf{W}_c$ is an $L \times R$ overall filter for channelization code c ,

$\hat{\mathbf{b}}_c(s)$ is an estimate of $\mathbf{b}_c(s)$, and

“*” denotes a complex conjugate.

[0042] $\mathbf{\Theta}_c(s)$ is a $T \times 1$ vector of despread symbols for channelization code c and is obtained based on the transmitted chips. $\mathbf{n}_c(s)$ is an $R \times 1$ noise vector for channelization code c after descrambling and despreading. $\mathbf{n}_c(s)$ conserves the statistics of $\mathbf{n}(k)$, which are independent of channelization code c . $\mathbf{\chi}_c(s)$ is an $R \times 1$ vector of despread symbols for channelization code c and is obtained based on the received samples. $\mathbf{W}_c$ includes the bank of L filters for channelization code c . Equation (10) indicates that the processing with $\mathbf{W}_c$ may equivalently be performed on the symbols in $\mathbf{\chi}_c(s)$ instead of the received samples in $\mathbf{x}(k)$.

[0043] Filter $\mathbf{W}_c$ may be a Wiener filter, which may be derived as:

$$\begin{aligned}
\underline{\mathbf{W}}_c &= E \{ \underline{\chi}_c(s) \underline{\chi}_c^H(s) \}^{-1} \cdot E \{ \underline{\chi}_c(s) \underline{\mathbf{b}}_c^H(s) \} \\
&= [\underline{\mathbf{H}} E \{ \underline{\Theta}_c(s) \underline{\Theta}_c^H(s) \} \underline{\mathbf{H}}^H + \underline{\mathbf{R}}_{nn}]^{-1} \cdot [\underline{\mathbf{H}} E \{ \underline{\Theta}_c(s) \underline{\mathbf{b}}_c^H(s) \}] \\
&= [\underline{\mathbf{H}} \underline{\Gamma}_c \underline{\mathbf{H}}^H + \underline{\mathbf{R}}_{nn}]^{-1} \underline{\mathbf{H}} \underline{\gamma}_c \\
&= \underline{\mathbf{R}}_c^{-1} \underline{\mathbf{H}} \underline{\gamma}_c = \underline{\mathbf{R}}_c^{-1} \underline{\Phi}_c
\end{aligned} \tag{Eq 14}$$

$$\text{where } \underline{\Gamma}_c = E \{ \underline{\Theta}_c(s) \underline{\Theta}_c^H(s) \}, \tag{Eq 15}$$

$$\underline{\gamma}_c = E \{ \underline{\Theta}_c(s) \underline{\mathbf{b}}_c^H(s) \}, \text{ and} \tag{Eq 16}$$

$$\underline{\mathbf{R}}_c = \underline{\mathbf{H}} \underline{\Gamma}_c \underline{\mathbf{H}}^H + \underline{\mathbf{R}}_{nn}. \tag{Eq 17}$$

[0044] $\underline{\Gamma}_c$ is a $T \times T$ covariance matrix of $\underline{\Theta}_c(s)$. $\underline{\gamma}_c$ is a $T \times L$ correlation matrix of $\underline{\Theta}_c(s)$ and $\underline{\mathbf{b}}_c(s)$ and is indicative of the transmitted signal characteristic. $\underline{\mathbf{R}}_c$ is an $R \times R$ correlation matrix for the wireless channel, signal, and noise statistics. Matrices $\underline{\Gamma}_c$, $\underline{\gamma}_c$, and $\underline{\mathbf{R}}_c$ are dependent on channelization code c through transmit matrix $\underline{\mathbf{B}}_c$ and gain $g_{m,c}$.

[0045] Matrix $\underline{\Gamma}_c$ may be expressed as:

$$\underline{\Gamma}_c = \tilde{\underline{\Gamma}} + \underline{\gamma}_c \underline{\gamma}_c^H, \tag{Eq 18}$$

where $\tilde{\underline{\Gamma}}$ is a part of $\underline{\Gamma}_c$ that is not dependent on channelization code c . It can be shown that only M elements of $\underline{\gamma}_c$ are non-zero. The indices of the non-zero elements of $\underline{\gamma}_c$ are such that the $R \times L$ matrix $\underline{\Phi}_c$ may be expressed as:

$$\underline{\mathbf{H}} \underline{\gamma}_c = \underline{\Phi}_c = \tilde{\underline{\mathbf{H}}} \underline{\mathbf{G}}_c \underline{\mathbf{B}}_c, \tag{Eq 19}$$

where $\tilde{\underline{\mathbf{H}}}$ is an $R \times M$ matrix containing M “on-time” columns of $\underline{\mathbf{H}}$, and

$\underline{\mathbf{G}}_c$ is an $M \times M$ diagonal matrix containing $\sqrt{C} \cdot g_{m,c}$, for $m = 1, \dots, M$, along the diagonal and zeros elsewhere.

[0046] As shown in equation (5), $\underline{\mathbf{H}}$ contains $T = T_1 + T_2 + \dots + T_M$ columns for the M transmit antennas. $\tilde{\underline{\mathbf{H}}}$ contains M columns of $\underline{\mathbf{H}}$, or one column for each transmit antenna. The M columns of $\tilde{\underline{\mathbf{H}}}$ multiply the M elements in $\underline{\mathbf{y}}(k)$ that have their chip

time aligned with the despreader operation in equation (10). These M columns may be identified as described below.

[0047] Matrix $\underline{\mathbf{R}}_c$ in equation (17) may then be expressed as:

$$\begin{aligned}\underline{\mathbf{R}}_c &= \underline{\mathbf{H}} \underline{\Gamma}_c \underline{\mathbf{H}}^H + \underline{\mathbf{R}}_{nn} \\ &= \underline{\mathbf{H}} \underline{\tilde{\Gamma}} \underline{\mathbf{H}}^H + \underline{\mathbf{R}}_{nn} + \underline{\mathbf{H}} \underline{\gamma}_c \underline{\gamma}_c^H \underline{\mathbf{H}}^H \\ &= \underline{\tilde{\mathbf{R}}} + \underline{\Phi}_c \underline{\Phi}_c^H\end{aligned}\tag{Eq (20)}$$

$$\text{where } \underline{\tilde{\mathbf{R}}} = \underline{\mathbf{H}} \underline{\tilde{\Gamma}} \underline{\mathbf{H}}^H + \underline{\mathbf{R}}_{nn}.\tag{Eq (21)}$$

[0048] Matrix $\underline{\mathbf{W}}_c$ in equation (14) may then be expressed as:

$$\begin{aligned}\underline{\mathbf{W}}_c &= \underline{\mathbf{R}}_c^{-1} \underline{\tilde{\mathbf{H}}} \underline{\mathbf{G}}_c \underline{\mathbf{B}}_c \\ &= [\underline{\tilde{\mathbf{R}}} + \underline{\Phi}_c \underline{\Phi}_c^H]^{-1} \underline{\tilde{\mathbf{H}}} \underline{\mathbf{G}}_c \underline{\mathbf{B}}_c \\ &= \underline{\tilde{\mathbf{R}}}^{-1} \underline{\tilde{\mathbf{H}}} \underline{\mathbf{G}}_c \underline{\mathbf{B}}_c (\underline{\mathbf{I}} + \underline{\mathbf{B}}_c^H \underline{\mathbf{G}}_c \underline{\tilde{\mathbf{H}}}^H \underline{\tilde{\mathbf{R}}}^{-1} \underline{\tilde{\mathbf{H}}} \underline{\mathbf{G}}_c \underline{\mathbf{B}}_c)^{-1} \\ &= \underline{\mathbf{F}} \underline{\Delta}_c\end{aligned}\tag{Eq (22)}$$

$$\text{where } \underline{\mathbf{F}} = \underline{\tilde{\mathbf{R}}}^{-1} \underline{\tilde{\mathbf{H}}}, \text{ and}\tag{Eq (23)}$$

$$\underline{\Delta}_c = \underline{\mathbf{G}}_c \underline{\mathbf{B}}_c (\underline{\mathbf{I}} + \underline{\mathbf{B}}_c^H \underline{\mathbf{G}}_c \underline{\tilde{\mathbf{H}}}^H \underline{\tilde{\mathbf{R}}}^{-1} \underline{\tilde{\mathbf{H}}} \underline{\mathbf{G}}_c \underline{\mathbf{B}}_c)^{-1}.\tag{Eq (24)}$$

[0049] In equation (23), $\underline{\mathbf{F}}$ is a relatively large $R \times M$ matrix that is not dependent on channelization code. In equation (24), $\underline{\Delta}_c$ is a small $M \times L$ matrix that contains all of the code dependent matrices in $\underline{\mathbf{W}}_c$.

[0050] Equations (10) through (24) indicate that the processing at receiver 150 may be performed in two stages. The first stage filters the received samples $\underline{\mathbf{x}}(k)$ with a front-end filter $\underline{\mathbf{F}}$ that is not dependent on channelization code and further despreads and descrambles the filtered samples to obtain filtered symbols. A single front-end filter may be used for all channelization codes. The second stage combines the filtered symbols with a combiner matrix $\underline{\Delta}_c$ for each channelization code c to obtain detected symbols for that channelization code. The front-end filter and combiner matrices may be updated separately at the same rate or different rates.

[0051] The multi-stage receiver processing may be performed in various manners. In the following description, pilot symbols are assumed to be sent with a transmit matrix

of $\underline{\mathbf{B}}_c = \underline{\mathbf{I}}$ and using the same channelization code p for each of the M transmit antennas. The pilot symbols are also assumed to be uncorrelated or orthogonal so that $E \{ \underline{\mathbf{b}}_p(s) \underline{\mathbf{b}}_p^H(s) \} = \underline{\mathbf{I}}$, where $\underline{\mathbf{b}}_p(s)$ is an $M \times 1$ vector of pilot symbols sent from the M transmit antennas in symbol period s .

[0052] In one receiver design, front-end filter $\underline{\mathbf{F}}$ is derived and used for the first stage (e.g., for block 160 in FIG. 1), and combiner matrix $\underline{\Delta}_c$ is computed for each channelization code and used for the second stage (e.g., for block 170 in FIG. 1).

[0053] For symbol level training, a filter may be derived based on pilot symbols using least squares criterion, as follows:

$$\underline{\mathbf{W}}_p = \min_{\underline{\mathbf{W}}_h} \sum_{s=s_0}^{s_0+P-1} \| \underline{\mathbf{W}}_h^H \underline{\chi}_p(s) - \underline{\mathbf{b}}_p(s) \|^2, \quad \text{Eq (25)}$$

where $\underline{\chi}_p(s)$ is an $R \times 1$ vector of despread pilot symbols,

$\underline{\mathbf{W}}_h$ is an $R \times M$ candidate filter matrix,

$\underline{\mathbf{W}}_p$ is an $R \times M$ filter matrix derived based on pilot symbols, and

P is the number pilot symbols used to derive $\underline{\mathbf{W}}_p$.

[0054] The despread pilot symbols in $\underline{\chi}_p(s)$ may be obtained as shown in equation (13), albeit with pilot channelization code p instead of channelization code c . The least squares criterion selects a candidate filter matrix from among all candidate filter matrices such that the quantity on the right hand side is minimized. The selected filter matrix is provided as the least squares filter matrix $\underline{\mathbf{W}}_p$.

[0055] In the limit, with $P \rightarrow \infty$ in equation (25), $\underline{\mathbf{W}}_p$ may be expressed as:

$$\underline{\mathbf{W}}_p = [E \{ \underline{\chi}_p(s) \underline{\chi}_p^H(s) \}]^{-1} E \{ \underline{\chi}_p(s) \underline{\mathbf{b}}_p^H(s) \}. \quad \text{Eq (26)}$$

[0056] For symbol level training, $\underline{\mathbf{W}}_p$ may be derived as follows. Despread pilot symbols in $\underline{\chi}_p(s)$ may be obtained from the received samples as shown in equation (13). An $R \times R$ outer product $\underline{\chi}_p(s) \underline{\chi}_p^H(s)$ may be computed and averaged over a sufficient number of pilot symbols. An $R \times M$ outer product $\underline{\chi}_p(s) \underline{\mathbf{b}}_p^H(s)$ may also be

computed and averaged. $\underline{\mathbf{W}}_p$ may be computed based on the two averaged outer products. $\underline{\mathbf{W}}_p$ may also be derived based on recursive least squares (RLS), block least squares, or some other techniques known in the art.

[0057] For chip level training, a filter may be derived based on received samples using the least squares criterion, as follows:

$$\underline{\mathbf{W}}_p = \min_{\underline{\mathbf{W}}_h} \sum_{k=s_0}^{(s_0+P)C-1} \|\underline{\mathbf{W}}_h^H \underline{\mathbf{x}}(k) - \underline{\mathbf{b}}_p(s) \cdot v_p(k \bmod C) \cdot p(k)\|^2, \quad \text{Eq (27)}$$

where $\underline{\mathbf{b}}_p(s) \cdot v_p(k) \cdot p(k)$ is an $M \times 1$ vector of pilot chips obtained by spreading and scrambling the pilot symbols.

[0058] In the limit, with $P \rightarrow \infty$ in equation (27), $\underline{\mathbf{W}}_p$ may be expressed as:

$$\underline{\mathbf{W}}_p = [E \{ \underline{\mathbf{x}}(k) \underline{\mathbf{x}}^H(k) \}]^{-1} E \{ \underline{\mathbf{x}}(k) \underline{\mathbf{b}}_p^H(s) \cdot v_p(k) \cdot p(k) \}. \quad \text{Eq (28)}$$

[0059] For chip level training, $\underline{\mathbf{W}}_p$ may be derived as follows. An $R \times R$ outer product $\underline{\mathbf{x}}(k) \underline{\mathbf{x}}^H(k)$ may be computed based on the received samples and averaged over a sufficient number of pilot symbols. An $R \times M$ outer product $\underline{\mathbf{x}}(k) \underline{\mathbf{b}}_p^H(s) \cdot v_p(k) \cdot p(k)$ may also be computed and averaged. $\underline{\mathbf{W}}_p$ may then be computed based on the two averaged outer products.

[0060] From equation (22), $\underline{\mathbf{W}}_p$ may be expressed as:

$$\begin{aligned} \underline{\mathbf{W}}_p &= \tilde{\mathbf{R}}^{-1} \tilde{\mathbf{H}} \underline{\mathbf{G}}_p (\mathbf{I} + \underline{\mathbf{G}}_p \tilde{\mathbf{H}}^H \tilde{\mathbf{R}}^{-1} \tilde{\mathbf{H}} \underline{\mathbf{G}}_p)^{-1} \\ &= \underline{\mathbf{F}} \underline{\mathbf{G}}_p (\mathbf{I} + \underline{\mathbf{P}}_p)^{-1} \end{aligned} \quad \text{Eq (29)}$$

$$\text{where } \underline{\mathbf{P}}_p = \underline{\mathbf{G}}_p \tilde{\mathbf{H}}^H \tilde{\mathbf{R}}^{-1} \tilde{\mathbf{H}} \underline{\mathbf{G}}_p. \quad \text{Eq (30)}$$

[0061] The $M \times M$ matrix $\underline{\mathbf{P}}_p$ may be estimated as follows:

$$\underline{\mathbf{P}}_p = \frac{1}{P} \cdot \sum_{s=s_0}^{s_0+P-1} \underline{\mathbf{F}}_p^H \underline{\mathbf{z}}_p(s) \underline{\mathbf{b}}_p^H(s). \quad \text{Eq (31)}$$

The estimate of $\underline{\mathbf{P}}_p$ becomes more accurate in the limit, with $P \rightarrow \infty$. Similar to $\underline{\mathbf{W}}_p$, $\underline{\mathbf{P}}_p$ may also be estimated based on pilot chips instead of pilot symbols.

[0062] Front-end filter $\underline{\mathbf{F}}$ may be derived based on equation (29), as follows:

$$\underline{\mathbf{F}} = \underline{\mathbf{W}}_p (\underline{\mathbf{I}} + \underline{\mathbf{P}}_p) \underline{\mathbf{G}}_p^{-1}. \quad \text{Eq (32)}$$

[0063] Combiner matrix $\underline{\Delta}_c$ may be derived based on equations (24) and (30), as follows:

$$\underline{\Delta}_c = \underline{\mathbf{G}}_c \underline{\mathbf{B}}_c (\underline{\mathbf{I}} + \underline{\mathbf{B}}_c^H \underline{\mathbf{G}}_c \underline{\mathbf{G}}_p^{-1} \underline{\mathbf{P}}_p \underline{\mathbf{G}}_p^{-1} \underline{\mathbf{G}}_c \underline{\mathbf{B}}_c)^{-1}. \quad \text{Eq (33)}$$

[0064] Filter $\underline{\mathbf{W}}_c$ may be expressed based on $\underline{\mathbf{W}}_p$ and $\underline{\mathbf{P}}_p$, as follows:

$$\begin{aligned} \underline{\mathbf{W}}_c &= \underline{\mathbf{F}} \underline{\mathbf{G}}_c \underline{\mathbf{B}}_c (\underline{\mathbf{I}} + \underline{\mathbf{B}}_c^H \underline{\mathbf{G}}_c \underline{\tilde{\mathbf{H}}}^H \underline{\tilde{\mathbf{R}}}^{-1} \underline{\tilde{\mathbf{H}}} \underline{\mathbf{G}}_c \underline{\mathbf{B}}_c)^{-1} \\ &= \underline{\mathbf{W}}_p (\underline{\mathbf{I}} + \underline{\mathbf{P}}_p) \underline{\mathbf{G}}_p^{-1} \underline{\mathbf{G}}_c \underline{\mathbf{B}}_c (\underline{\mathbf{I}} + \underline{\mathbf{B}}_c^H \underline{\mathbf{G}}_c \underline{\mathbf{G}}_p^{-1} \underline{\mathbf{P}}_p \underline{\mathbf{G}}_p^{-1} \underline{\mathbf{G}}_c \underline{\mathbf{B}}_c)^{-1}. \end{aligned} \quad \text{Eq (34)}$$

[0065] As shown in equation (32), front-end filter $\underline{\mathbf{F}}$ may be derived based on $\underline{\mathbf{W}}_p$ and $\underline{\mathbf{P}}_p$, which may in turn be estimated based on pilot symbols or chips. As shown in equation (33), combiner matrix $\underline{\Delta}_c$ for each channelization code c may be derived based on $\underline{\mathbf{P}}_p$, gain matrices $\underline{\mathbf{G}}_p$ and $\underline{\mathbf{G}}_c$ for pilot and data, and transmit matrix $\underline{\mathbf{B}}_c$ for channelization code c . $\underline{\mathbf{G}}_c \underline{\mathbf{G}}_p^{-1}$ is also referred to as a traffic-to-pilot ratio and may be known (e.g., via signaling) or estimated by the receiver. It is normally sufficient to estimate the traffic-to-pilot ratio $\underline{\mathbf{G}}_c \underline{\mathbf{G}}_p^{-1}$, and $\underline{\mathbf{G}}_p$ and $\underline{\mathbf{G}}_c$ do not need to be estimated separately.

[0066] Receiver 150 may recover the data symbols in $\underline{\mathbf{b}}_c(s)$ as follows:

$$\hat{\underline{\mathbf{b}}}_c(s) = \underline{\Delta}_c^H \left(\frac{1}{\sqrt{C}} \cdot \sum_{k=sC}^{(s+1)C-1} \underline{\mathbf{F}}^H \underline{\mathbf{x}}(k) \cdot [v_c(k \bmod C) \cdot p(k)]^* \right). \quad \text{Eq (35)}$$

[0067] Receiver 150 may perform the processing for equation (35) in two stages. In the first part of the first stage, receiver 150 may perform front-end filtering in each chip period k , as follows:

$$\underline{\mathbf{y}}(k) = \underline{\mathbf{F}}^H \underline{\mathbf{x}}(k) , \quad \text{Eq (36)}$$

where $\underline{\mathbf{y}}(k)$ is an $M \times 1$ vector of filtered samples.

[0068] In the second part of the first stage, receiver 150 may despread and descramble the filtered samples for each channelization code c , as follows:

$$\hat{\underline{\mathbf{d}}}_c(s) = \frac{1}{\sqrt{C}} \cdot \sum_{k=sC}^{(s+1)C-1} \underline{\mathbf{y}}(k) \cdot [v_c(k \bmod C) \cdot p(k)]^* , \quad \text{Eq (37)}$$

where $\hat{\underline{\mathbf{d}}}_c(s)$ an $M \times 1$ vector of filtered symbols, which is an estimate of $\underline{\mathbf{d}}_c(s)$.

[0069] In the second stage, receiver 150 may perform combining for each channelization code c , as follows:

$$\hat{\underline{\mathbf{b}}}_c(s) = \underline{\Delta}_c^H \hat{\underline{\mathbf{d}}}_c(s) . \quad \text{Eq (38)}$$

[0070] Receiver 150 may also recover the data symbols in $\underline{\mathbf{b}}_c(s)$ as follows:

$$\hat{\underline{\mathbf{b}}}_c(s) = \underline{\Delta}_c^H \underline{\mathbf{F}}^H \frac{1}{\sqrt{C}} \cdot \sum_{k=sC}^{(s+1)C-1} \underline{\mathbf{x}}(k) \cdot [v_c(k \bmod C) \cdot p(k)]^* . \quad \text{Eq (39)}$$

[0071] Equation (39) may also be performed in two stages. In the first part of the first stage, receiver 150 may despread and descramble the received samples for each channelization code c to obtain despread symbols for that channelization code. In the second part of the first stage, receiver 150 may perform filtering on the despread symbols for each channelization code c with the same front-end filter $\underline{\mathbf{F}}$ to obtain filtered symbols for that channelization code. In the second stage, receiver 150 may combine the filtered symbols for each channelization code c with the combiner matrix $\underline{\Delta}_c$ for that channelization code.

[0072] $\underline{\mathbf{W}}_p$ is obtained based on pilot symbols or chips and may not have converged to the optimal settings, e.g., because a finite number of pilot symbols (P) is used for training. In such instances, using $\underline{\mathbf{F}}$ and $\underline{\Delta}_c$ derived based on $\underline{\mathbf{W}}_p$ may result in performance degradation.

[0073] In another receiver design, $\underline{\mathbf{W}}_p$ is used as the front-end filter for the first stage. A combiner matrix $\underline{\mathbf{D}}_c$ is computed for each channelization code c and used for the second stage.

[0074] The filtered symbols obtained with $\underline{\mathbf{W}}_p$ may be expressed as:

$$\begin{aligned}\underline{\mathbf{z}}_c(s) &= \underline{\mathbf{W}}_p^H \underline{\chi}_c(s) \\ &= \underline{\mathbf{W}}_p^H \underline{\tilde{\mathbf{H}}} \underline{\mathbf{G}}_c \underline{\mathbf{B}}_c \underline{\mathbf{b}}_c(s) + \underline{\mathbf{n}}_c(s) \\ &= \underline{\mathbf{A}}_c \underline{\mathbf{b}}_c(s) + \underline{\mathbf{n}}_c(s)\end{aligned}\quad \text{Eq (40)}$$

where $\underline{\mathbf{A}}_c = \underline{\mathbf{W}}_p^H \underline{\tilde{\mathbf{H}}} \underline{\mathbf{G}}_c \underline{\mathbf{B}}_c$, and Eq (41)

$\underline{\mathbf{z}}_c(s)$ is an $M \times 1$ vector of filtered symbols for channelization code c .

[0075] The data symbols in $\underline{\mathbf{b}}_c(s)$ may be obtained as:

$$\hat{\underline{\mathbf{b}}}_c(s) = \underline{\mathbf{D}}_c^H \underline{\mathbf{z}}_c(s), \quad \text{Eq (42)}$$

where $\underline{\mathbf{D}}_c$ is an $M \times L$ combiner matrix for channelization code c .

[0076] Combiner matrix $\underline{\mathbf{D}}_c$ may be derived based on minimum mean square error (MMSE) criterion, as follows:

$$\underline{\mathbf{D}}_c = (\underline{\mathbf{A}}_c \underline{\mathbf{A}}_c^H + \underline{\mathbf{R}}_{\text{nn},c})^{-1} \underline{\mathbf{A}}_c. \quad \text{Eq (43)}$$

[0077] The noise covariance matrix $\underline{\mathbf{R}}_{\text{nn},c}$ may be estimated as follows:

$$\begin{aligned}\underline{\mathbf{R}}_{\text{nn},c} &= \frac{1}{P} \cdot \sum_{s=s_0}^{s_0+P-1} [\underline{\mathbf{z}}_p(s) - \underline{\mathbf{A}}_p(s) \underline{\mathbf{b}}_p(s)] [\underline{\mathbf{z}}_p(s) - \underline{\mathbf{A}}_p(s) \underline{\mathbf{b}}_p(s)]^H \\ &= \frac{1}{P} \cdot \sum_{s=s_0}^{s_0+P-1} \underline{\mathbf{z}}_p(s) \underline{\mathbf{z}}_p^H(s) - \underline{\mathbf{P}}_p \underline{\mathbf{P}}_p^H, \end{aligned}\quad \text{Eq (44)}$$

where $\underline{\mathbf{A}}_p = \underline{\mathbf{W}}_p^H \underline{\tilde{\mathbf{H}}} \underline{\mathbf{G}}_p$. The second equality in equation (44) results from the fact that an unbiased estimate of $\underline{\mathbf{A}}_p$ is simply $\underline{\mathbf{A}}_p = \underline{\mathbf{P}}_p$.

[0078] The $M \times L$ matrix $\underline{\mathbf{A}}_c$ may be expressed as:

$$\underline{\mathbf{A}}_c = \underline{\mathbf{A}}_p \underline{\mathbf{G}}_p^{-1} \underline{\mathbf{G}}_c \underline{\mathbf{B}}_c . \quad \text{Eq (45)}$$

[0079] As shown in equation (45), matrix $\underline{\mathbf{A}}_c$ may be computed for each channelization code c based on (a) matrix $\underline{\mathbf{P}}_p$ estimated from pilot symbols or chips and applicable for all channelization codes and (b) traffic-to-pilot ratio $\underline{\mathbf{G}}_p^{-1} \underline{\mathbf{G}}_c$ and transmit matrix $\underline{\mathbf{B}}_c$ that are specific for channelization code c . As shown in equation (43), combiner matrix $\underline{\mathbf{D}}_c$ may be computed for each channelization code c based on (a) noise covariance matrix $\underline{\mathbf{R}}_{nn,c}$ that is applicable for all channelization codes and (b) matrix $\underline{\mathbf{A}}_c$ computed for channelization code c .

[0080] Combiner matrix $\underline{\mathbf{D}}_c$ may also be estimated for each channelization code c as follows:

$$\underline{\mathbf{R}}_{zz} = E \left\{ \frac{1}{C} \cdot \sum_{c=1}^C \underline{\mathbf{z}}_c(s) \underline{\mathbf{z}}_c^H(s) \right\} , \quad \text{and} \quad \text{Eq (46)}$$

$$\underline{\mathbf{D}}_c = \underline{\mathbf{R}}_{zz}^{-1} \underline{\mathbf{A}}_c , \quad \text{Eq (47)}$$

where $\underline{\mathbf{R}}_{zz}$ is an $M \times M$ covariance matrix for $\underline{\mathbf{z}}_c(s)$.

[0081] In equation (46), an outer product $\underline{\mathbf{z}}_c(s) \underline{\mathbf{z}}_c^H(s)$ may be computed for each channelization code, then averaged over all channelization codes with the same transmit matrix $\underline{\mathbf{B}}_c$, and further averaged over a sufficient number of symbol periods to obtain covariance matrix $\underline{\mathbf{R}}_{zz}$. $\underline{\mathbf{D}}_c$ may then be computed based on $\underline{\mathbf{A}}_c$ and the inverse of $\underline{\mathbf{R}}_{zz}$.

[0082] Receiver 150 may recover the data symbols in $\underline{\mathbf{b}}_c(s)$ as follows:

$$\hat{\underline{\mathbf{b}}}_c(s) = \underline{\mathbf{D}}_c^H \left(\frac{1}{\sqrt{C}} \cdot \sum_{k=sC}^{(s+1)C-1} \underline{\mathbf{W}}_p^H \underline{\mathbf{x}}(k) \cdot [v_c(k \bmod C) \cdot p(k)]^* \right) . \quad \text{Eq (48)}$$

[0083] In equation (48), receiver 150 may filter the received samples with front-end filter $\underline{\mathbf{W}}_p$, then despread and descramble the filtered samples for each channelization code c , and then combine the filtered symbols for each channelization code with combiner matrix $\underline{\mathbf{D}}_c$.

[0084] Receiver 150 may also recover the data symbols in $\underline{\mathbf{b}}_c(s)$ as follows:

$$\hat{\underline{\mathbf{b}}}_c(s) = \underline{\mathbf{D}}_c^H \underline{\mathbf{W}}_p^H \frac{1}{\sqrt{C}} \cdot \sum_{k=sC}^{(s+1)C-1} \underline{\mathbf{x}}(k) \cdot [v_c(k \bmod C) \cdot p(k)]^* . \quad \text{Eq (49)}$$

[0085] In equation (49), receiver 150 may despread and descramble the received samples for each channelization code c , then filter the despread symbols for each channelization code with front-end filter $\underline{\mathbf{W}}_p$, and then combine the filtered symbols for each channelization code with combiner matrix $\underline{\mathbf{D}}_c$.

[0086] The receiver processing may also be performed in multiple stages in other manners. A specific receiver design is described below.

[0087] **FIG. 4** shows a block diagram of a design of various blocks within receiver 150 in FIG. 1. In this design, front-end filtering is performed prior to CDMA demodulation. Block 160 includes a front-end filter/equalizer 410 and a CDMA demodulator 420. Demodulator 420 includes C descramblers/despreaders 422a through 422C for up to C channelization codes used for traffic data. Block 170 includes C combiners 432a through 432C for up to C channelization codes used for traffic data.

[0088] Within channel processor 156, a timing estimator 442 determines the timing of the received signals. Unit 442 may estimate channel impulse responses and/or power delay profiles for different antennas and may determine the center of gravity of the channel impulse responses and/or power delay profiles. Unit 442 may then determine the timing of the received signals based on the center of gravity.

[0089] A coefficient computation unit 444 derives coefficients for filter $\underline{\mathbf{W}}_p$ based on the received samples, e.g., as shown in equation (28). Unit 444 may also derive $\underline{\mathbf{W}}_p$ based on RLS, block least squares, or some other technique. The timing information from unit 442 may be used for training, e.g., to align the locally generated pilot chips with the received samples. Unit 444 provides $\underline{\mathbf{W}}_p$ to front-end filter 410.

[0090] Filter 410 performs front-end filtering/equalization on the received samples $\underline{\mathbf{x}}(k)$ with $\underline{\mathbf{W}}_p$ and provides filtered samples. Within demodulator 420, each unit 422 despreads and descrambles the filtered samples for a different channelization code and provides filtered symbols $\underline{\mathbf{z}}_c(s)$ for that channelization code.

[0091] A unit 446 despreads and descrambles the filtered samples for pilot channelization code p . Units 422 and 446 perform despreading and descrambling based

on the timing provided by unit 442. A channel estimator 448 estimates an $M \times M$ matrix $\underline{\mathbf{W}}_p^H \tilde{\underline{\mathbf{H}}}$ based on the filtered pilot symbols from unit 446. A unit 450 computes the outer product $\underline{\mathbf{z}}_c(s) \underline{\mathbf{z}}_c^H(s)$ for each channelization code, averages the outer product across channelization codes and symbol periods, and provides correlation matrix $\underline{\mathbf{R}}_{zz}$, e.g., as shown in equation (46). A unit 452 derives the coefficients for combiner matrix $\underline{\mathbf{D}}_c$ for each channelization code c based on matrix $\underline{\mathbf{W}}_p^H \tilde{\underline{\mathbf{H}}}$ from unit 448, the correlation matrix $\underline{\mathbf{R}}_{zz}$ from unit 450, and code-specific matrices, as follows:

$$\underline{\mathbf{D}}_c = \underline{\mathbf{R}}_{zz}^{-1} \underline{\mathbf{W}}_p^H \tilde{\underline{\mathbf{H}}} \underline{\mathbf{G}}_c \underline{\mathbf{B}}_c . \quad \text{Eq (50)}$$

[0092] Within RX MIMO processor 170, each combiner 452 combines the filtered symbols for a different channelization code c based on the combiner matrix $\underline{\mathbf{D}}_c$ and provides detected symbols for that channelization code.

[0093] Receiver 150 may estimate the received signal quality, which may be quantified by a signal-to-interference-and-noise ratio (SINR), etc. The detected symbols from equation (42) may be expressed as:

$$\begin{aligned} \hat{\underline{\mathbf{b}}}_c(s) &= \underline{\mathbf{D}}_c^H \underline{\mathbf{A}}_c \underline{\mathbf{b}}_c(s) + \underline{\mathbf{D}}_c^H \underline{\mathbf{n}}_c(s) \\ &= \underline{\mathbf{L}}_c^H \underline{\mathbf{b}}_c(s) + \underline{\mathbf{w}}_c(s) \end{aligned} \quad \text{Eq (51)}$$

where $\underline{\mathbf{L}}_c^H = \underline{\mathbf{D}}_c^H \underline{\mathbf{A}}_c$ and $\underline{\mathbf{w}}_c(s) = \underline{\mathbf{D}}_c^H \underline{\mathbf{n}}_c(s)$. The covariance of $\underline{\mathbf{n}}_c(s)$ may be given as $\underline{\mathbf{R}}_{nn,c} = E \{ \underline{\mathbf{n}}_c(s) \underline{\mathbf{n}}_c^H(s) \}$. The covariance of $\underline{\mathbf{w}}_c(s)$ may be given $\underline{\mathbf{R}}_{ww,c} = \underline{\mathbf{D}}_c^H \underline{\mathbf{R}}_{nn,c} \underline{\mathbf{D}}_c$.

[0094] The SINR of the ℓ -th element of $\hat{\underline{\mathbf{b}}}_c(s)$, $b_{\ell,c}(s)$, may be expressed as:

$$\text{SINR} \{b_{\ell,c}(s)\} = \frac{|L_c(\ell, \ell)|^2}{R_{ww,c}(\ell, \ell) + \sum_{i=1, i \neq \ell}^L |L_c(\ell, i)|^2} , \quad \text{Eq (52)}$$

where $L_c(\ell, \ell)$ is the (ℓ, ℓ) -th element of $\underline{\mathbf{L}}_c$, and

$R_{ww,c}(\ell, \ell)$ is the (ℓ, ℓ) -th element of $\underline{\mathbf{R}}_{ww,c}$.

[0095] $\text{SINR} \{b_{\ell,c}(s)\}$ is the SINR of the ℓ -th data stream sent with channelization code c and may be used to select a data rate for that data stream. The SINR for each

channelization code c is dependent on transmit matrix $\underline{\mathbf{B}}_c$ used for that channelization code. Receiver 150 may determine the SINR for different possible transmit matrices and select the transmit matrix with the highest SINR. Receiver 150 may send feedback information to transmitter 110. This feedback information may comprise the transmit matrix selected for each channelization code, the SINR or data rate for each channelization code, an average SINR or data rate for all channelization codes, etc.

[0096] In general, front-end filtering may be performed in the first stage to process non on-time signal components in the M received signals. The front-end filter is, in general, not dependent on how the signals are processed at the transmitter prior to transmission. For a CDM transmission, the front-end filter may be applicable for all channelization codes. The second stage may combine the on-time signal components to recover the L transmitted signals. The combiner matrices used in the second stage may be dependent on how the signals are processed prior to transmission (e.g., the transmit matrix $\underline{\mathbf{B}}_c$ and gain matrix $\underline{\mathbf{G}}_c$ used by the transmitter) and other factors (e.g., the channel response $\tilde{\underline{\mathbf{H}}}$ and signal statistics $\underline{\mathbf{R}}_{zz}$).

[0097] FIG. 5 shows a design of a process 500 for recovering a MIMO transmission at a receiver. A front-end filter for processing (e.g., compensating for, suppressing, or mitigating) non on-time signal components in multiple received signals is derived (block 512). The front-end filter does not isolate the non on-time signal components. Instead, the front-end filter processes the non on-time signal components in a desirable/beneficial manner and may also (incidentally) process on-time signal components. At least one combiner matrix for combining on-time signal components for multiple transmitted signals is also derived (block 514). The front-end filter may be $\underline{\mathbf{F}}$ derived as shown in equation (32), and the combiner matrices may be $\underline{\Delta}_c$ derived as shown in equation (33). The front-end filter may also be $\underline{\mathbf{W}}_p$ derived as shown in equation (26) or (28), and the combiner matrices may be $\underline{\mathbf{D}}_c$ derived, e.g., as shown in equation (43), (47) or (50). The front-end filter and combiner matrices may also be derived in other manners. The front-end filter may be derived based on received data for pilot and in accordance with, e.g., least squares criterion. The combiner matrices may be derived based on the transmit matrices used to send data, the gains used for data, a channel response estimate, the front-end filter, signal and/or noise statistics, etc. The combiner matrices may also be derived in accordance with MMSE or some other criterion.

[0098] Received data is filtered to process the non on-time signal components in the multiple received signals (block 516). Filtered data is processed to combine the on-time signal components for the multiple transmitted signals (block 518). Received data for more than one symbol period may be filtered to process the non on-time signal components. Filtered data for one symbol period may be processed to combine the on-time signal components. The received data and filtered data may be given in samples, symbols, etc.

[0099] For a CDM transmission sent with multiple channelization codes, a single front-end filter may be derived and used to process the non on-time signal components, and multiple combiner matrices may be derived and used to combine the on-time signal components for the multiple channelization codes. In one scheme, the received data is first filtered with the front-end filter to obtain intermediate data. The intermediate data is then despread for each channelization code to obtain filtered data for the channelization code. The filtered data for each channelization code is further processed with a combiner matrix for that channelization code to obtain output data for the channelization code. In another scheme, the received data is first despread for each channelization code to obtain despread data for the channelization code. The despread data for each channelization code is then filtered with the same front-end filter to obtain filtered data for the channelization code. The filtered data for each channelization code is further processed with a combiner matrix for the channelization code to obtain output data for the channelization code.

[00100] For CDM, the front-end filter may be derived based on the received data and known pilot, e.g., based on (a) samples for the received data and known pilot chips for chip level training or (b) despread pilot symbols obtained from the received data and known pilot symbols for symbol level training. The combiner matrices may be derived based on the transmit matrices used for the multiple channelization codes, the gains for the multiple channelization codes, a channel response estimate, the front-end filter, signal and/or noise statistics, etc., or a combination thereof. The front-end filter and channel response estimate may be estimated jointly, e.g., as $\mathbf{W}_p^H \tilde{\mathbf{H}}$.

[00101] The multi-stage receiver may also be used for other communication systems. For example, in a time division multiplexed (TDM) system, a front-end filter may be derived based on pilot received in a first time interval, and a combiner matrix for a second time interval may be derived based on a transmit matrix used in the second time

interval. Data received in the second time interval may be filtered with the front-end filter, and the filtered data may be further processed with the combiner matrix.

[00102] In general, a filter may be derived based on pilot, which may be sent on a particular channelization code and/or time interval and using a particular transmit matrix and gain. The filter derived from the pilot may be used to derive a filter for data, which may be sent on other channelization codes and/or time intervals and possibly using different transmit matrices and gains.

[00103] For CDMA, the on-time and non on-time signal components may be distinguished by the time at which they are transmitted. The receiver may process a window of samples in order to recover a desired symbol transmitted by the transmitter. The timing of the equalizer determines the time instant at which the desired symbol is transmitted relative to the window. The samples obtained by the receiver contain different additive signal components including on-time and non on-time signal components. The on-time signal components are signal components for the desired symbol as well as other symbols transmitted at the same time as the desired symbol. All other signal components are non on-time signal components, which include signal components tracing back to symbols transmitted before and after the desired symbol.

[00104] A symbol may be transmitted by a transmit function, which may be dependent on one or more parameters. For example, the transmit function may be dependent on symbol period s , channelization code c , frequency slot or subcarrier index n , etc., and may be denoted as $f(s, c, n, \dots)$. For simplicity, the transmit function may be dependent on three parameters s , c and n , or a tuple (s, c, n) . The transmit functions for different symbols may be orthogonal so that $\langle f(s_1, c_1, n_1), f(s_2, c_2, n_2) \rangle \neq 0$ only if $s_1 = s_2$, $c_1 = c_2$ and $n_1 = n_2$, which may be expressed as $(s_1, c_1, n_1) = (s_2, c_2, n_2)$.

[00105] A received signal may include (a) desired signal components from a desired transmit function $f(s_1, c_1, n_1)$ defined by tuple (s_1, c_1, n_1) and (b) other signal components from other transmit functions $f(s, c, n)$, with $(s, c, n) \neq (s_1, c_1, n_1)$. The front-end filtering in the first stage would process the other signal components. The combiner in the second stage would process the desired signal components.

[00106] For CDM, the transmit functions for symbol period s are determined by channelization codes of length C multiplied by scrambling sequence $p(k)$. The transmit function for symbol period s and channelization code c may be denoted as $f(s, c)$.

From the perspective of a symbol transmitted with transmit function $f(s_1, c_1)$, the received signal contains the following:

1. non on-time signal components corresponding to $f(s, c)$ for $s \neq s_1$,
2. on-time signal components corresponding to $f(s_1, c)$ and composed of:
 - a. on-time signal components from the desired channelization code and corresponding to $f(s_1, c_1)$, and
 - b. on-time signal components from other channelization codes and corresponding to $f(s_1, c_2)$ for $c_1 \neq c_2$.

[00107] The front-end filter processes the non on-time signal components corresponding to $f(s, c)$. The descrambling and despreading by the front-end filter also cancels the on-time signal components from other channelization codes and corresponding to $f(s_1, c_2)$. The combiner processes the on-time signal components from the desired channelization code and corresponding to $f(s_1, c_1)$.

[00108] In a single carrier system that does not utilize CDM, the transmit functions may be simply digital deltas in time and may be given as $f(s) = \delta(t - s)$. As time t advances, the position of the delta changes in time.

[00109] In an OFDM-based system, the transmit functions may be for different subcarriers and may be given as $f(s, n)$, where n is a subcarrier index. The subcarriers in OFDM may correspond to the channelization codes in CDM. A transmitter may send N data/pilot symbols on N subcarriers in an OFDM symbol period from a given transmit antenna by (a) converting the N data/pilot symbols to the time-domain with an inverse fast Fourier transform (IFFT) to obtain N time-domain samples and (b) appending a cyclic prefix to the time-domain samples to obtain an OFDM symbol. A receiver may obtain received data/pilot symbols for a given receive antenna by (a) removing the cyclic prefix in the received samples and (b) converting N received samples to the frequency-domain with a fast Fourier transform (FFT) to obtain N received symbols for the N subcarriers. The received symbols may correspond to $\underline{z}_c(s)$ in equation (40), where subscript c is replaced with subcarrier index n . For OFDM, the on-time signal components may be signal components sent on a particular subcarrier from different transmit antennas. The non on-time signal components may be signal components sent on other subcarriers. The front-end filter may be implemented by the FFT and cyclic

prefix removal at the receiver. A combiner matrix $\underline{D}_c$ may be computed for each subcarrier and used to combine received symbols from all received antennas for that subcarrier.

[00110] Those of skill in the art would understand that information and signals may be represented using any of a variety of different technologies and techniques. For example, data, instructions, commands, information, signals, bits, symbols, and chips that may be referenced throughout the above description may be represented by voltages, currents, electromagnetic waves, magnetic fields or particles, optical fields or particles, or any combination thereof.

[00111] Those of skill would further appreciate that the various illustrative logical blocks, modules, circuits, and algorithm steps described in connection with the disclosure herein may be implemented as electronic hardware, computer software, or combinations of both. To clearly illustrate this interchangeability of hardware and software, various illustrative components, blocks, modules, circuits, and steps have been described above generally in terms of their functionality. Whether such functionality is implemented as hardware or software depends upon the particular application and design constraints imposed on the overall system. Skilled artisans may implement the described functionality in varying ways for each particular application, but such implementation decisions should not be interpreted as causing a departure from the scope of the present disclosure.

[00112] The various illustrative logical blocks, modules, and circuits described in connection with the disclosure herein may be implemented or performed with a general-purpose processor, a digital signal processor (DSP), an application specific integrated circuit (ASIC), a field programmable gate array (FPGA) or other programmable logic device, discrete gate or transistor logic, discrete hardware components, or any combination thereof designed to perform the functions described herein. A general-purpose processor may be a microprocessor, but in the alternative, the processor may be any conventional processor, controller, microcontroller, or state machine. A processor may also be implemented as a combination of computing devices, e.g., a combination of a DSP and a microprocessor, a plurality of microprocessors, one or more microprocessors in conjunction with a DSP core, or any other such configuration.

[00113] The steps of a method or algorithm described in connection with the disclosure herein may be embodied directly in hardware, in a software module executed

by a processor, or in a combination of the two. A software module may reside in RAM memory, flash memory, ROM memory, EPROM memory, EEPROM memory, registers, hard disk, a removable disk, a CD-ROM, or any other form of storage medium known in the art. An exemplary storage medium is coupled to the processor such that the processor can read information from, and write information to, the storage medium. In the alternative, the storage medium may be integral to the processor. The processor and the storage medium may reside in an ASIC. The ASIC may reside in a user terminal. In the alternative, the processor and the storage medium may reside as discrete components in a user terminal.

[00114] The previous description of the disclosure is provided to enable any person skilled in the art to make or use the disclosure. Various modifications to the disclosure will be readily apparent to those skilled in the art, and the generic principles defined herein may be applied to other variations without departing from the spirit or scope of the disclosure. Thus, the disclosure is not intended to be limited to the examples described herein but is to be accorded the widest scope consistent with the principles and novel features disclosed herein.

[00115] **WHAT IS CLAIMED IS:**

CLAIMS

1. An apparatus comprising:
at least one processor to filter received data to process non on-time signal components in multiple received signals and to obtain filtered data, and to process the filtered data to combine on-time signal components for multiple transmitted signals; and
a memory coupled to the at least one processor.
2. The apparatus of claim 1, wherein the at least one processor filters the received data for more than one symbol period to process the non on-time signal components, and processes the filtered data for one symbol period to combine the on-time signal components.
3. The apparatus of claim 1, wherein the at least one processor filters the received data in time domain.
4. The apparatus of claim 1, wherein the at least one processor derives a front-end filter for processing the non on-time signal components and derives at least one combiner matrix for combining the on-time signal components.
5. The apparatus of claim 4, wherein the at least one processor derives the front-end filter based on received data for pilot and derives the at least one combiner matrix based on at least one transmit matrix used to send data in the multiple transmitted signals.
6. The apparatus of claim 1, wherein the at least one processor derives a front-end filter for processing the non on-time signal components and derives multiple combiner matrices for combining the on-time signal components for multiple channelization codes used for the multiple transmitted signals.
7. The apparatus of claim 6, wherein the at least one processor filters the received data with the front-end filter and processes the filtered data with the multiple combiner matrices to obtain output data for the multiple channelization codes.

8. The apparatus of claim 6, wherein the at least one processor filters the received data with the front-end filter to obtain intermediate data, despreads the intermediate data for each of the multiple channelization codes to obtain filtered data for the channelization code, and processes the filtered data for each channelization code with a combiner matrix for the channelization code to obtain output data for the channelization code.

9. The apparatus of claim 6, wherein the at least one processor despreads the received data for each of the multiple channelization codes to obtain despread data for the channelization code, processes the despread data for each channelization code with the front-end filter to obtain filtered data for the channelization code, and processes the filtered data for each channelization code with a combiner matrix for the channelization code to obtain output data for the channelization code.

10. The apparatus of claim 6, wherein the at least one processor derives the front-end filter based on the received data and known pilot.

11. The apparatus of claim 6, wherein the at least one processor derives the front-end filter based on samples for the received data and known pilot chips.

12. The apparatus of claim 6, wherein the at least one processor despreads the received data with a pilot channelization code to obtain despread pilot symbols, and derives the front-end filter based on the despread pilot symbols and known pilot symbols.

13. The apparatus of claim 6, wherein the at least one processor derives the front-end filter based on least squares criterion.

14. The apparatus of claim 6, wherein the at least one processor derives the multiple combiner matrices based on multiple transmit matrices used for the multiple channelization codes.

15. The apparatus of claim 14, wherein the at least one processor derives the multiple combiner matrices further based on gains for the multiple channelization codes.

16. The apparatus of claim 14, wherein the at least one processor derives the multiple combiner matrices further based on a channel response estimate and the front-end filter.

17. The apparatus of claim 14, wherein the at least one processor derives the multiple combiner matrices based on minimum mean square error (MMSE) criterion.

18. The apparatus of claim 6, wherein the at least one processor derives a combiner matrix for each channelization code based on a noise covariance matrix, the front-end filter, a channel response estimate, and a transmit matrix for the channelization code.

19. The apparatus of claim 6, wherein the at least one processor derives a correlation matrix based on the filtered data and derives a combiner matrix for each channelization code based on the correlation matrix, the front-end filter, a channel estimate, and a transmit matrix for the channelization code.

20. The apparatus of claim 6, wherein the at least one processor updates the front-end filter at a first update rate and updates the multiple combiner matrices at a second update rate different from the first update rate.

21. The apparatus of claim 1, wherein the at least one processor derives a front-end filter based on pilot received in a first time interval, derives a combiner matrix for a second time interval based on a transmit matrix used in the second time interval, filters received data for the second time interval with the front-end filter to obtain filtered data for the second time interval, and processes the filtered data with the combiner matrix.

22. The apparatus of claim 1, wherein the at least one processor estimates received signal quality for at least one data signal sent in the multiple transmitted signals.

23. An apparatus comprising:

at least one processor to derive a front-end filter for processing non on-time signal components in multiple received signals, to derive multiple combiner matrices for combining on-time signal components for multiple transmitted signals sent with multiple channelization codes, to filter received samples with the front-end filter and obtain filtered symbols for the multiple channelization codes, and to process filtered symbols for each of the multiple channelization codes with a combiner matrix for the channelization code to obtain output symbols for the channelization code; and

a memory coupled to the at least one processor.

24. The apparatus of claim 23, wherein the at least one processor derives the front-end filter based on the received samples and known pilot chips.

25. The apparatus of claim 23, wherein the at least one processor derives a combiner matrix for each channelization code based on a transmit matrix used for the channelization code.

26. A method comprising:

filtering received data to process non on-time signal components in multiple received signals and obtain filtered data; and

processing the filtered data to combine on-time signal components for multiple transmitted signals.

27. The method of claim 26, further comprising:

deriving a front-end filter for processing the non on-time signal components; and

deriving at least one combiner matrix for combining the on-time signal components.

28. The method of claim 26, further comprising:

deriving a front-end filter for processing the non on-time signal components; and

deriving multiple combiner matrices for combining the on-time signal components for multiple channelization codes used for the multiple transmitted signals.

29. The method of claim 28, wherein the filtering the received data comprises filtering the received data with the front-end filter to obtain intermediate data,

and despreading the intermediate data for each of the multiple channelization codes to obtain filtered data for the channelization code, and

wherein the processing the filtered data comprises processing the filtered data for each channelization code with a combiner matrix for the channelization code to obtain output data for the channelization code.

30. The method of claim 28, wherein the deriving the front-end filter comprises deriving the front-end filter based on samples for the received data and known pilot chips.

31. The method of claim 28, wherein the deriving the multiple combiner matrices comprises deriving a combiner matrix for each of the multiple channelization codes based on a transmit matrix used for the channelization code.

32. An apparatus comprising:

means for filtering received data to process non on-time signal components in multiple received signals and obtain filtered data; and

means for processing the filtered data to combine on-time signal components for multiple transmitted signals.

33. The apparatus of claim 32, further comprising:

means for deriving a front-end filter for processing the non on-time signal components; and

means for deriving multiple combiner matrices for combining the on-time signal components for multiple channelization codes used for the multiple transmitted signals.

34. The apparatus of claim 33, wherein the means for filtering the received data comprises means for filtering the received data with the front-end filter to obtain intermediate data, and means for despreading the intermediate data for each of the multiple channelization codes to obtain filtered data for the channelization code, and

wherein the means for processing the filtered data comprises means for processing the filtered data for each channelization code with a combiner matrix for the channelization code to obtain output data for the channelization code.

35. The apparatus of claim 33, wherein the means for deriving the front-end filter comprises means for deriving the front-end filter based on samples for the received data and known pilot chips, and wherein the means for deriving the multiple combiner matrices comprises means for deriving a combiner matrix for each of the multiple channelization codes based on a transmit matrix used for the channelization code.

36. A processor readable media for storing instructions operable to:
filter received data to process non on-time signal components in multiple received signals and obtain filtered data; and
process the filtered data to combine on-time signal components for multiple transmitted signals.

37. An apparatus comprising:
at least one processor to perform processing for non on-time signal components in multiple received signals to obtain received symbols, to derive a combiner matrix for combining on-time signal components for multiple transmitted signals sent on a subcarrier, and to process received symbols for the subcarrier with the combiner matrix to obtain output symbols for the subcarrier; and
a memory coupled to the at least one processor.

38. The apparatus of claim 37, wherein the at least one processor performs processing for the non on-time signal components by removing cyclic prefix and performing a fast Fourier transform (FFT) on received samples to obtain the received symbols.

39. The apparatus of claim 37, wherein the at least one processor derives the combiner matrix based on at least one of a noise covariance matrix, a channel response estimate, and a transmit matrix for the subcarrier.

40. The apparatus of claim 37, wherein the at least one processor derives a second combiner matrix for combining on-time signal components for multiple transmitted signals sent on a second subcarrier, and processes received symbols for the

second subcarrier with the second combiner matrix to obtain output symbols for the second subcarrier.