

04-15-03

PTO/SB/16 (10-01)
Approved for use through 10/31/2002. OMB 0651-0032

U.S. Patent and Trademark Office; U.S. DEPARTMENT OF COMMERCE

Under the Paperwork Reduction Act of 1995, no persons are required to respond to a collection of information unless it displays a valid OMB control number.

PROVISIONAL APPLICATION FOR PATENT COVER SHEET

This is a request for filing a PROVISIONAL APPLICATION FOR PATENT under 37 CFR 1.53(c).

Express Mail Label No.

EV 195655905 US

#A/PRV

INVENTOR(S)					
Given Name (first and middle [if any])		Family Name or Surname		Residence (City and either State or Foreign Country)	
Thomas P.		McElwain		Merrimack, NH	
☐ Additional inventors are being named on the _____ separately numbered sheets attached hereto					
TITLE OF THE INVENTION (500 characters max)					
Joint Symbol, Amplitude, And Rate Estimator (JSAR)					
Direct all correspondence to: CORRESPONDENCE ADDRESS					
☒ Customer Number		22500		 22500 PATENT TRADEMARK OFFICE	
OR		Type Customer Number here			
☐ Firm or Individual Name					
Address					
Address					
City		State		ZIP	
Country		Telephone		Fax	
ENCLOSED APPLICATION PARTS (check all that apply)					
☒ Specification Number of Pages		26		☐ CD(s), Number	
☒ Drawing(s) Number of Sheets		5		☐ Other (specify)	
☐ Application Data Sheet. See 37 CFR 1.76					
METHOD OF PAYMENT OF FILING FEES FOR THIS PROVISIONAL APPLICATION FOR PATENT					
☐ Applicant claims small entity status. See 37 CFR 1.27.				FILING FEE AMOUNT (\$)	
☐ A check or money order is enclosed to cover the filing fees					
☒ The Commissioner is hereby authorized to charge filing fees or credit any overpayment to Deposit Account Number:				190130	
☐ Payment by credit card. Form PTO-2038 is attached.				\$160.00	
The invention was made by an agency of the United States Government or under a contract with an agency of the United States Government.					
☒ No.					
☐ Yes, the name of the U.S. Government agency and the Government contract number are: _____					

Respectfully submitted,

SIGNATURE

Daniel J. Long

TYPED or PRINTED NAME

Daniel J. Long

TELEPHONE

603-885-2643

Date

04/14/2003

REGISTRATION NO.

(if appropriate)

Docket Number:

29,404

20030076 PRO

USE ONLY FOR FILING A PROVISIONAL APPLICATION FOR PATENT

This collection of information is required by 37 CFR 1.51. The information is used by the public to file (and by the PTO to process) a provisional application. Confidentiality is governed by 35 U.S.C. 122 and 37 CFR 1.14. This collection is estimated to take 8 hours to complete, including gathering, preparing, and submitting the complete provisional application to the PTO. Time will vary depending upon the individual case. Any comments on the amount of time you require to complete this form and/or suggestions for reducing this burden, should be sent to the Chief Information Officer, U.S. Patent and Trademark Office, U.S. Department of Commerce, Washington, D.C. 20231. DO NOT SEND FEES OR COMPLETED FORMS TO THIS ADDRESS. SEND TO: Box Provisional Application, Assistant Commissioner for Patents, Washington, D.C. 20231.

EXPRESS MAIL CERTIFICATE

EXPRESS MAIL CERTIFICATE EV 195655905 US

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

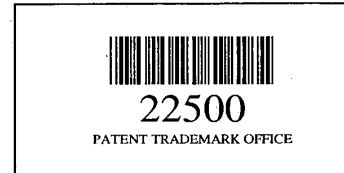
In re Application of: MCELWAIN, Thomas P.

Filed: Herein

Atty. Dkt. No: 20030076 PRO

For: JOINT SYMBOL, AMPLITUDE, AND RATE ESTIMATOR (JSAR)

To: Box Provisional Application & Fee
Commissioner for Patents
Washington, D.C. 20231



CERTIFICATE OF MAILING 37 CFR 1.10: I certify that this correspondence is being deposited on the below date with the U.S. Postal Service with sufficient postage as EXPRESS MAIL POST OFFICE TO ADDRESSEE addressed to: Box Provisional Application and Fee, Commissioner for Patents, Washington, DC 20231.

Date: 4-14-03

Gloria Abbasciano
Gloria Abbasciano

Dear Honorable Commissioner:

LETTER OF TRANSMITTAL

Submitted herewith is a Provisional Patent Application consisting of 1 pages of cover sheet, 26 pages of specification and claims, 5 sheets of drawings.

☐ Invention was made by an agency of the United States Government or under a contract with an agency of the United States Government.

DEPOSIT ACCOUNT **190130** AUTHORIZATION – The Office is hereby authorized to charge the required fee, and any deficiency or credit any overpayment in the fees to the above deposit account, owned by BAE SYSTEMS Information and Electronic Systems Integration Inc.

Respectfully submitted,

Daniel J. Long
Daniel J. Long, Reg. No. 29,404

USPTO Cus. No. 22500
BAE Systems
PO Box 868, NHQ01-719
Nashua, NH 03061-0868
Tel 603.885.2643
Fax 603.885.2167

JOINT SYMBOL, AMPLITUDE, AND RATE ESTIMATOR (JSAR)

Background of the Invention

1. Field of the Invention

The present invention relates to communications and more particularly to co-channel communications.

2. Brief Description of Prior Developments

A great number of communications and data transfer systems operate at or near a full capacity. If a means could be found to increase the number of available channels, by reassigning channels to be perhaps slightly interfering, the overall throughput could be increased without increasing bandwidth. Conventional receivers fail in the presence of this sort of interference (co-channel interference), in addition, many receivers require prior knowledge of signal parameters such as phase and amplitude of the channel. The problem solved by this invention is an efficient means of jointly estimating symbols, channel amplitude, and data rate transmitted in a super-saturated communications channel. The most noteworthy accomplishment of this invention is the ability to estimate the symbols and data rate, blindly without prior knowledge of channel amplitudes and phase.

The conventional Minimum Mean Squared Error (MMSE) Multiuser detector with prior information is described by Wang and Poor in the IEEE publication 'Iterative (Turbo) Soft Interference Cancellation and Decoding for Coded CDMA', in the Transactions on Communications, July 1999. See also Alexander, Reed, Asenstorfer, and Schlegel, "Iterative Multiuser Interference Reduction: Turbo CDMA," *IEEE Trans on Comm*, July 1999; and Poor, "Turbo Multiuser Detection: An Overview," *ISSSTA 2000*.

MMSE detectors with prior information usually show up in a turbo setting, but this is not required. In this publication, a complete solution to the problem is computed, and one approximate (low complexity) method is described. This prior art was not designed for overloaded systems, which are of particular interest to our application.

In the prior art, algebraic means are used to compute linear operators for the entire set of users (communications channels) simultaneously. This is done by utilizing prior information, or knowledge of the likely value of each user's bit of information, each at a particular instant in time.

The prior art, however, has a disadvantage in that it requires knowledge of all parameters.

The above referenced prior art is incorporated herein by reference.

The following prior art is also incorporated herein by reference.

S. Verdu, *Multiuser Detection*, Cambridge Press, 1998.

Chan and Wornell, "A Class of Asymptotically Optimum Iterated-Decision Multiuser Detectors", preprint.

Summary of Invention

The invention solves the previously mentioned problem of tightly integrating the following components: parameter estimation, symbol hypothesis testing, decoding, and rate identification. The approach assumes a module implemented prior to this module has identified the modulation scheme (e.g. CDMA). The solution presented here is a windowed-based Minimum Mean Squared Error (MMSE) approach to Turbo-decoding for joint signal demodulation, which is easily extended to time-varying channels. The

invention described here is based on an iterative decoding solution that exploits error correction codes. In addition, the approach defined here is suitable for symbol asynchronous as well as symbol synchronous. Also, this algorithm is suitable for a variety of signaling schemes such as M-ary Phase Shift Keying (MPSK).

Brief Description of the Drawings

The present invention is further described with regard to the accompanying drawings wherein:

Figure 1 is a schematic block diagram showing joint symbol amplitude and data rate estimator;

Figure 2 is a schematic block diagram showing initial amplitude estimation (Module 20 in Figure 1);

Figure 3 is a schematic block diagram showing joint amplitude estimator (Module 23 of Figure 1);

Figure 4 is a schematic block diagram showing parallelization of JSAR; and

Figure 5 is a schematic block diagram showing hardware conceptualization.

Detailed Description of the Preferred Embodiment

The apparatus in Figure 1 is now described in more detail. A data stream 10, potentially complex, is received from some source. For the case of Code Division Multiple Access (CDMA) communications schemes, the data stream is sampled by some multiple of the chip rate. For TDMA communication schemes, the data stream is

sampled at some multiple of the symbol rate. The received signal can effectively be modeled as the linear combination of many co-channel signals, mathematically illustrated by

$$r[n] = \sum_{i=1}^F \sum_{k=1}^K b_k[i] a_k s_k(nT_n - iT_i) + n_w(nT_n). \quad \text{Equation 1}$$

The term $b_k[i]$ represents the symbol for user k at time index i . The term $s_k(nT_n - iT_i)$ represents the user's signal characteristic for sample index n for the symbol period i for the user k . The characteristic waveform, $s_k(\cdot)$, is normalized such that the amplitude of the transmitted signal for user k is represented by a_k . The symbol period is defined by T_i and the sample period is represented by T_n . The linear model represents a summation of K separate users and over F symbols. The received signal can represent a collection from multiple sources. For example, the collection may consists of multiple antennas, not necessarily co-located, with potentially different polarizations.

The user's signal characteristic, $s_k(nT_n - iT_i)$, combines a sequence of signal transformations: pulse shaping filter, signal delays, receiver filters, code channels (for CDMA) and multipath. Note, that the characteristic waveform is not restricted to a sample or symbol interval.

The multiple access interference model in Equation 1 is represented concisely using matrix notation in Equation 2. The KF symbols corresponding to the KF simultaneous users is represented by the KF -element vector, $\mathbf{b}[i]$ at time i . A linear model is used to represent the received signal at time index i by

$$\mathbf{r}[i] = \mathbf{S}\mathbf{A}\mathbf{b}[i] + \mathbf{n}_w[i], \quad \text{Equation 2}$$

where $\mathbf{b}[i]$ represents the KF symbols at time index i . The term $\mathbf{S}$ is a $N \times KF$ matrix representing the combination of the spreading code, channel codes, pulse shaping filter, and propagation effects

$$\mathbf{S} = \begin{bmatrix} s_1[1,i] & s_2[1,i] & \cdots & s_{KF}[1,i] \\ s_1[2,i] & s_2[2,i] & \cdots & s_{KF}[2,i] \\ \vdots & \vdots & \ddots & \vdots \\ s_1[N,i] & s_2[N,i] & \cdots & s_{KF}[N,i] \end{bmatrix} \quad \text{Equation 3}$$

The matrix entries, $s_k[n,i]$, represents the n^{th} sample of the signal characteristic waveform for user k at time i . The term $\mathbf{A}$ is a $KF \times KF$ diagonal matrix that represents the complex signal amplitudes, which can include phase and polarization weighting. The term $\mathbf{n}_w[i]$ is a $N \times 1$ vector that represents additive white Gaussian noise (AWGN).

Returning to Figure 1, data 10 represents a vector of data, representing the received signal. This data 10 is passed over line 11 to the symbol hypothesis testing module (21) labeled in Figure 1 as Minimum Mean Squared Error (MMSE) based symbol estimator. As indicated by the name, the preferred approach because of computational efficiency is an MMSE implementation for symbol hypothesis testing (21). This computational efficiency will become more clear as we discuss the details of the MMSE-based joint amplitude estimator and the parallelization of this algorithm. The symbol hypothesis testing conducted in 21 is performed following completion of the initial amplitude estimate.

Note, any hypothesis testing approach, such as the maximum likelihood detector, is suitable for module 21. The maximum likelihood detector is a brute force approach which requires an exhaustive search, is suitable for 21. The solution consists of exhaustively evaluating the Euclidean distance between the received samples and the linear model of the samples using every possible hypothesis of the bit sequence. Instead of the optimal detector applied in module 21, a more computationally efficient form of symbol hypothesis testing may be applied, such as the M-algorithm, T-algorithm, etc. However, the most efficient approach is an MMSE-based symbol estimator which will be made more clear when discussing the MMSE-based amplitude estimator. Note, module 21 also includes an efficient implementation, referred to as the Q-algorithm. As is disclosed in U.S. Patent application Serial No. 10/105,918, filed March 25, 2002 entitled "System for Decreasing Processing Time in an Iterative Multi-User Detector System on the Q-algorithm the contents of which are incorporated by reference, the MMSE-symbol estimator cancels interference based on symbol estimates obtained from thresholding the symbol likelihoods.

The same vector 10 is passed over line 12 to the initial amplitude estimation module 20. As indicated by the name, the purpose of 20 is to provide an initial estimates of the amplitude for each interfering signal. The approach implemented in 20 consists of incoherently integrating the energy out of a properly normalized matched filter for all users determined to be active. Note these users can be transmitted over code, temporal, and/or frequency channels and this estimation procedure is appropriate for these modes of digital communication. Part of the function performed in 20, is the identification of which

users are active. User identification can be easily obtained via an energy detector based on some a sufficiently long observation interval.

The initial amplitudes are passed to the MMSE-based symbol estimator, 21, over line 24 and the MMSE-based Joint Amplitude Estimator, 23 over line 13. The MMSE-based symbol estimator uses the initial amplitude estimate only on the first iteration, for all other iterations it uses the amplitude estimates from 23 that are received over line 19. Recall this module may implement any type of symbol hypothesis testing procedure, however, an MMSE-based approach is preferred because of computational efficiency.

The MMSE-based symbol estimator provides symbols estimates to a bank of single user decoders in 22 over line 14. The symbols estimates out of 21 are routed to the decoders based on user number. For example, all symbol likelihoods corresponding to the first user are passed to the decoder dedicated to the first user. This mapping is repeated for all users. The single user decoders consists of a bank of K decoders that may consist of Soft-output Viterbi or Maximum A Posteriori based techniques such as BCJR. Soft-input, soft-output decoders are preferred for this implementation but not required. Inherent in the bank of symbol decoders is a de-interleaving, rate identification, de-puncturing and de-scrambling. Symbol estimates and their likelihoods are passed out of the bank of decoders for one of the most-likely data rates. In addition, symbol likelihood estimates are fed back around to both the MMSE-based symbol estimator (21) and MMSE-based amplitude estimator (23) over lines 17 and 18, respectively.

The new symbol likelihoods are used to update previous amplitude estimates in 23. The procedure for implementing the MMSE-based amplitude estimator will be

discussed shortly. The new amplitude estimates are passed to 21 over line 19. Module 21 then re-estimates the symbols based on prior information on the symbols and the updated parameter estimate. These updated symbol estimates are then routed to the bank of single-user decoders. This process is repeated for either some fixed number of iterations or until the estimates of the bit error rate or frame error rate achieve some desired level.

The apparatus in Figure 2 describes the procedure for calculating initial amplitude estimates in module 20 of Figure 1. The initial amplitude estimate obtained in module 20 of Figure 1 is based on the incoherent integration of the matched filter output over some observation interval. This procedure consists of applying a bank of matched filters for all users K and all symbols F to the data vector received over line 10 of Figure 1 and line 50 of Figure 2. The $KF \times 1$ vector representing the matched filter outputs is represented by

$$\mathbf{y} = \mathbf{S}^H \mathbf{r} = \mathbf{S}^H \mathbf{S} \mathbf{a} \mathbf{b} + \mathbf{S}^H \mathbf{n}_w. \quad \text{Equation 4}$$

The estimate of the amplitude is based on summing the squared outputs of the matched filter for each user, which is represented by

$$\hat{a}_k^2 = \frac{1}{F} \sum_{i=1}^F |y_{i,k}|^2 \quad \text{Equation 5}$$

The summation over F is performed for all K users to obtain K initial amplitude estimates for the K users. Since the estimate is based on incoherent integration, the estimate is biased which is illustrated by the expected value of the initial amplitude estimate for an arbitrary users k

$$E[\hat{a}_k^2] = a_k^2 + \frac{1}{F} \sum_{i=1}^F \sum_{\substack{j=1 \\ j \neq k}}^K \rho_{j,k}^2(i) a_j^2 \quad \text{Equation 6}$$

where the bias is represented by the last term in Equation 6 and $\rho_{j,k}^2(i) = (\mathbf{s}_k^H(i) \mathbf{s}_j(i))^2$.

An inactive user is defined as a signature signal that is not present. The expected value of the amplitude estimate for an arbitrary inactive channel is defined as

$$E[\hat{a}_{inactive}^2] = \frac{1}{F} \sum_{i=1}^F \sum_{j=1}^K \rho_{j,inactive}^2(i) a_j^2 \quad \text{Equation 7}$$

where $\rho_{j,inactive}^2(i) = (\mathbf{s}_{inactive}^H(i) \mathbf{s}_j(i))^2$. Since no signal is present for inactive users, a nonzero value for the expected value of the estimate for amplitude squared for inactive signature vectors is due to the presence of other active channels. Comparing Equations 6 and 7 suggests that examination of the estimate of squared amplitudes for non-active users can be used to estimate the bias relative to active users. The concept based on Equations 5 through 7 is illustrated in Figure 2, titled Initial Amplitude Estimation.

A data stream 50 representing the received signal, potentially complex, is received from some source. A bank of filters, represented by 51, 56, 61, 66, and 71 is applied to the data stream. The first set of filters (51, 56 and 56) consists of filters matched to users that have been determined to be active based on an energy detector. Filter 51 corresponds to a filter matched to the signature signal for user 1. Similarly, filter 56 corresponds to a filter matched to the signature signal for user 2. Note, the size of the bank of filter is based on the number of users determined to be active. This number is represented by the term K . A squaring operation (52, 57, and 62) is applied to the output

of each matched filter. These outputs are then passed to a bank of summers (53, 58, and 63) which are used to implement Equation 5 and accumulate energy in each of their respective processing chains, note the proper scaling required for averaging is included in the output of the summers. As previously mentioned these estimates of squared amplitude are biased, therefore, the bias removal component will now be discussed.

A second set of filters (66 and 71) consists of filters representing other channels that are were determined to be inactive. A bank of filters matched to these inactive user's signature signals is applied to the data stream 50. These channels have been determined to be inactive based on an energy detector. Note, the size of the bank of filter is based on an arbitrary number of users determined to be inactive.

Following the application of the inactive filters, amplitude squared estimates are divided into groups for averaging, where these groups can be based on various types of criteria. For example, in the case of CDMA, the estimate of the bias for all code channels corresponding to a sector is based on an average of the amplitude estimates for the non-active channels in that same sector. Therefore, all outputs corresponding to inactive users in the same sector are grouped together for averaging. This procedure is illustrated in Figure 2, by the averaging of the amplitude squared estimates in module 69. This combines all the estimates corresponding to the inactive users 1 through J . Using this type of approach for a CDMA cellular system, each sector will have a different estimate of the amplitude bias and all channels corresponding to the same sector use the same estimate of amplitude bias. For another multiple access system, the criteria for grouping prior to bias averaging may differ.

Following the computation of the amplitude bias in module 69, the bias is passed to modules 54, 59, and 64 for bias removal. Note, that source of the bias is a function of the grouping criteria. For example, for the case of CDMA, only the bias from the same sectors are used for bias removal. The illustration in Figure 2 shows the output 70 going to all bias removal modules (54, 59, 64) which presumes they obey the grouping criteria. The initial amplitude squared estimate obtained for the active channels has an estimate of the bias removed in modules 54, 59, and 64. Following the bias removal, the initial amplitude squared estimates are passed out to the remaining modules in Figure 1, over lines 55, 60, and 65 in Figure 2.

The initial amplitude squared estimates in module 20 of Figure 1 are updated in module 23 of Figure 1. The apparatus for updating these amplitude squared estimates is described in Figure 3. The solution described in Figure 3 is based on the assumption that the amplitude is nearly constant over some observation window. Since this solution is based on the MMSE criteria, this approach is easily extended to time-varying amplitudes, however, only the windowed based estimator will be discussed here.

A stream of data representing the received signal on line 10 of Figure 1 is also passed to the MMSE-based Joint Amplitude Estimator in module 23 over line 25. This data stream is divided up into M observation intervals as illustrated by r_1 , r_2 , through r_M on lines 101, 102 and 103 respectively, in Figure 3. These intervals can be based on time, code, or frequency. For the example presented here, let the observation interval refer to time. These observation intervals each enter a different processing module, illustrated by modules 111, 112, and 113. Note, that module 113 refers to the M^{th}

observation interval. In each processing module, an MMSE filter is computed which is defined for an arbitrary observation interval m as

$$G_m = \mathbf{B}_m \mathbf{S}_m^H (\mathbf{S}_m \chi_m \mathbf{S}_m^H + \sigma_n^2 \mathbf{I})^{-1}. \quad \text{Equation 8}$$

The term $\mathbf{B}_m$ refers to a diagonal matrix representing the expected bit or symbol value in interval m for all K users. For example,

$$\mathbf{B}_m = \begin{bmatrix} \hat{b}_1 & 0 & \cdots & 0 \\ 0 & \hat{b}_2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \hat{b}_K \end{bmatrix}, \quad \text{Equation 9}$$

where $\hat{b}_k$ represents the expected value of the symbol for user k . For illustrative purposes, for BPSK systems, $\hat{b}_k = 2p_k - 1$, where p_k equals the probability that the symbol equals 1. The term $\mathbf{S}_m$, is similar to that presented in Equation 2, except here $\mathbf{S}_m$ represents the “S-matrix” defined over a single observation interval, while $\mathbf{S}$ in Equation 2 is defined over F observation intervals. Therefore, $\mathbf{S}$ in Equation 2 contains the $\mathbf{S}_m$ defined here. The term χ_m represents the amplitude uncertainty and is rewritten as

$$\chi_m = (\Omega + \mathbf{A}^2) \Lambda_m \quad \text{Equation 10}$$

which more clearly defined the amplitude uncertainty in terms of the bit uncertainty, Λ_m , the initial amplitude uncertainty, Ω , and the expected value of the amplitude $\mathbf{A}$. The terms, Λ_m , Ω , and $\mathbf{A}$ are all diagonal matrices each element along the diagonal

corresponding to a different active user, therefore, χ_m is a diagonal matrix. This formulation assumes the amplitudes and corresponding uncertainties are constant for the M observation intervals.

Following the computation of M different MMSE filters in modules 111, 112, and 113, the MMSE filter is applied to components of the data stream corresponding to the same observation interval. However, prior to application of the MMSE filter, interference cancellation is performed based on the expected value of the bits, represented by $\mathbf{B}_m$ and prior estimates of the amplitude for all K users, represented by the $K \times 1$ vector $\underline{a}^{l-1}$. The interference cancellation is represented by

$$\tilde{r}_m = r_m - \mathbf{S}_m \mathbf{B}_m \underline{a}^{l-1}. \quad \text{Equation 11}$$

Therefore, application of the MMSE filter following interference cancellation is represented by

$$\Delta \underline{a}_m = G_m \left(r_m - \mathbf{S}_m \mathbf{B}_m \underline{a}^{l-1} \right) \quad \text{Equation 12.}$$

The amplitude estimator is defined in module 114 of Figure 3. It is an iterative formulation that updates previous estimates, represented by $\underline{a}^{l-1}$, with information based on new estimates of the users bits and data rate. These new estimates are represented by $\underline{a}^l$. The MMSE-based joint amplitude estimator in module 114 combines the individual estimates, $\Delta \underline{a}_m$, in a weighted manner. The weighted combination is expressed by

$$\underline{a}^l = \underline{a}^{l-1} + \left(\Omega^{1/2} (Q + I)^{-1} \Omega^{1/2} \right) \sum_{m=1}^M \Delta \underline{a}_m \quad \text{Equation 13}$$

where

$$Q = \sum_{m=1}^M Q_m \quad \text{Equation 14}$$

As indicated by Equation 13 and module 114, the individual estimates from modules 111-113 are passed to module 114 over lines 107-109. These individual estimates, $\Delta \underline{a}_m$, are $K \times 1$ vectors which are then summed together in module 114. These estimates are then weighted by $\left(\Omega^{1/2} (Q + I)^{-1} \Omega^{1/2} \right)$ where $\Omega^{1/2}$ is a diagonal matrix containing the standard deviation of the amplitude estimates and Q is represented in Equation 14. The term Q is a summation of quadratics computed in modules 111-113. Each quadratic is computed by

$$Q_m = G_m S_m B_m \quad \text{Equation 15}$$

which is shown to be based on the MMSE filter for the respective observation region defined in interval m . Therefore each module responsible for an observation interval must compute: the MMSE filter (see Equation 8); apply interference cancellation to the section of the data stream corresponding to the observation interval (see Equation 11); apply the MMSE filter to the interference mitigated data vector (see Equation 12); and computed the quadratic (see Equation 15).

The outputs of each module (111-113) is a $K \times 1$ vector $\Delta \underline{a}_m$ and a hermitian matrix $\underline{Q}_m$ which is a $K \times K$ matrix with $K(K+1)/2$ unique elements. These two outputs are passed to the Amplitude Estimator in module 114 over lines 107-109. The prior amplitude estimates, $\underline{a}^{l-1}$ are updated to $\underline{a}^l$ and passed out of module 114 over line 110 to the individual modules defined for the observation intervals (see 104-106). This feedback is part of the iterative implementation inherent in this MMSE formulation. However, the amplitude estimates are not updated until the bits (or symbols) have been updated in module 22 of Figure 1. Following the implementation of 21 and 22 in Figure 1, the symbols are fed back to the MMSE-based joint amplitude estimator in module 23 of Figure 1 where prior amplitude estimates will be updated as described in Figure 3. As previously mentioned the number of iterations is dictated by the user which is defined for either a fixed iteration number or until the frame error rate or estimate bit error rate metric reaches some desired level.

The amplitude estimate also exits module 114 over line 115 to the MMSE-based symbol estimator. Since for this particular implementation, the amplitudes are constant over the observation interval described for M smaller observation intervals, the same amplitude estimate is reported to each MMSE-based symbol estimator in module 21 of Figure 1. This approach is easily modified for a variable amplitude estimate which varies over the entire observation interval.

The apparatus in Figure 4 provides a more detailed characterization of Figure 1. Figure 4 shows an implementation that parallelizes the joint estimation based on M observation intervals. The data stream in 10 of Figure 1 is partitioned into

M observation intervals which are seen in Figure 4 as the data vectors arriving on lines 201-204. As in previous diagrams, the number of observation intervals is generalized to M intervals. These data vectors are processed immediately by a bank of MMSE filters which are implemented on M processors, 209-212, for all M observation intervals. Each processor simultaneously estimates symbols for all K users. Note, more than one processor can exist per node. A node is a generic processing component and therefore may be considered as a microprocessor chip or processing board. In addition, these data vectors are read into memory local to the processors, 205-208 because they are used repeatedly for the multiple iterations in this iterative approach to symbol, amplitude, and rate estimation.

The output of each MMSE processor $\underline{b}_m$ is a vector of K symbol likelihoods or symbol bits for observation interval m . Symbol bits are provided on the output if thresholding is performed within the MMSE processor. The benefit of supplying likelihoods to the decoders is the improved performance when using soft-decoders. The outputs, 217-220, are intended to represent $M \times K \times 1$ vectors passed to the system area network (SAN) which will then pass $K \times M \times 1$ frames of likelihoods (or symbols) to a bank of decoders. Note the same nodes used for implementing the bank of MMSE filters for symbol estimation will now be used to implement the K decoders, 226-229. These decoders are partitioned between the compute nodes.

Symbol likelihoods are routed over the SAN based on the assignment of decoders to processing components. The SAN routes the $M \times K \times 1$ vectors of symbol likelihoods from the processing nodes, previously used to perform MMSE-based symbol estimation,

to the same set of processing nodes which will now perform the bank of decoders. The SAN passes K frames of $M \times 1$ symbol estimates to the K decoders over lines 222-225. The frames of symbols are represented by $\underline{f}_k$ for the K users.

The number of decoders corresponds to the number of users, K . The bank of decoders may implement hard-in, soft-out or soft-in, soft-out decoders. In addition to decoding, this bank of decoders must perform descrambling, de-interleaving, de-puncturing, and rate identification. The output of each decoder, $\underline{f}_k$, is $M \times 1$ vector representing and update of M likelihoods in the frame following the implementation of the decoders. The K frames of $M \times 1$ likelihoods, $\underline{f}_k$, are presented on lines 230-233. In addition, the decoders supply likelihoods of the bits values on lines 234-237 for the same collection of symbols, the number of bits per user is dependent on the data rate. These bit likelihoods are available at any time in the iterative scheme for post-processing such as a cyclic reduction check, which is used to determine frame errors.

The likelihoods of the symbols are fed-back around to the bank of MMSE filter, however, now the MMSE filter are used for amplitude estimation. The K $M \times 1$ vectors of likelihoods, $\underline{f}_k$, are routed using the SAN to the same bank of processing nodes which will now perform the another bank of MMSE filters. The K $M \times 1$ vectors of likelihoods, $\underline{f}_k$, on lines 238-241 prior to the SAN are sent to the the MMSE filters as M $K \times 1$ vectors of likelihoods, $\underline{b}_m$, over lines 243-246.

The same bank of processors used to perform the MMSE filters in modules 213-216 will now be used for MMSE filter implementation for amplitude estimation,

illustrated by modules 251-254. The MMSE modules in amplitude updating will focus on the same observation intervals analyzed during symbol estimation. The MMSE processors at this stage will be implementing the functions in modules 111-113 in Figure 3. This bank of MMSE filters draw from the local memory (247-250) to retrieve prior amplitude estimates, updated symbol estimates, S-matrices, and the received data vector corresponding to the observation interval m .

Recall from Figure 3, each amplitude MMSE computes an amplitude update vector, $\Delta \underline{a}_m$, which are $K \times 1$ vectors. Each amplitude update vector must be passed to a central node which will implement the functions in module 114 in Figure 3. Therefore, all individual update vectors are passed to a central processing node via the SAN over lines 255-258. Recall, that the amplitude estimate is updated by weighting the sum of all amplitude updating vectors. The weighting is based on quadratic terms reflecting the uncertainty in the update, q_m . These quadratic terms are also passed to the central node to provide the necessary weighting.

The SAN passes all $\Delta \underline{a}_m$ vectors (M vectors of size $K \times 1$) and Q_m matrices, (M hermitian $K \times K$ matrices with $K(K+1)/2$ unique elements) over lines 260-263 to a central processing node, 264. Module 264 computes the functions in 114 of Figure 3 and the updated estimate gets distributed to all M processors used for the next MMSE-based symbol estimation. Therefore, module 264 passes the updated amplitude estimate to the SAN over line 231 and the SAN fans out this estimate for all K users to the M processors over lines 266-269. The updated amplitude estimate refreshes the amplitude estimates stored in the local memory to all MMSE processors. Following the receipt of

the updated amplitude estimates, the next iteration of the MMSE-based symbol estimator is implemented. Therefore, modules 209-212 repeat the symbol estimation on the next iteration using the new amplitude estimates. As mentioned in the discussions describing Figure 1, Figure 4 is implemented until estimates of the bit error rate metric achieves a certain level or until a fixed number of iterations has been executed.

The configuration presented in Figure 4 is an attractive implementation because bandwidth requirements are minimized by passing a total of only $K \cdot M$ floating point numbers over the SAN. The exception to this is during amplitude updating when the matrices Q_m are passed to a central node. Bandwidth can be minimized here by approximating Q_m as a diagonal and therefore pass a total of $2K \cdot M$ floating point numbers during amplitude updating. Since the MMSE operations, both symbol and amplitude, occur at the same processing nodes, less data is passed along the SAN and efficient techniques can be used to exploit previous MMSE filter weights.

The communications configuration described in Figure 4, is more easily recognized in Figure 5. Figure 5 shows L nodes responsible for all processing. These L nodes pass data between themselves using a high speed interconnect for the system area network (SAN). The modules 311-315 correspond to the time when the L nodes are used to implement the bank of M MMSE filters for symbol estimation in M observation intervals. In the second column, the same L nodes are used to implement the SISO decoders, which is illustrated by modules 321-325 in Figure 5. This stage of the algorithm corresponds to modules 226-229 in Figure 4. The decoders for K users are distributed among the L nodes.

The communication between nodes from the MMSE stage to the nodes for the decoding stage of the algorithm is illustrated by example. To simplify this discussion, let the number of nodes equal the number of observation intervals, $L = M$. This simplification is not an architecture restriction, but simply a convenient way to illustrate the communication between nodes. Based on this simplification, the output of each node in the first column is a $K \times 1$ vector representing symbol estimates for the K users. Each node in the second column is responsible for decoding a subset of the users. To further simplify the discussion, let each node in column 2 execute 2 decoders. Therefore, based on these simplifications, for purposes of explanation only, the symbols corresponding to users 1 and 2 from the first observation interval, module 301, remain at Node 1 for the decoding users 1 and 2 (see line 306). The symbols for the same 2 users in the second observation interval, 302, are passed over the high-speed interconnect to Node 1 in the second column, 321 over line 307. Similarly, the symbols for the same two users in the third observation interval are also passed to 321 over line 308. This procedure continues for the M observation intervals for the symbols corresponding to the first 2 users (see lines 309–310).

Node 4 in the second column is responsible for decoding users 7 and 8, therefore, 324 will receive 2 symbols corresponding to users 7 and 8 from all of the L observation nodes (see 301-305). For example, Node 1 passes the symbol estimates of users 7 and 8 to 324 over line 311. Similarly, Node 2, passes the symbol estimates for users 7 and 8 for the 2nd observation interval to 324 over line 312. This process is repeated for Node 3, see line 313. For Node 4, the symbol estimates of users 7 and 8 corresponding to the 4th observation interval remain at the 4th node for the MMSE implementation, see line 314.

This procedure continues for all L nodes in the second column, modules 321-325.

Therefore, symbol estimates out of the MMSE symbol estimators are routed from the M observation intervals to the K decoders, based on the mapping of the K decoders to the L available processing nodes.

The third column represents the L nodes, implementing the MMSE stage of the amplitude estimator, see modules 251-254 in Figure 4. Prior to implementing the amplitude estimator, the nodes 341-345 must receive all symbol likelihoods following the execution of the single user decoders in modules 321-325. The likelihoods of the symbols out of the K decoders are routed back to the M MMSE filters for the amplitude estimation stage of the algorithm described in Figure 4. For this example, 2 decoders are implemented per node in the 2nd column (321-325) and 1 MMSE is executed per node in the 3rd column (341-345). The output of Node 1 in the 2nd column is a $M \times 2$ array of likelihoods corresponding to the M symbol likelihoods for the first two users after implementing the decoder, descrambler, de-interleaver, and rate identification. Node 1 in the 3rd column retains the symbol likelihoods for the first two users in the first interval, illustrated by line 326. Node 2 in the 3rd column is responsible for the MMSE implementation over the second observation interval. Therefore, the second set of bits in the $M \times 2$ array is passed from Node 1 in 2nd column, 321, to Node 2 in the 3rd column (see line 327). This is repeated for all M observation intervals (lines 326-330). Therefore, the outputs of the decoder are passed to all nodes computing the amplitude updating vector in the M observation intervals.

The 4th column is responsible for updating the amplitude estimate by implementing the functions in 114 of Figure 3 and 264 of Figure 4. This is shown by

performing this operation at a single node, Node 3 of Figure 5, module 353. Note, the other nodes at this stage of the algorithm remain idle while the amplitude estimation is completed. Therefore the high-speed interconnect passes the amplitude updating vector and quadratics from the L nodes in the 3rd column to a central node over lines 346-350. The functions in module 114 of Figure 3 are implemented and the amplitude estimate is “fanned-out” to all of the L processing nodes for the next round of symbol estimation occurring at the next iteration. Therefore, the new amplitude estimate for all K users is distributed to all the L nodes over lines 356-360. The L nodes in the 5th column recompute the symbol estimates, using the new amplitude estimates, for the K users in the M observation intervals in modules 361-365. The stages and data routing over the high-speed interconnect are repeated until the maximum number of iterations is achieved or the desired bit error rate is reached.

It will be appreciated that this new implementation:

- a.) Solves the problem of the unknown channel parameters in the Wang / Poor detector
- b.) Can work with many varieties of multiuser detectors, currently in the public domain, or emerging
- c.) Dramatically reduces the number of computations needed in a blind approach (no pilot tones or training sequences are required) to joint signal demodulation, so that reliable operation can be achieved in a real-time implementation.

It will also be appreciated that:

- a.) Since this technique does not require knowledge of training sequences or existence of a pilot signal, it is considered a blind approach to symbol and data rate estimation
- b.) Efficient implementation of blind estimation because of the high degree of parallelism
- c.) Efficient implementation because of reduced number of operations associated with updating MMSE filter tap weights.
- d.) Efficient implementation because the number of iterations required to achieve a certain bit-error-rate for case of unknown amplitudes is typically similar to the case when the amplitudes are completely known. This efficiency occurs because the degradation in bit-error-rate for the case of unknown amplitudes is manageable relative to the case of known amplitudes.
- e.) This approach represents a new method to estimate channel amplitudes, extensions to time-varying channels is obvious.
- f.) This approach represents a new method to reduce biases in amplitude estimates based on incoherent integration.

While the present invention has been described in connection with the preferred embodiments of the various figures, it is to be understood that other similar embodiments may be used or modifications and additions may be made to the described embodiment for performing the same function of the present invention without deviating therefrom. Therefore, the present invention should not be limited to any single embodiment, but

20030076 PRO

rather construed in breadth and scope in accordance with the recitation of the appended claims.

Claims

What is claimed is:

1. A method for turbo-decoding for joint signal demodulation by a windowed-based Minimum Mean Squared Error (MMSE) which is easily extended to time-varying channels comprising the step of using an iterative decoding solution that exploits error correction codes.

Abstract

The invention solves the previously mentioned problem tightly integrating the following components: parameter estimation, symbol hypothesis testing, decoding, and rate identification. The approach assumes a module implemented prior to this module has identified the modulation scheme (e.g. CDMA). The solution presented here is a windowed-based Minimum Mean Squared Error (MMSE) approach to Turbo-decoding for joint signal demodulation, which is easily extended to time-varying channels. The invention described here is based on an iterative decoding solution that exploits error correction codes. In addition, the approach defined here is suitable for symbol asynchronous as well as symbol synchronous. Also, this algorithm is suitable for a variety of signaling schemes such as M-ary Phase Shift Keying (MPSK).

Drawings

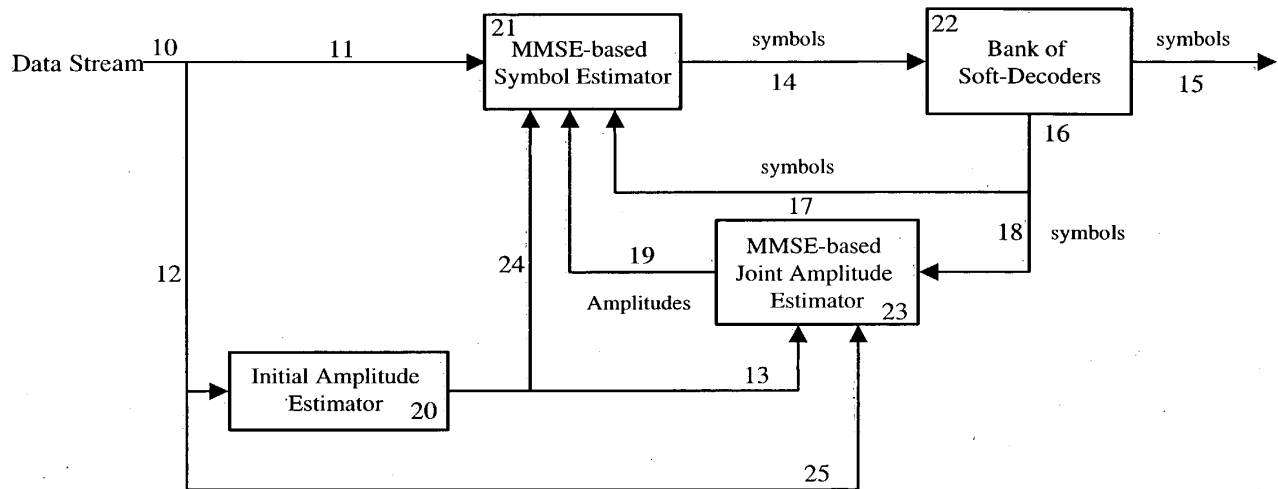


Figure 1

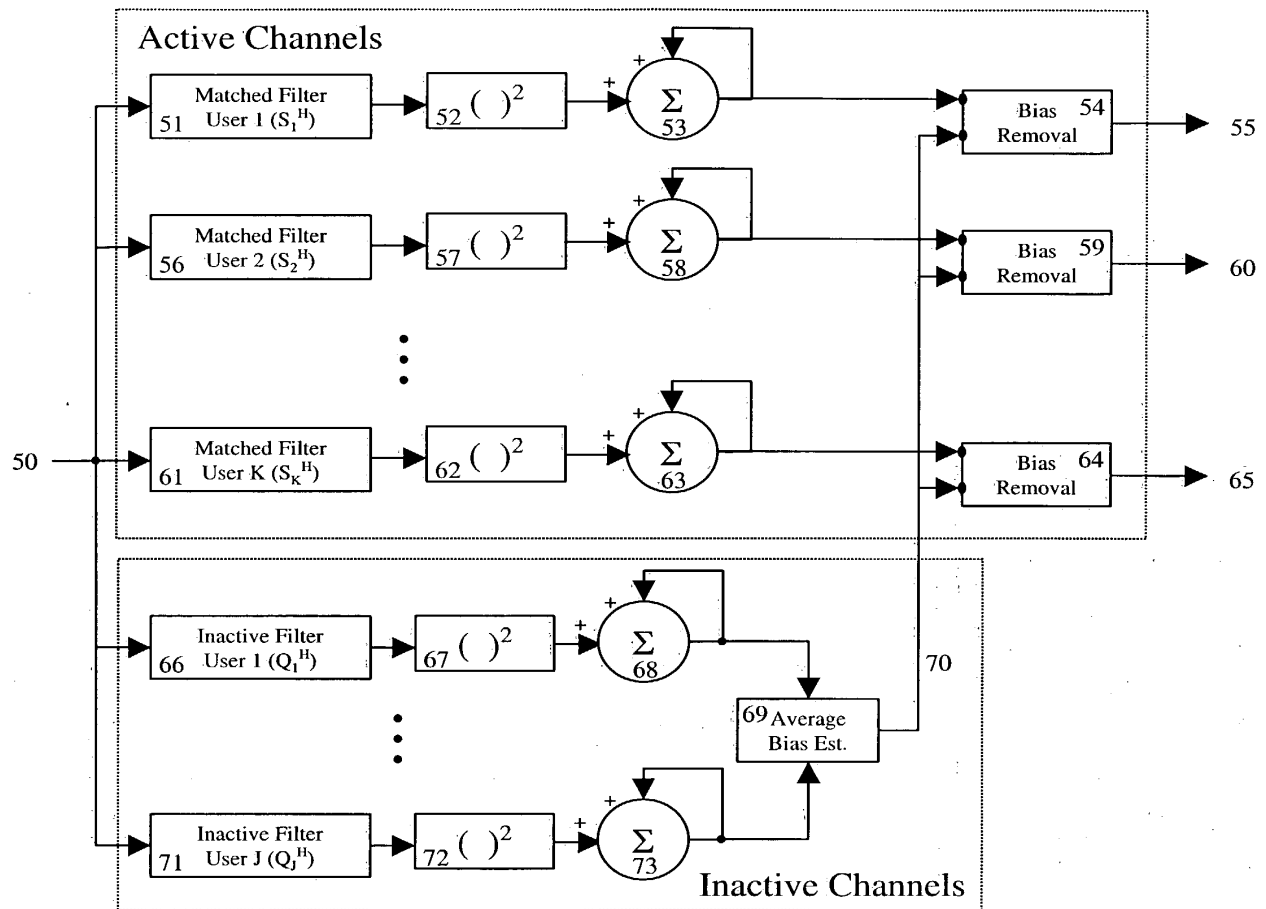


Figure 2

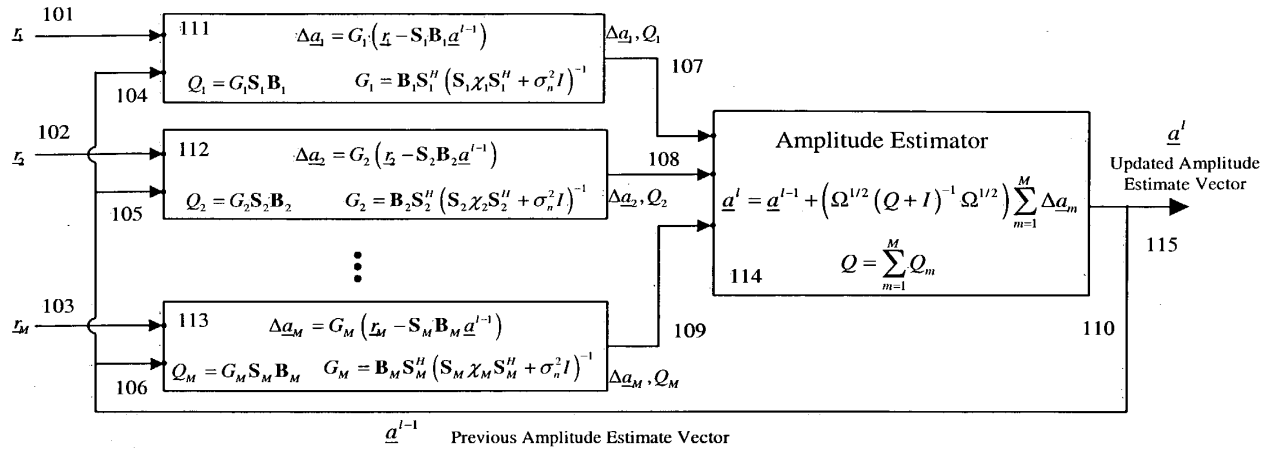


Figure 3

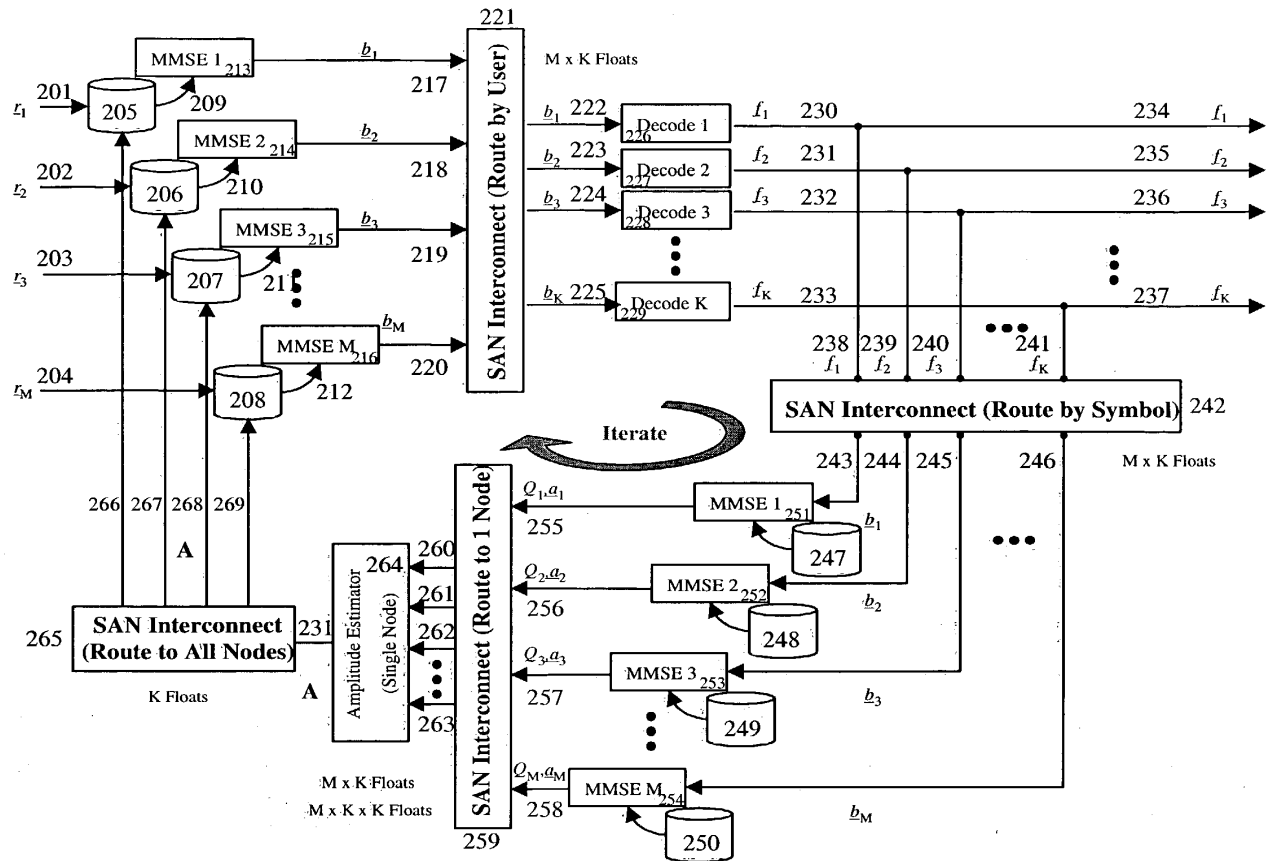


Figure 4

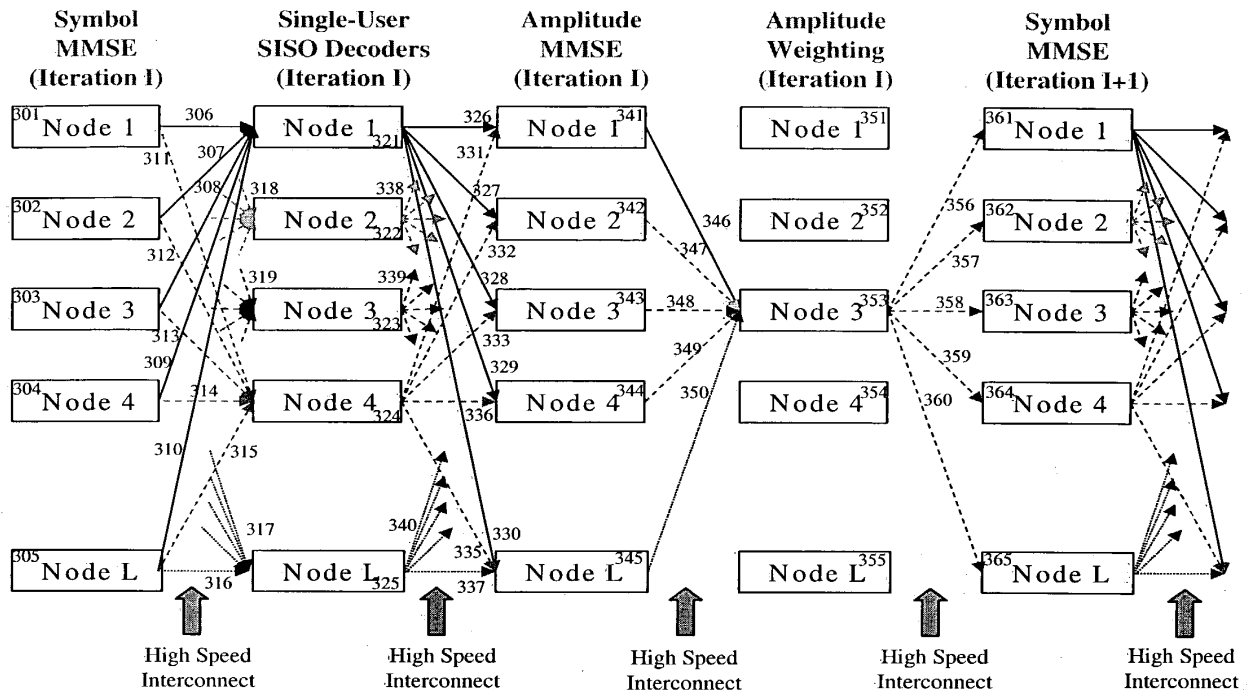


Figure 5