US007020829B2

(12) **United States Patent**
Eroz et al.

(10) **Patent No.: US 7,020,829 B2**
(45) **Date of Patent: Mar. 28, 2006**

(54) **METHOD AND SYSTEM FOR DECODING LOW DENSITY PARITY CHECK (LDPC) CODES**

(75) Inventors: **Mustafa Eroz**, Germantown, MD (US); **Feng-Wen Sun**, Germantown, MD (US); **Lin-Nan Lee**, Potomac, MD (US)

(73) Assignee: **Hughes Electronics Corporation**, El Segundo, CA (US)

(*) Notice: Subject to any disclaimer, the term of this patent is extended or adjusted under 35 U.S.C. 154(b) by 368 days.

(21) Appl. No.: **10/454,439**

(22) Filed: **Jun. 4, 2003**

(65) **Prior Publication Data**

US 2004/0005865 A1 Jan. 8, 2004

Related U.S. Application Data

(60) Provisional application No. 60/456,220, filed on Mar. 20, 2003, provisional application No. 60/447,641, filed on Feb. 14, 2003, provisional application No. 60/440,199, filed on Jan. 15, 2003, provisional application No. 60/423,710, filed on Nov. 4, 2002, provisional application No. 60/421,999, filed on Oct. 29, 2002, provisional application No. 60/421,505, filed on Oct. 25, 2002, provisional application No. 60/403,812, filed on Aug. 15, 2002, provisional application No. 60/398,760, filed on Jul. 26, 2002, and provisional application No. 60/393,457, filed on Jul. 3, 2002.

(51) **Int. Cl.**
H03M 13/00 (2006.01)
H03M 13/03 (2006.01)

(52) **U.S. Cl.** **714/794**; 714/781

(58) **Field of Classification Search** 714/794, 714/781

See application file for complete search history.

(56) **References Cited**

U.S. PATENT DOCUMENTS

6,539,367	B1	3/2003	Blanksby et al.	706/14
6,633,856	B1	10/2003	Richardson et al.	706/15
2002/0048329	A1 *	4/2002	Tran et al.	375/340
2002/0136318	A1 *	9/2002	Gorokhov et al.	375/261
2003/0037298	A1 *	2/2003	Eleftheriou et al.	714/752
2003/0065989	A1	4/2003	Yedida et al.	714/703
2003/0079171	A1	4/2003	Coe	714/758
2003/0081691	A1	5/2003	Sutskover et al.	375/286

FOREIGN PATENT DOCUMENTS

CA 2 310 186 12/2001

OTHER PUBLICATIONS

Zhang, T.; Parhi, K.K.; High–performance, low–complexity decoding of generalized low–density parity–check codes, IEEE Global Telecommunications Conference, vol.: 1, Nov. 25–29, 2001, pp.: 181–185.*

Boutros, J.; Pothier, O.; Zemor, G.; Generalized low density (Tanner) codes, 1999 IEEE International Conference on Communications, vol.: 1, Jun. 6–10, 1999, pp.: 441–445.*

Engling Yeo; Pakzad, P.; Nikolic, B.; Anantharam, V.; VLSI architectures for iterative decoders in magnetic recording channels, IEEE Transactions on Magnetics, vol.: 37, Issue: 2, Mar. 2001, pp.: 748–755.*

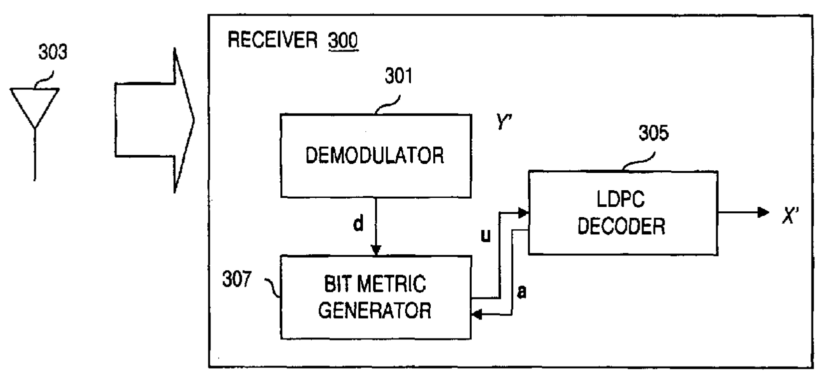
(Continued)

Primary Examiner—Joseph Torres
(74) *Attorney, Agent, or Firm*—John T. Whelan

(57) **ABSTRACT**

An approach is provided for transmitting messages using low density parity check (LDPC) codes. Input messages are encoded according to a structured parity check matrix that imposes restrictions on a sub-matrix of the parity check matrix to generate LDPC codes. The LDPC codes are transmitted over a radio communication system (e.g., satellite network), wherein a receiver communicating over the radio communication system is configured to iteratively decode the received LDPC codes according to a signal constellation associated with the LDPC codes. The receiver is configured to iteratively regenerating signal constellation bit metrics after one or more decoding iterations.

19 Claims, 16 Drawing Sheets



OTHER PUBLICATIONS

S.Y. Le Goff, "Channel capacity of bit–interleaved coded modulation schemes using 8–ary signal constellations", Electronics Letters, Feb. 14, 2002, vol. 28, No. 4.

R.G. Gallager, "Low Density Parity–Check Codes", XP–000992693, IRE Transactions of Information Theory, 1962, pp. 21–28.

Hisashi Futaki & Tomoaki Ohtsuki, "Performance of Low–Density Parity–Check (LDPC) Coded OFDM Systems", Tokyo University of Science, Faculty of Science and Technology, Tokyo University of Science, Japan, 2000 IEEE, pp. 1696–1700.

E. Eleftheriou & S. Olcer, "Low–Density Parity–Check Codes for Digital Subscriber Lines", IBM Research Zurich Research Laboratory, Switzerland, 2002 IEEE, pp. 1752–1757.

Tolga M. Duman & Masoud Salehi, "Performance Bounds for Turbo–Coded Modulation Systems", Transactions Papers, XP–002263494, 1999 IEEE, pp. 511–521.

S. ten Brink, J. Speidel and R.H. Yan, "Iterative demapping for QPSK Modulation", Electronics Letters, Jul. 23, 1998, vol. 34, No. 15.

Tong Zhang and Keshab K. Parhi, "Joint Code and Decoder Design for Implementation–Oriented (3,k)–regular LDPC Codes", Department of Electrical and Computer Engineering, University of Minnesota, Minneapolis, MN, USA, 2001 IEEE, pp. 1232–1236.

Thomas J. Richardson and Rudiger L. Urbanke, "Efficient Encoding of Low–Density Parity–Check Codes", XP–002965294, IEEE Transactions on Information Theory, vol. 47, No. 2, Feb. 2001, pp. 638–656.

K.R. Narayanan and J. Li, "Bandwidth Efficient Low Density Parity Check Coding Using Multi Level Coding And Iterative Multi Stage Decoding", Texas A&M University, College Station, TX, USA, 2^{nd} International Symposium on Turbo–Codes & Related Topics–Brest–France, 2000, XP009021988, pp. 165–168.

David J. C. MacKay, "Good Error–Correcting Codes Based on Very Sparse Matrices", XP–002143042, IEEE Transaction on Information Theory, vol. 45, No. 2, Mar. 1999, pp. 399–431.

Xiaodong Li and J.A. Ritcey, "Bit–Interleaned Coded Modulation With Iterative Decoding Using Soft Feedback", Electronics Letters, May 14, 1998, vol. 34, No. 10.

* cited by examiner

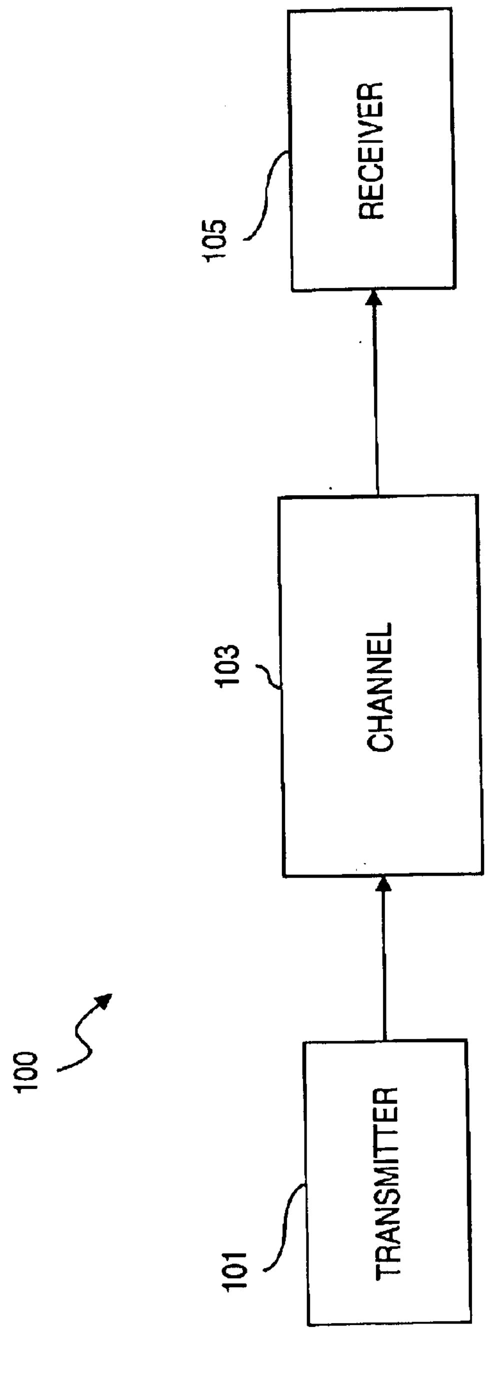


FIG. 1

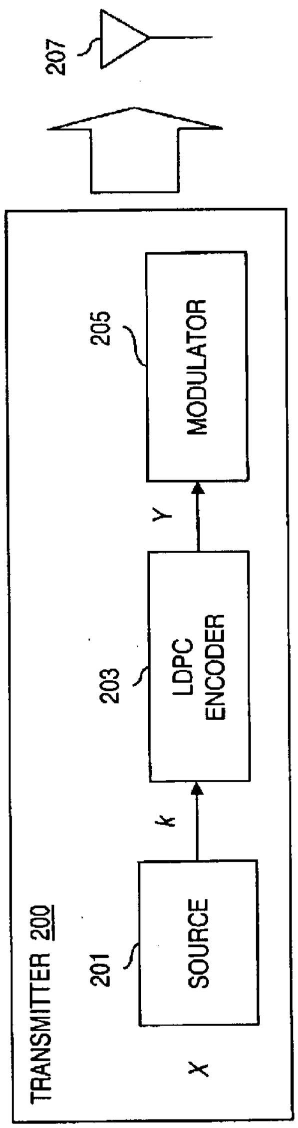

207
TRANSMITTER 200
201
SOURCE
X
k
203
LDPC
ENCODER
Y
205
MODULATOR

FIG. 2

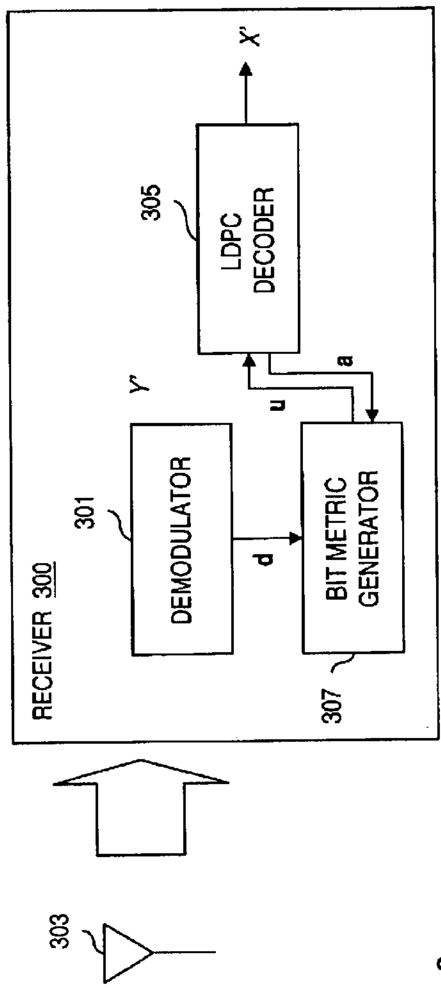

303
RECEIVER 300
301
DEMODULATOR
d
BIT METRIC
GENERATOR
307
Y'
u
a
305
LDPC
DECODER
X'

FIG. 3

$$H = \begin{array}{c} \\ \end{array}\begin{array}{cccccccc} n_1 & n_2 & n_3 & n_4 & n_5 & n_6 & n_7 & n_8 \end{array}$$

$$H = \begin{bmatrix} 1 & 0 & 0 & 1 & 1 & 0 & 0 & 1 \\ 0 & 1 & 1 & 0 & 1 & 0 & 1 & 0 \\ 1 & 0 & 1 & 0 & 0 & 1 & 0 & 1 \\ 0 & 1 & 0 & 1 & 0 & 1 & 1 & 0 \end{bmatrix} \begin{matrix} m_1 \\ m_2 \\ m_3 \\ m_4 \end{matrix}$$

FIG. 4

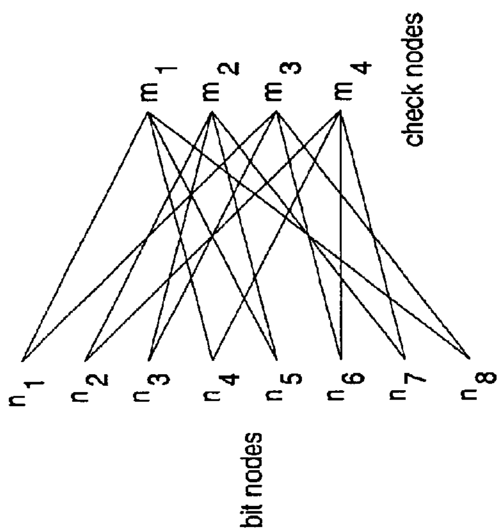


FIG. 5

$$B = \begin{bmatrix} 1 & & & & & \mathbf{0} \\ x & 1 & & & & \\ x & x & 1 & & & \\ x & x & x & 1 & & \\ x & x & x & x & 1 & \\ x & x & x & x & x & 1 \end{bmatrix}$$

FIG. 6

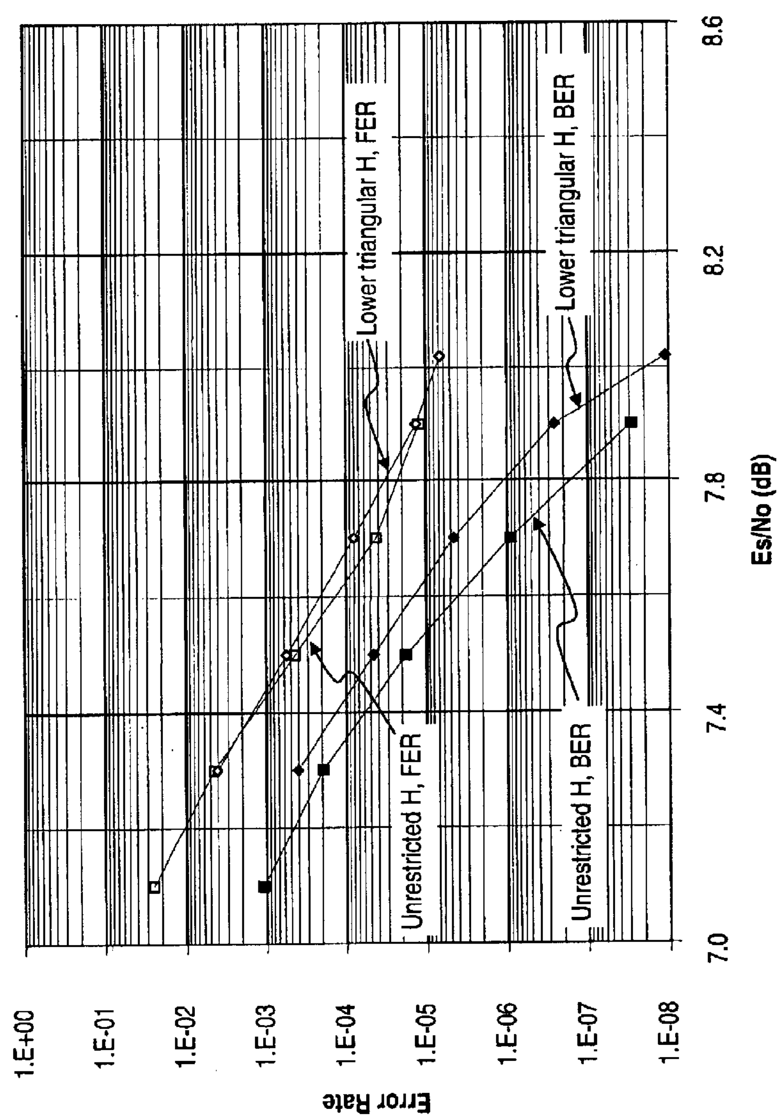

Lower triangular H, FER
Lower triangular H, BER
Unrestricted H, FER
Unrestricted H, BER
Error Rate
1.E+00
1.E-01
1.E-02
1.E-03
1.E-04
1.E-05
1.E-06
1.E-07
1.E-08
7.0
7.4
7.8
8.2
8.6
Es/No (dB)

FIG. 7

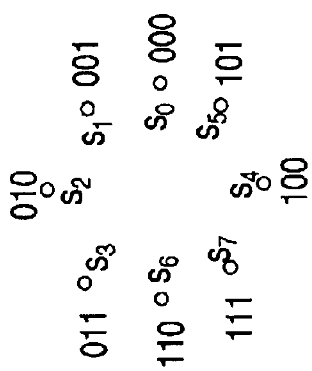

010
S_2
S_1 001
S_0 000
S_5 101
S_4
100
011 S_3
110 S_6
111 S_7

FIG. 8A

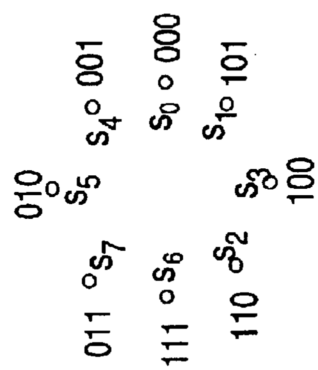

010
S_5
S_4 001
S_0 000
S_1 101
S_3
100
011 S_7
111 S_6
110 S_2

FIG. 8B

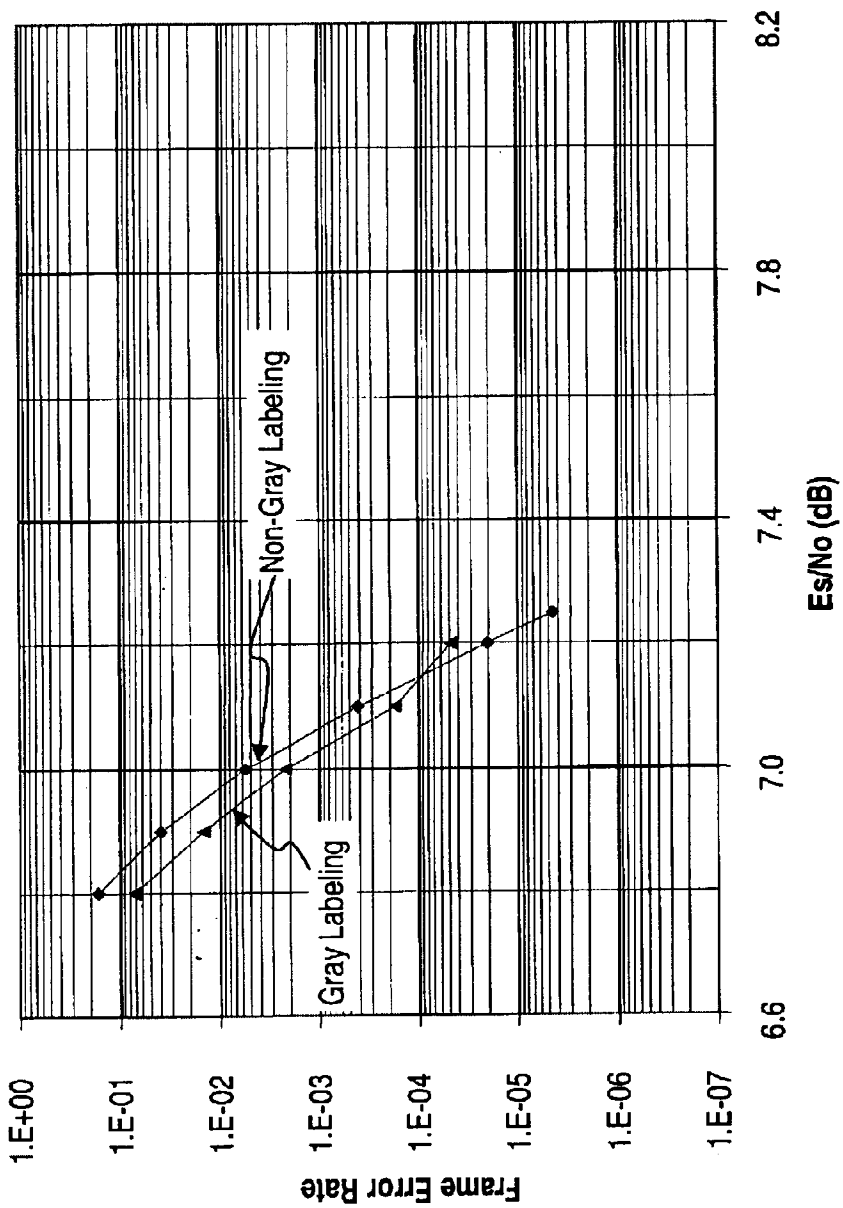

Non-Gray Labeling
Gray Labeling
Es/No (dB)
6.6
7.0
7.4
7.8
8.2
Frame Error Rate
1.E+00
1.E-01
1.E-02
1.E-03
1.E-04
1.E-05
1.E-06
1.E-07

FIG. 9

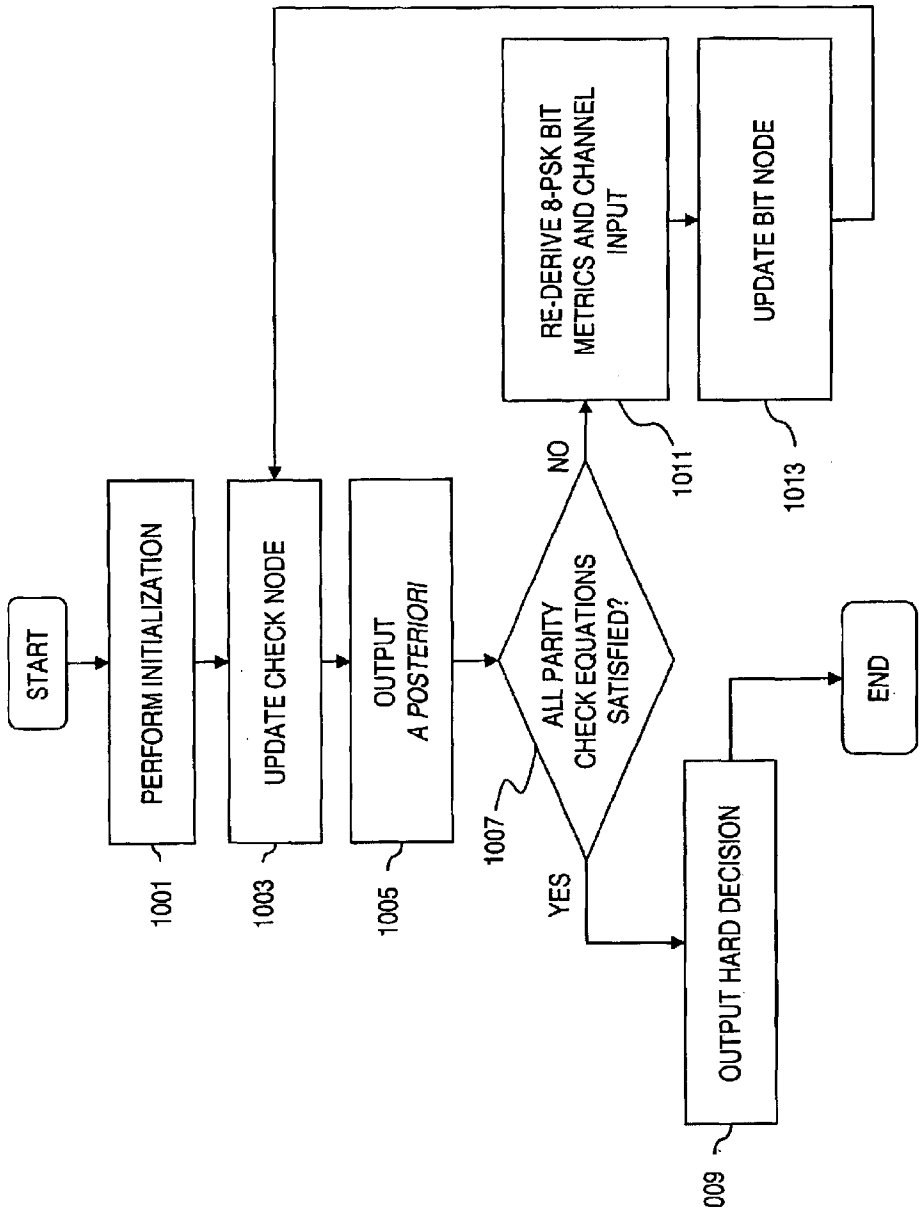

START
PERFORM INITIALIZATION
1001
UPDATE CHECK NODE
1003
OUTPUT
A POSTERIORI
1005
ALL PARITY
CHECK EQUATIONS
SATISFIED?
1007
NO
YES
RE-DERIVE 8-PSK BIT
METRICS AND CHANNEL
INPUT
1011
UPDATE BIT NODE
1013
OUTPUT HARD DECISION
1009
END

FIG. 10

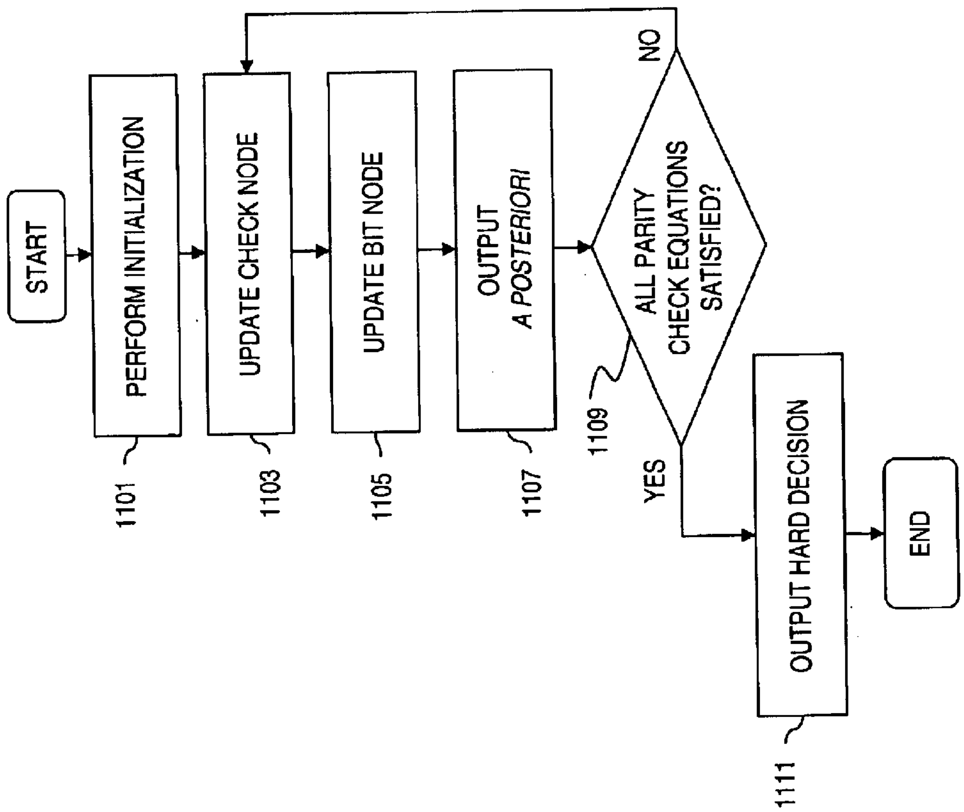

START
PERFORM INITIALIZATION
1101
UPDATE CHECK NODE
1103
UPDATE BIT NODE
1105
OUTPUT
A POSTERIORI
1107
ALL PARITY
CHECK EQUATIONS
SATISFIED?
1109
NO
YES
OUTPUT HARD DECISION
1111
END

FIG. 11

FIG. 12A

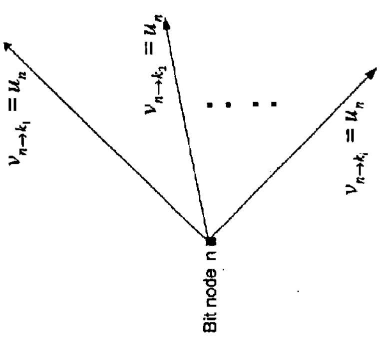
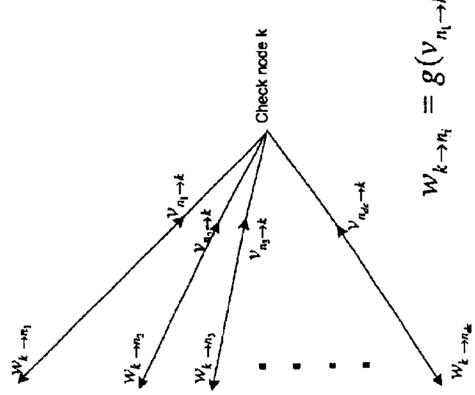
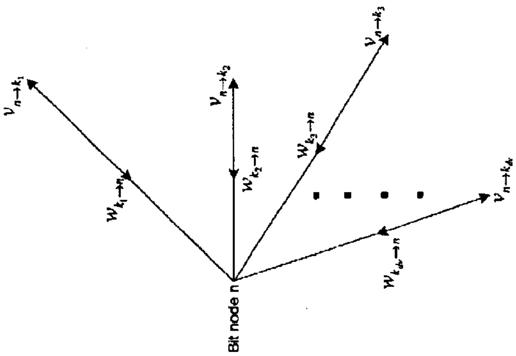


FIG. 12B



$$w_{k\to n_i} = g(v_{n_1\to k}, v_{n_2\to k}, \ldots, v_{n_{i-1}\to k}, v_{n_{i+1}\to k}, \ldots, v_{n_{dc}\to k})$$

FIG. 12C



$$v_{n\to k_i} = u_n + \sum_{j\neq i} w_{k_j\to n}$$

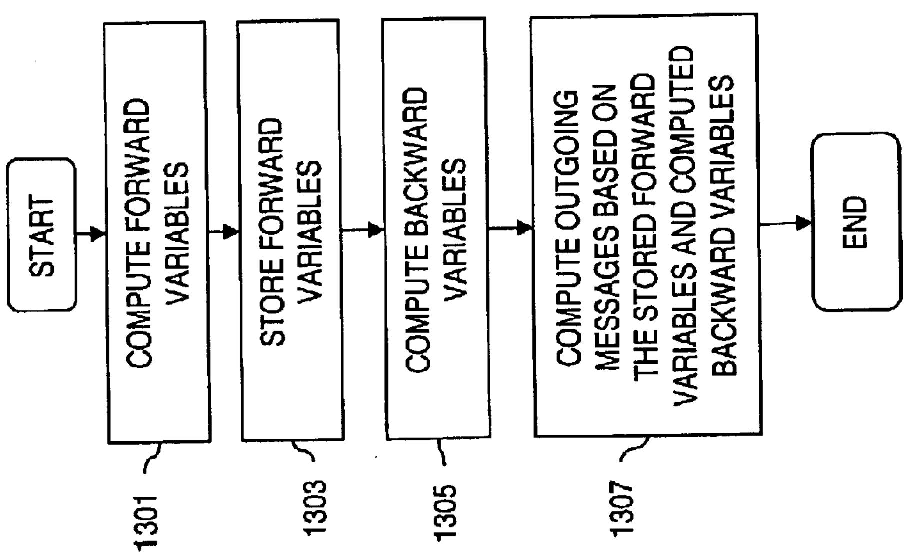


FIG. 13A

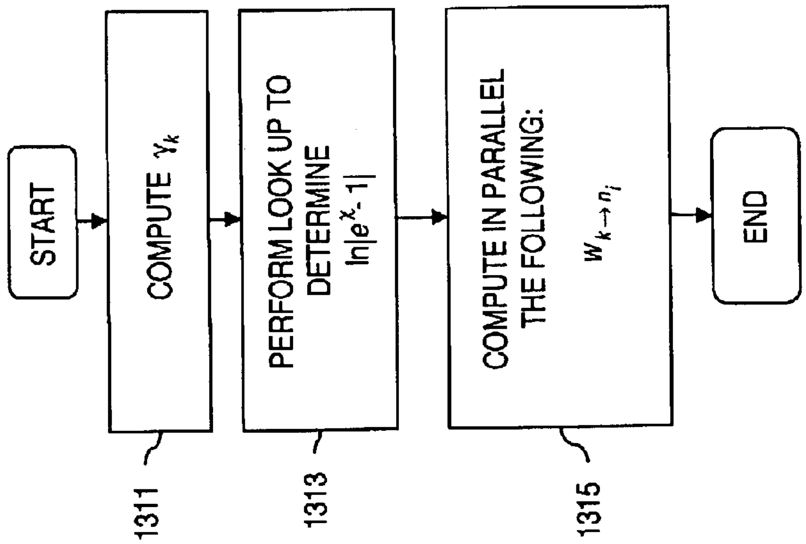
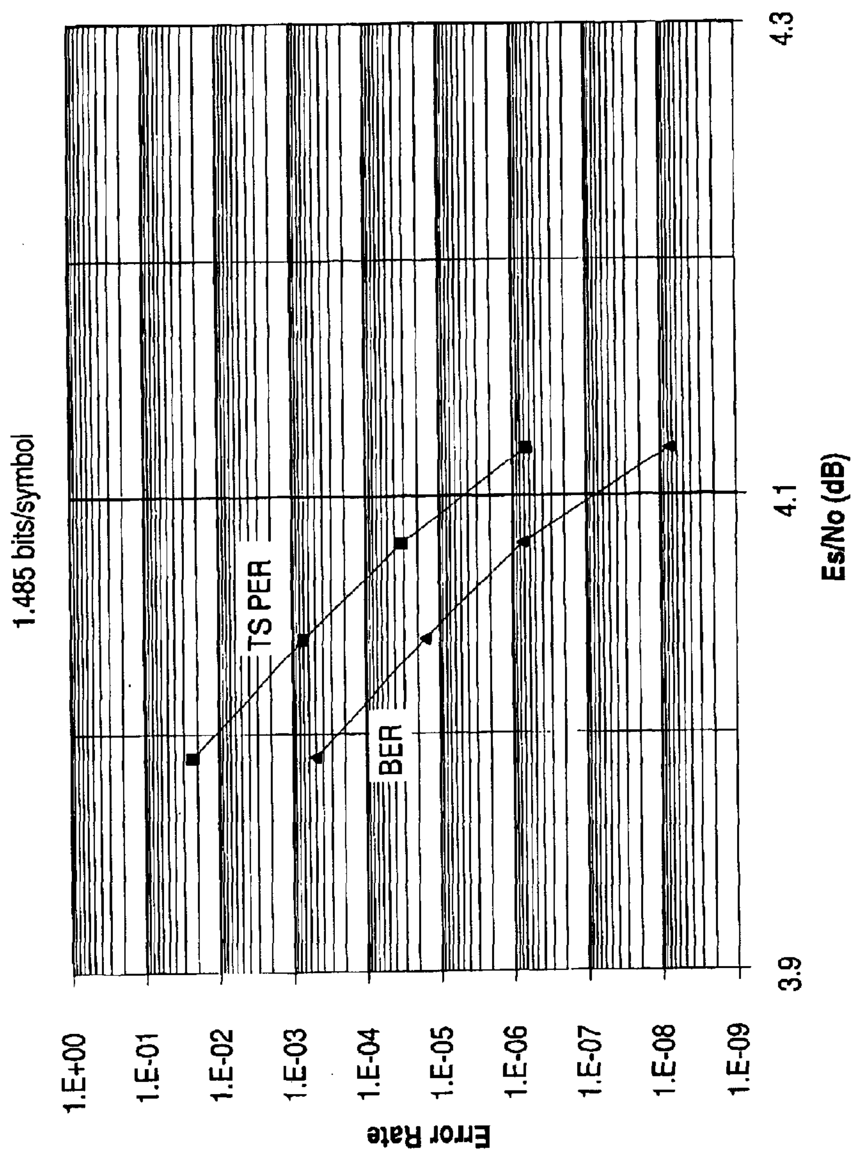


FIG. 13B



1.485 bits/symbol
TS PER
BER
Error Rate
1.E+00
1.E-01
1.E-02
1.E-03
1.E-04
1.E-05
1.E-06
1.E-07
1.E-08
1.E-09
3.9
4.1
4.3
Es/No (dB)

FIG. 14A

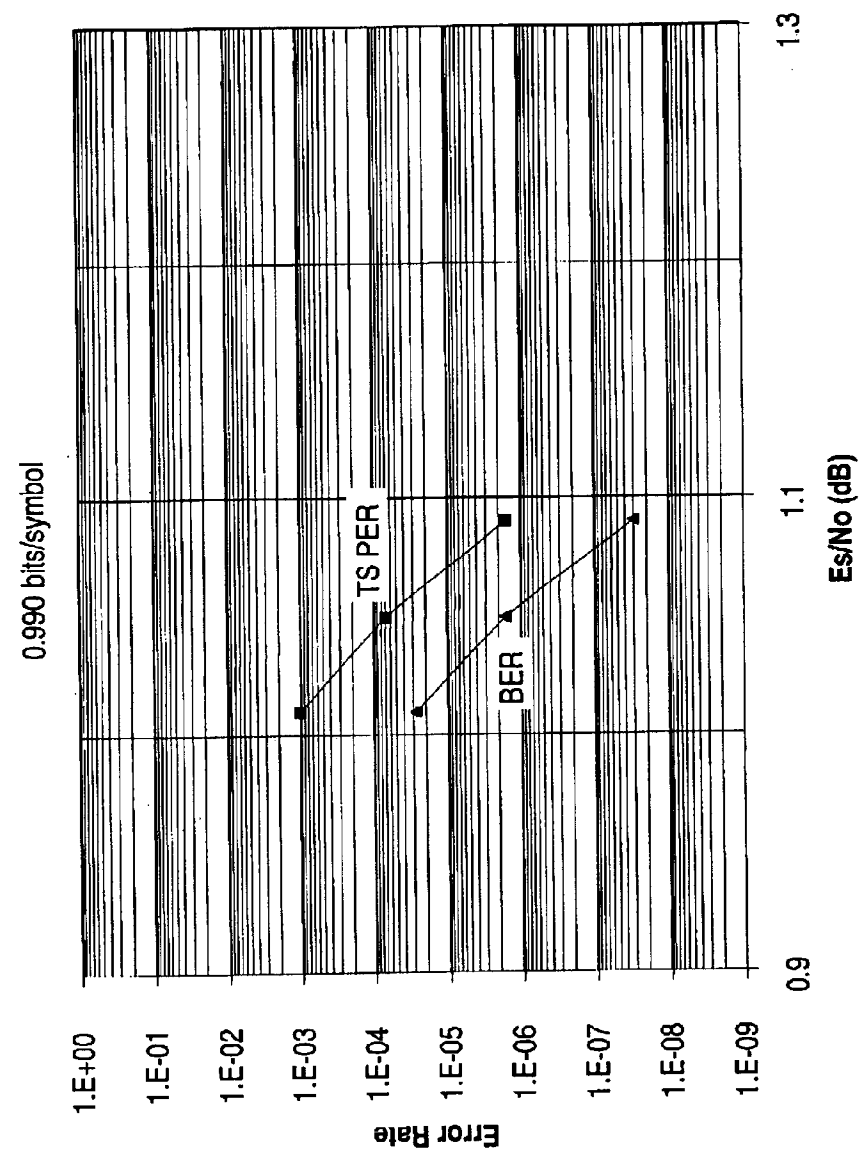

0.990 bits/symbol
TS PER
BER
Es/No (dB)
Error Rate
0.9
1.1
1.3
1.E+00
1.E-01
1.E-02
1.E-03
1.E-04
1.E-05
1.E-06
1.E-07
1.E-08
1.E-09

FIG. 14B

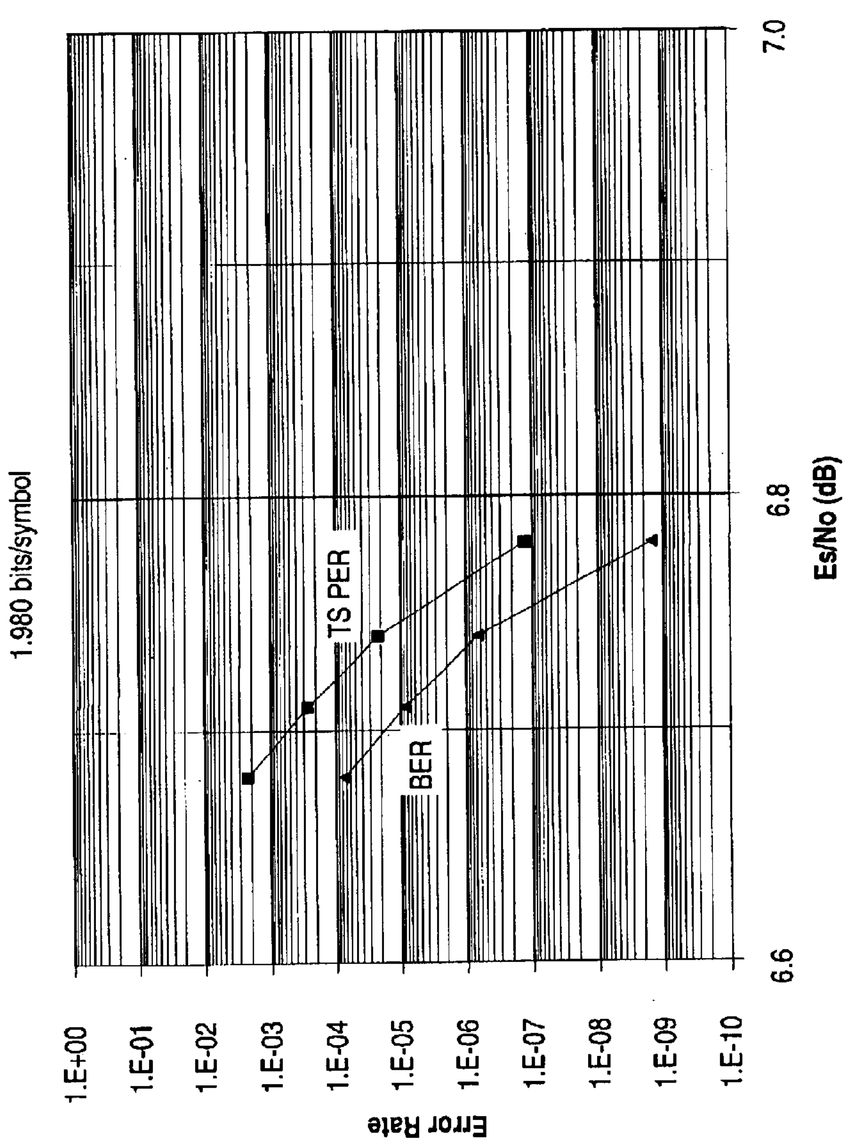

1.980 bits/symbol
TS PER
BER
Error Rate
1.E+00
1.E-01
1.E-02
1.E-03
1.E-04
1.E-05
1.E-06
1.E-07
1.E-08
1.E-09
1.E-10
6.6
6.8
7.0
Es/No (dB)

FIG. 14C

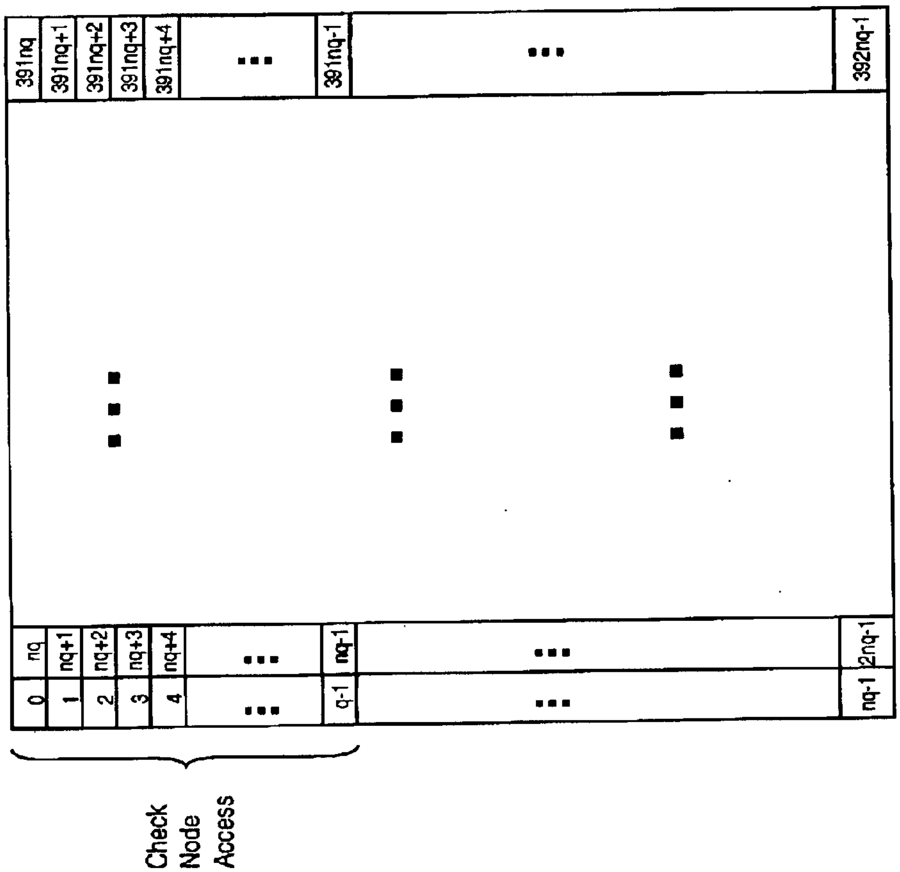

391nq
391nq+1
391nq+2
391nq+3
391nq+4
391nq-1
392nq-1
nq
nq+1
nq+2
nq+3
nq+4
nq-1
2nq-1
0
1
2
3
4
q-1
nq-1
Check Node Access

FIG. 15A

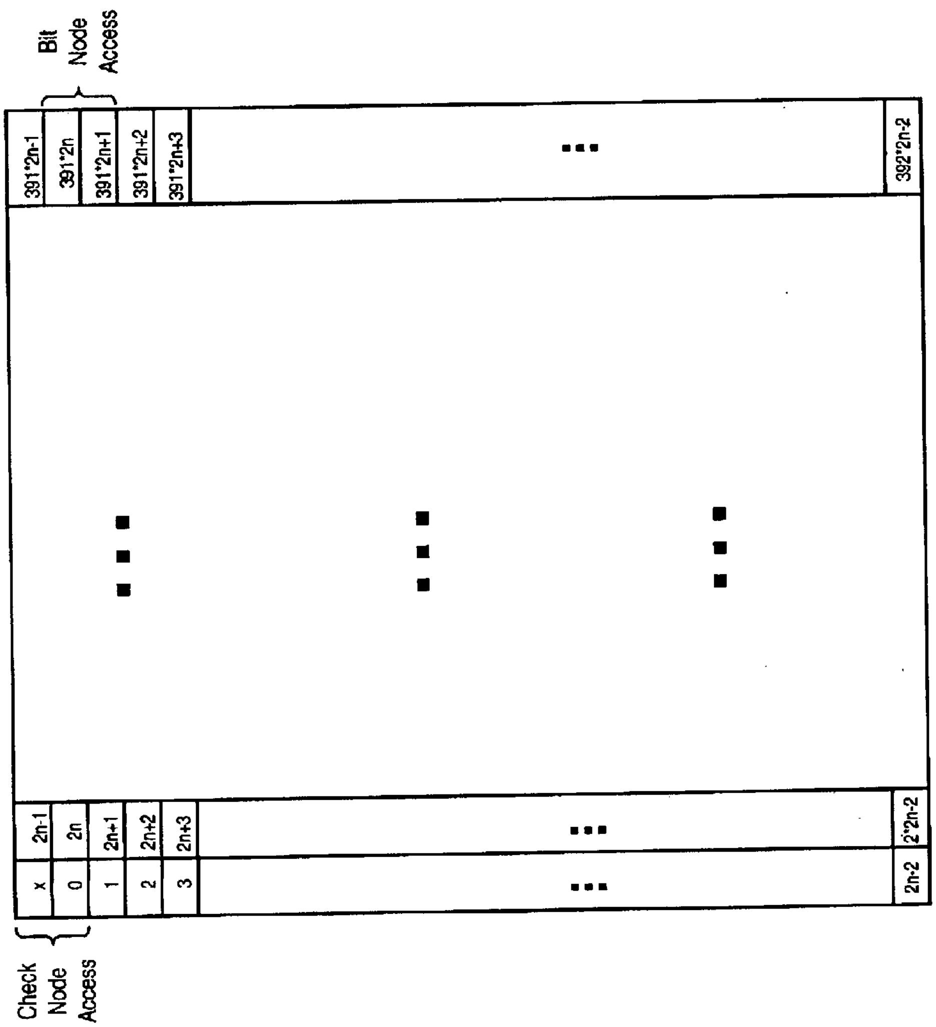

Bit Node Access
391*2n-1
391*2n
391*2n+1
391*2n+2
391*2n+3
392*2n-2
2n-1
2n
2n+1
2n+2
2n+3
2*2n-2
x
0
1
2
3
2n-2
Check Node Access

FIG. 15B

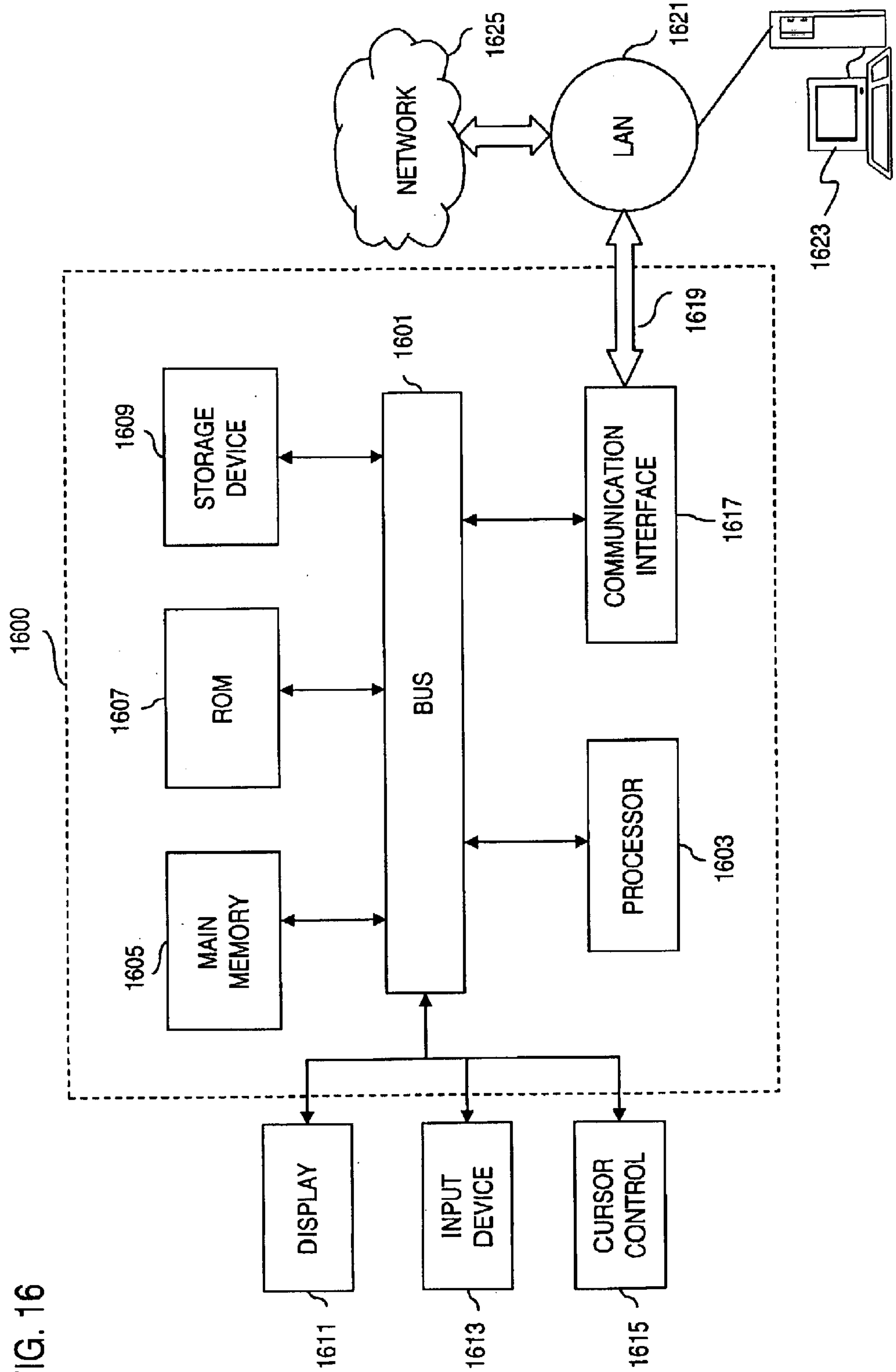

1625
NETWORK
1621
LAN
1623
1600
1609
STORAGE DEVICE
1607
ROM
1605
MAIN MEMORY
1601
BUS
1619
COMMUNICATION INTERFACE
1617
PROCESSOR
1603
DISPLAY
1611
INPUT DEVICE
1613
CURSOR CONTROL
1615

FIG. 16

METHOD AND SYSTEM FOR DECODING LOW DENSITY PARITY CHECK (LDPC) CODES

RELATED APPLICATIONS

This application is related to, and claims the benefit of the earlier filing date under 35 U.S.C. §119(e) of, U.S. Provisional Patent Application Ser. No. 60/393,457 filed Jul. 3, 2002, entitled "Code Design and Implementation Improvements for Low Density Parity Check Codes," U.S. Provisional Patent Application Ser. No. 60/398,760 filed Jul. 26, 2002, entitled "Code Design and Implementation Improvements for Low Density Parity Check Codes," U.S. Provisional Patent Application Ser. No. 60/403,812 filed Aug. 15, 2002, entitled "Power and Bandwidth Efficient Modulation and Coding Scheme for Direct Broadcast Satellite and Broadcast Satellite Communications," U.S. Provisional Patent Application Ser. No. 60/421,505, filed Oct. 25, 2002, entitled "Method and System for Generating Low Density Parity Check Codes," U.S. Provisional Patent Application Ser. No. 60/421,999 filed Oct. 29, 2002, entitled "Satellite Communication System Utilizing Low Density Parity Check Codes," U.S. Provisional Patent Application Ser. No. 60/423,710, entitled "Code Design and Implementation Improvements for Low Density Parity Check Codes," U.S. Provisional Patent Application Ser. No. 60/440,199 filed Jan. 15, 2003, entitled "Novel Solution to Routing Problem in Low Density Parity Check Decoders," U.S. Provisional Patent Application Ser. No. 60/447,641 filed Feb. 14, 2003 entitled "Low Density Parity Check Code Encoder Design," and U.S. Provisional Patent Application Ser. No. 60/456,220 filed Mar. 20, 2003, entitled "Description LDPC and BCH Encoders"; the entireties of which are incorporated herein by reference.

FIELD OF THE INVENTION

The present invention relates to communication systems, and more particularly to coded systems.

BACKGROUND OF THE INVENTION

Communication systems employ coding to ensure reliable communication across noisy communication channels. These communication channels exhibit a fixed capacity that can be expressed in terms of bits per symbol at certain signal to noise ratio (SNR), defining a theoretical upper limit (known as the Shannon limit). As a result, coding design has aimed to achieve rates approaching this Shannon limit. One such class of codes that approach the Shannon limit is Low Density Parity Check (LDPC) codes.

Traditionally, LDPC codes have not been widely deployed because of a number of drawbacks. One drawback is that the LDPC encoding technique is highly complex. Encoding an LDPC code using its generator matrix would require storing a very large, non-sparse matrix. Additionally, LDPC codes require large blocks to be effective; consequently, even though parity check matrices of LDPC codes are sparse, storing these matrices is problematic.

From an implementation perspective, a number of challenges are confronted. For example, storage is an important reason why LDPC codes have not become widespread in practice. Also, a key challenge in LDPC code implementation has been how to achieve the connection network between several processing engines (nodes) in the decoder. Further, the computational load in the decoding process, specifically the check node operations, poses a problem.

Therefore, there is a need for a LDPC communication system that employs simple encoding and decoding processes. There is also a need for using LDPC codes efficiently to support high data rates, without introducing greater complexity. There is also a need to improve performance of LDPC encoders and decoders. There is also a need to minimize storage requirements for implementing LDPC coding. There is a further need for a scheme that simplifies the communication between processing nodes in the LDPC decoder.

SUMMARY OF THE INVENTION

These and other needs are addressed by the present invention, wherein an approach for decoding a structured Low Density Parity Check (LDPC) codes is provided. Structure of the LDPC codes is provided by restricting portion part of the parity check matrix to be lower triangular and/or satisfying other requirements such that the communication between processing nodes of the decoder becomes very simple. Also, the approach can advantageously exploit the unequal error protecting capability of LDPC codes on transmitted bits to provide extra error protection to more vulnerable bits of high order modulation constellations (such as 8-PSK (Phase Shift Keying)). The decoding process involves iteratively regenerating signal constellation bit metrics into an LDPC decoder after each decoder iteration or several decoder iterations. The above arrangement provides a computational efficient approach to decoding LDPC codes.

According to one aspect of an embodiment of the present invention, a method for decoding low density parity check (LDPC) codes is disclosed. The method includes receiving a priori probability information based on distance vector information relating to distances between received noisy symbol points and symbol points of a signal constellation associated with the LDPC codes. The method also includes transmitting a posteriori probability information based on the a priori probability information. The method includes determining whether parity check equations associated with the LDPC codes are satisfied according to the a priori probability and the a posteriori probability information. Additionally, the method includes selectively regenerating the signal constellation bit metrics based on the determining step. Further, the method includes outputting decoded messages based on the regenerated signal constellation bit metrics.

According to another aspect of an embodiment of the present invention, a system for decoding low density parity check (LDPC) codes is disclosed. The system includes means for receiving a priori probability information based on distance vector information relating to distances between received noisy symbol points and symbol points of a signal constellation associated with the LDPC codes. The system also includes means for transmitting a posteriori probability information based on the a priori probability information. Additionally, the system includes means for determining whether parity check equations associated with the LDPC codes are satisfied according to the a priori probability and the a posteriori probability information. The system includes means for selectively regenerating the signal constellation bit metrics based on the determination. Further, the system includes means for outputting decoded messages based on the regenerated signal constellation bit metrics.

According to another aspect of an embodiment of the present invention, a receiver for decoding low density parity check (LDPC) codes is disclosed. The receiver includes a bit metric generator configured to generate a priori probability information based on distance vector information relating to distances between received noisy symbol points and symbol points of a signal constellation associated with the LDPC codes. The receiver also includes a decoder configured to output a posteriori probability information based on the a priori probability information received from the bit metric generator, wherein the decoder is further configured to determine whether parity check equations associated with the LDPC codes are satisfied according to the a priori probability and the a posteriori probability information. The decoder outputs decoded messages based on a regenerated signal constellation bit metrics if the parity check equations are not satisfied.

According to another aspect of an embodiment of the present invention, a method for transmitting messages using low density parity check (LDPC) codes is disclosed. The method includes encoding input messages according to a structured parity check matrix that imposes restrictions on a sub-matrix of the parity check matrix to generate LDPC codes. The method also includes transmitting the LDPC codes over a radio communication system, wherein a receiver communicating over the radio communication system is configured to iteratively decode the received LDPC codes according to a signal constellation associated with the LDPC codes. The receiver is configured to iteratively regenerating signal constellation bit metrics after one or more decoding iterations.

Still other aspects, features, and advantages of the present invention are readily apparent from the following detailed description, simply by illustrating a number of particular embodiments and implementations, including the best mode contemplated for carrying out the present invention. The present invention is also capable of other and different embodiments, and its several details can be modified in various obvious respects, all without departing from the spirit and scope of the present invention. Accordingly, the drawing and description are to be regarded as illustrative in nature, and not as restrictive.

BRIEF DESCRIPTION OF THE DRAWINGS

The present invention is illustrated by way of example, and not by way of limitation, in the figures of the accompanying drawings and in which like reference numerals refer to similar elements and in which:

FIG. **1** is a diagram of a communications system configured to utilize Low Density Parity Check (LDPC) codes, according to an embodiment of the present invention;

FIG. **2** is a diagram of an exemplary transmitter in the system of FIG. **1**;

FIG. **3** is a diagram of an exemplary receiver in the system of FIG. **1**;

FIG. **4** is a diagram of a sparse parity check matrix, in accordance with an embodiment of the present invention;

FIG. **5** is a diagram of a bipartite graph of an LDPC code of the matrix of FIG. **4**;

FIG. **6** is a diagram of a sub-matrix of a sparse parity check matrix, wherein the sub-matrix contains parity check values restricted to the lower triangular region, according to an embodiment of the present invention;

FIG. **7** is a graph showing performance between codes utilizing unrestricted parity check matrix (H matrix) versus restricted H matrix having a sub-matrix as in FIG. **6**;

FIGS. **8**A and **8**B are, respectively, a diagram of a non-Gray 8-PSK modulation scheme, and a Gray 8-PSK modulation, each of which can be used in the system of FIG. **1**;

FIG. **9** is a graph showing performance between codes utilizing Gray labeling versus non-Gray labeling;

FIG. **10** is a flow chart of the operation of the LDPC decoder using non-Gray mapping, according to an embodiment of the present invention;

FIG. **11** is a flow chart of the operation of the LDPC decoder of FIG. **3** using Gray mapping, according to an embodiment of the present invention;

FIGS. **12**A–**12**C are diagrams of the interactions between the check nodes and the bit nodes in a decoding process, according to an embodiment of the present invention;

FIGS. **13**A and **13**B are flowcharts of processes for computing outgoing messages between the check nodes and the bit nodes using, respectively, a forward-backward approach and a parallel approach, according to various embodiments of the present invention;

FIGS. **14**A–**14** are graphs showing simulation results of LDPC codes generated in accordance with various embodiments of the present invention;

FIGS. **15**A and **15**B are diagrams of the top edge and bottom edge, respectively, of memory organized to support structured access as to realize randomness in LDPC coding, according to an embodiment of the present invention; and

FIG. **16** is a diagram of a computer system that can perform the processes of encoding and decoding of LDPC codes, in accordance with embodiments of the present invention.

DESCRIPTION OF THE PREFERRED EMBODIMENT

A system, method, and software for efficiently decoding structured Low Density Parity Check (LDPC) codes are described. In the following description, for the purposes of explanation, numerous specific details are set forth in order to provide a thorough understanding of the present invention. It is apparent, however, to one skilled in the art that the present invention may be practiced without these specific details or with an equivalent arrangement. In other instances, well-known structures and devices are shown in block diagram form in order to avoid unnecessarily obscuring the present invention.

FIG. **1** is a diagram of a communications system configured to utilize Low Density Parity Check (LDPC) codes, according to an embodiment of the present invention. A digital communications system **100** includes a transmitter **101** that generates signal waveforms across a communication channel **103** to a receiver **105**. In this discrete communications system **100**, the transmitter **101** has a message source that produces a discrete set of possible messages; each of the possible messages has a corresponding signal waveform. These signal waveforms are attenuated, or otherwise altered, by communications channel **103**. To combat the noise channel **103**, LDPC codes are utilized.

The LDPC codes that are generated by the transmitter **101** enables high speed implementation without incurring any performance loss. These structured LDPC codes output from the transmitter **101** avoid assignment of a small number of check nodes to the bit nodes already vulnerable to channel errors by virtue of the modulation scheme (e.g., 8-PSK).

Such LDPC codes have a parallelizable decoding algorithm (unlike turbo codes), which advantageously involves simple operations such as addition, comparison and table look-up. Moreover, carefully designed LDPC codes do not exhibit any sign of error floor.

According to one embodiment of the present invention, the transmitter **101** generates, using a relatively simple encoding technique, LDPC codes based on parity check matrices (which facilitate efficient memory access during decoding) to communicate with the receiver **105**. The transmitter **101** employs LDPC codes that can outperform concatenated turbo+RS (Reed-Solomon) codes, provided the block length is sufficiently large.

FIG. **2** is a diagram of an exemplary transmitter in the system of FIG. **1**. A transmitter **200** is equipped with an LDPC encoder **203** that accepts input from an information source **201** and outputs coded stream of higher redundancy suitable for error correction processing at the receiver **105**. The information source **201** generates k signals from a discrete alphabet, X. LDPC codes are specified with parity check matrices. On the other hand, encoding LDPC codes require, in general, specifying the generator matrices. Even though it is possible to obtain generator matrices from parity check matrices using Gaussian elimination, the resulting matrix is no longer sparse and storing a large generator matrix can be complex.

Encoder **203** generates signals from alphabet Y to a modulator **205** using a simple encoding technique that makes use of only the parity check matrix by imposing structure onto the parity check matrix. Specifically, a restriction is placed on the parity check matrix by constraining certain portion of the matrix to be triangular. The construction of such a parity check matrix is described more fully below in FIG. **6**. Such a restriction results in negligible performance loss, and therefore, constitutes an attractive trade-off.

Modulator **205** maps the encoded messages from encoder **203** to signal waveforms that are transmitted to a transmit antenna **207**, which emits these waveforms over the communication channel **103**. Accordingly, the encoded messages are modulated and distributed to a transmit antenna **207**. The transmissions from the transmit antenna **207** propagate to a receiver, as discussed below.

FIG. **3** is a diagram of an exemplary receiver in the system of FIG. **1**. At the receiving side, a receiver **300** includes a demodulator **301** that performs demodulation of received signals from transmitter **200**. These signals are received at a receive antenna **303** for demodulation. After demodulation, the received signals are forwarded to a decoder **305**, which attempts to reconstruct the original source messages by generating messages, X', in conjunction with a bit metric generator **307**. With non-Gray mapping, the bit metric generator **307** exchanges probability information with the decoder **305** back and forth (iteratively) during the decoding process, which is detailed in FIG. **10**. Alternatively, if Gray mapping is used (according to one embodiment of the present invention), one pass of the bit metric generator is sufficient, in which further attempts of bit metric generation after each LDPC decoder iteration are likely to yield limited performance improvement; this approach is more fully described with respect to FIG. **11**. To appreciate the advantages offered by the present invention, it is instructive to examine how LDPC codes are generated, as discussed in FIG. **4**.

FIG. **4** is a diagram of a sparse parity check matrix, in accordance with an embodiment of the present invention. LDPC codes are long, linear block codes with sparse parity check matrix $H_{(n-k)xn}$. Typically the block length, n, ranges from thousands to tens of thousands of bits. For example, a parity check matrix for an LDPC code of length n=8 and rate ½ is shown in FIG. **4**. The same code can be equivalently represented by the bipartite graph, per FIG. **5**.

FIG. **5** is a diagram of a bipartite graph of an LDPC code of the matrix of FIG. **4**. Parity check equations imply that for each check node, the sum (over GF (Galois Field)(2)) of all adjacent bit nodes is equal to zero. As seen in the figure, bit nodes occupy the left side of the graph and are associated with one or more check nodes, according to a predetermined relationship. For example, corresponding to check node m_1, the following expression exists $n_1+n_4+n_5+n_8=0$ with respect to the bit nodes.

Returning the receiver **303**, the LDPC decoder **305** is considered a message passing decoder, whereby the decoder **305** aims to find the values of bit nodes. To accomplish this task, bit nodes and check nodes iteratively communicate with each other. The nature of this communication is described below.

From check nodes to bit nodes, each check node provides to an adjacent bit node an estimate ("opinion") regarding the value of that bit node based on the information coming from other adjacent bit nodes. For instance, in the above example if the sum of n_4, n_5 and n_8 "looks like" 0 to m_1, then m_1 would indicate to n_1 that the value of n_1 is believed to be 0 (since $n_1+n_4+n_5+n_8=0$); otherwise m_1 indicate to n_1 that the value of n_1 is believed to be 1. Additionally, for soft decision decoding, a reliability measure is added.

From bit nodes to check nodes, each bit node relays to an adjacent check node an estimate about its own value based on the feedback coming from its other adjacent check nodes. In the above example n_1 has only two adjacent check nodes m_1 and m_3. If the feedback coming from m_3 to n_1 indicates that the value of n_1 is probably 0, then n_1 would notify m_1 that an estimate of n_1's own value is 0. For the case in which the bit node has more than two adjacent check nodes, the bit node performs a majority vote (soft decision) on the feedback coming from its other adjacent check nodes before reporting that decision to the check node it communicates. The above process is repeated until all bit nodes are considered to be correct (i.e., all parity check equations are satisfied) or until a predetermined maximum number of iterations is reached, whereby a decoding failure is declared.

FIG. **6** is a diagram of a sub-matrix of a sparse parity check matrix, wherein the sub-matrix contains parity check values restricted to the lower triangular region, according to an embodiment of the present invention. As described previously, the encoder **203** (of FIG. **2**) can employ a simple encoding technique by restricting the values of the lower triangular area of the parity check matrix. According to an embodiment of the present invention, the restriction imposed on the parity check matrix is of the form:

$$H_{(n-k)xn}=[A_{(n-k)xk}B_{(n-k)x(n-k)}],$$

where B is lower triangular.

Any information block $i=(i_0, i_1, \ldots i_{k-1})$ is encoded to a codeword $c=(i_0, i_1, \ldots, i_{k-1}, p_0, p_1, \ldots, p_{n-k-1})$ using $Hc^T=0$, and recursively solving for parity bits; for example,

$$a_{00}i_0+a_{01}i_1+\ldots+a_{0,k-1}i_{k-1}+p_0=0 \Rightarrow \text{Solve } p_0$$

$$a_{10}i_0+a_{11}i_1+\ldots+a_{1,k-1}i_{k-1}+b_{10}p_0+p_1=0 \Rightarrow \text{Solve } p_1$$

and similarly for $p_2, p_3, \ldots, p_{n-k-1}$.

FIG. **7** is a graph showing performance between codes utilizing unrestricted parity check matrix (H matrix) versus restricted H matrix of FIG. **6**. The graph shows the performance comparison between two LDPC codes: one with a general parity check matrix and the other with a parity check matrix restricted to be lower triangular to simplify encoding. The modulation scheme, for this simulation, is 8-PSK. The performance loss is within 0.1 dB. Therefore, the performance loss is negligible based on the restriction of the lower triangular H matrices, while the gain in simplicity of the encoding technique is significant. Accordingly, any parity check matrix that is equivalent to a lower triangular or upper triangular under row and/or column permutation can be utilized for the same purpose.

FIGS. **8**A and **8**B are, respectively, a diagram of a non-Gray 8-PSK modulation scheme, and a Gray 8-PSK modulation, each of which can be used in the system of FIG. **1**. The non-Gray 8-PSK scheme of FIG. **8**A can be utilized in the receiver of FIG. **3** to provide a system that requires very low Frame Erasure Rate (FER). This requirement can also be satisfied by using a Gray 8-PSK scheme, as shown in FIG. **8**B, in conjunction with an outer code, such as Bose, Chaudhuri, and Hocquenghem (BCH), Hamming, or Reed-Solomon (RS) code.

Under this scheme, there is no need to iterate between the LDPC decoder **305** (FIG. **3**) and the bit metric generator **307**, which may employ 8-PSK modulation. In the absence of an outer code, the LDPC decoder **305** using Gray labeling exhibit an earlier error floor, as shown in FIG. **9** below.

FIG. **9** is a graph showing performance between codes utilizing Gray labeling versus non-Gray labeling of FIGS. **8**A and **8**B. The error floor stems from the fact that assuming correct feedback from LDPC decoder **305**, regeneration of 8-PSK bit metrics is more accurate with non-Gray labeling since the two 8-PSK symbols with known two bits are further apart with non-Gray labeling. This can be equivalently seen as operating at higher Signal-to-Noise Ratio (SNR). Therefore, even though error asymptotes of the same LDPC code using Gray or non-Gray labeling have the same slope (i.e., parallel to each other), the one with non-Gray labeling passes through lower FER at any SNR.

On the other hand, for systems that do not require very low FER, Gray labeling without any iteration between LDPC decoder **305** and 8-PSK bit metric generator **307** may be more suitable because re-generating 8-PSK bit metrics before every LDPC decoder iteration causes additional complexity. Moreover, when Gray labeling is used, re-generating 8-PSK bit metrics before every LDPC decoder iteration yields only very slight performance improvement. As mentioned previously, Gray labeling without iteration may be used for systems that require very low FER, provided an outer code is implemented.

The choice between Gray labeling and non-Gray labeling depends also on the characteristics of the LDPC code. Typically, the higher bit or check node degrees, the better it is for Gray labeling, because for higher node degrees, the initial feedback from LDPC decoder **305** to 8-PSK (or similar higher order modulation) bit metric generator **307** deteriorates more with non-Gray labeling.

When 8-PSK (or similar higher order) modulation is utilized with a binary decoder, it is recognized that the three (or more) bits of a symbol are not received "equally noisy". For example with Gray 8-PSK labeling, the third bit of a symbol is considered more noisy to the decoder than the other two bits. Therefore, the LDPC code design does not assign a small number of edges to those bit nodes represented by "more noisy" third bits of 8-PSK symbol so that those bits are not penalized twice.

FIG. **10** is a flow chart of the operation of the LDPC decoder using non-Gray mapping, according to an embodiment of the present invention. Under this approach, the LDPC decoder and bit metric generator iterate one after the other. In this example, 8-PSK modulation is utilized; however, the same principles apply to other higher modulation schemes as well. Under this scenario, it is assumed that the demodulator **301** outputs a distance vector, d, denoting the distances between received noisy symbol points and 8-PSK symbol points to the bit metric generator **307**, whereby the vector components are as follows:

$$d_i = -\frac{E_s}{N_0}\{(r_x - s_{i,x})^2 + (r_y - s_{i,y})^2\} \quad i = 0, 1, \ldots 7.$$

The 8-PSK bit metric generator **307** communicates with the LDPC decoder **305** to exchange a priori probability information and a posteriori probability information, which respectively are represented as u, and a. That is, the vectors u and a respectively represent a priori and a posteriori probabilities of log likelihood ratios of coded bits.

The 8-PSK bit metric generator **307** generates the a priori likelihood ratios for each group of three bits as follows. First, extrinsic information on coded bits is obtained:

$$e_j = a_j - u_j \quad j=0,1,2.$$

Next, 8-PSK symbol probabilities, p_i i=0,1, . . . ,7, are determined.

$$*y_j = -f(0,e_j) \quad j=0,1,2.$$

where $f(a,b)=\max(a,b)+LUT_f(a,b)$ with $LUT_f(a,b)=\ln(1+e^{-|a-b|})$

$$*x_j = y_j + e_j \quad j=0,1,2$$

$$*p_0 = x_0 + x_1 + x_2$$

$$p_1 = x_0 + x_1 + y_2$$

$$p_2 = x_0 + y_1 + x_2$$

$$p_3 = x_0 + y_1 + y_2$$

$$p_4 = y_0 + x_1 + x_2$$

$$p_5 = y_0 + x_1 + y_2$$

$$p_6 = y_0 + y_1 + x_2$$

$$p_7 = y_0 + y_1 + y_2$$

Next, the bit metric generator **307** determines a priori log likelihood ratios of the coded bits as input to LDPC decoder **305**, as follows:

$$u_0 = f(d_0+p_0, d_1+p_1, d_2+p_2, d_3+p_3) - f(d_4+p_4, d_5+p_5, d_6p_6, d_7+p_7) - e_0$$

$$u_1 = f(d_0+p_0, d_1+p_1, d_4+p_4, d_5+p_5) - f(d_2+p_2, d_3+p_3, d_6+p_6, d_7+p_7) - e_1$$

$$u_2 = f(d_0+p_0, d_2+p_2, d_4+p_4, d_6+p_6) - f(d_1+p_1, d_3+p_3, d_5+p_5, d_7+p_7) - e_2$$

It is noted that the function $f(.)$ with more than two variables can be evaluated recursively; e.g. $f(a,b,c)=f(f(a,b),c)$.

The operation of the LDPC decoder **305** utilizing non-Gray mapping is now described. In step **1001**, the LDPC decoder **305** initializes log likelihood ratios of coded bits, v, before the first iteration according to the following (and as shown in FIG. **12**A):

$$v_{n \to k_i} = u_n, \quad n=0,1, \ldots, N-1, \; i=1,2, \ldots, \deg(\text{bit node } n)$$

Here, $v_{n \to k_i}$ denotes the message that goes from bit node n to its adjacent check node k_i, u_n denotes the demodulator output for the bit n and N is the codeword size.

In step **1003**, a check node, k, is updated, whereby the input v yields the output w. As seen in FIG. **12**B, the incoming messages to the check node k from its d_c adjacent bit nodes are denoted by $v_{n_1\to k}, v_{n_2\to k}, \ldots, v_{n_{dc}\to k}$. The goal is to compute the outgoing messages from the check node k back to d_c adjacent bit nodes. These messages are denoted by

$$w_{k\to n_1}, w_{k\to n_2}, \ldots, w_{k\to n_{dc}}, \text{ where}$$

$$w_{k\to n_i}=g(v_{n_1\to k}, v_{n_2\to k}, \ldots, v_{n_{i-1}\to k}, v_{n_{i+1}\to k}, \ldots, v_{n_{dc}\to k}).$$

The function g( ) is defined as follows:

$$g(a,b)=\text{sign}(a)\times\text{sign}(b)\times\{\min(|a|,|b|)\}+\text{LUT}_g(a,b),$$

where $\text{LUT}_g(a,b)=\ln(1+e^{-|a+b|})-\ln(1+e^{-|a-b|})$. Similar to function f, function g with more than two variables can be evaluated recursively.

Next, the decoder **305**, per step **1005**, outputs a posteriori probability information (FIG. **12**C), such that:

$$a_n = u_n + \sum_j w_{k_j\to n}.$$

Per step **1007**, it is determined whether all the parity check equations are satisfied. If these parity check equations are not satisfied, then the decoder **305**, as in step **1009**, re-derives 8-PSK bit metrics and channel input u_n. Next, the bit node is updated, as in step **1011**. As shown in FIG. **14**C, the incoming messages to the bit node n from its d_v adjacent check nodes are denoted by $w_{k_1\to n}, w_{k_2\to n}, \ldots, w_{k_{dv}\to n}$. The outgoing messages from the bit node n are computed back to d_v adjacent check nodes; such messages are denoted by $v_{n\to k_1}, v_{n\to k_2}, \ldots, v_{n\to k_{dv}}$, and computed as follows:

$$v_{n\to k_i} = u_n + \sum_{j\neq i} w_{k_j\to n}$$

In step **1013**, the decoder **305** outputs the hard decision (in the case that all parity check equations are satisfied):

$$\hat{c}_n = \begin{cases} 0, & a_n \geq 0 \\ 1, & a_n < 0 \end{cases} \text{ Stop if } H\hat{c}^T = 0$$

The above approach is appropriate when non-Gray labeling is utilized. However, when Gray labeling is implemented, the process of FIG. **11** is executed.

FIG. **11** is a flow chart of the operation of the LDPC decoder of FIG. **3** using Gray mapping, according to an embodiment of the present invention. When Gray labeling is used, bit metrics are advantageously generated only once before the LDPC decoder, as re-generating bit metrics after every LDPC decoder iteration may yield nominal performance improvement. As with steps **1001** and **1003** of FIG. **10**, initialization of the log likelihood ratios of coded bits, v, are performed, and the check node is updated, per steps **1101** and **1103**. Next, the bit node n is updated, as in step **1105**. Thereafter, the decoder outputs the a posteriori probability information (step **1107**). In step **1109**, a determination is made whether all of the parity check equations are satisfied; if so, the decoder outputs the hard decision (step **1111**). Otherwise, steps **1103**–**1107** are repeated.

FIG. **13**A is a flowchart of process for computing outgoing messages between the check nodes and the bit nodes using a forward-backward approach, according to an embodiment of the present invention. For a check node with d_c adjacent edges, the computation of $d_c(d_c-1)$ and numerous g(.,.) functions are performed. However, the forward-backward approach reduces the complexity of the computation to $3(d_c-2)$, in which d_c-1 variables are stored.

Referring to FIG. **12**B, the incoming messages to the check node k from d_c adjacent bit nodes are denoted by $v_{n_1\to k}, v_{n_2\to k}, \ldots, v_{n_{dc}\to k}$. It is desired that the outgoing messages are computed from the check node k back to d_c adjacent bit nodes; these outgoing messages are denoted by $w_{k\to n_1}, w_{k\to n_2}, \ldots, w_{k\to n_{dc}}$.

Under the forward-backward approach to computing these outgoing messages, forward variables, $f_1, f_2, \ldots, f_{dc}$, are defined as follows:

$$f_1=v_{1\to k}$$

$$f_2=g(f_1, v_{2\to k})$$

$$f_3=g(f_2, v_{3\to k})$$

$$\ldots$$

$$\ldots$$

$$f_{dc}=g(f_{dc-1}, v_{dc\to k})$$

In step **1301**, these forward variables are computed, and stored, per step **1303**.

Similarly, backward variables, $b_1, b_2, \ldots, b_{dc}$, are defined by the following:

$$b_{dc}=v_{dc\to k}$$

$$b_{dc-1}=g(b_{dc}, v_{dc-1\to k})$$

$$\ldots$$

$$\ldots$$

$$b_1=g(b_2, v_{1\to k})$$

In step **1305**, these backward variables are then computed. Thereafter, the outgoing messages are computed, as in step **1307**, based on the stored forward variables and the computed backward variables. The outgoing messages are computed as follows:

$$w_{k\to 1}=b_2$$

$$w_{k\to i}=g(f_{i-1}, b_{i+1}) \quad i=2,3,\ldots,d_c-1$$

$$w_{k\to dc}=f_{dc-1}$$

Under this approach, only the forward variables, $f_2, f_3, \ldots, f_{dc}$, are required to be stored. As the backward variables b_i are computed, the outgoing messages, $w_{k\to i}$, are simultaneously computed, thereby negating the need for storage of the backward variables.

The computation load can be further enhance by a parallel approach, as next discussed.

FIG. **13**B is a flowchart of process for computing outgoing messages between the check nodes and the bit nodes using a parallel approach, according to an embodiment of the present invention. For a check node k with inputs $v_{n_1 43}$ $k, v_{n_2\to k}, \ldots, v_{n_{dc}\to k}$ from d_c adjacent bit nodes, the following parameter is computed, as in step **1311**:

$$\gamma_k g(v_{n_1\to k}, v_{n_2\to k}, \ldots, v_{n_{dc}\to k}).$$

It is noted that the g(.,.) function can also be expressed as follows:

$$g(a,b)=\ln\frac{1+e^{a+b}}{e^a+e^b}.$$

Exploiting the recursive nature of the g(.,.) function, the following expression results:

$$\gamma_k = \ln\frac{1+e^{g(v_{n_1\to k},\ \ldots,v_{n_{i-1}\to k},v_{n_{i+1}\to k},\ \ldots,v_{n_{dc}\to k})+v_{n_i\to k}}}{e^{g(v_{n_1\to k},\ \ldots,v_{n_{i-1}\to k},v_{n_{i+1}\to k},\ \ldots,v_{n_{dc}\to k})}+e^{v_{n_i\to k}}} = \ln\frac{1+e^{w_{k\to n_i}+v_{n_i\to k}}}{e^{w_{k\to n_i}}+e^{v_{n_i\to k}}}$$

Accordingly, $w_{k\to n_i}$ can be solved in the following manner:

$$w_{k\to n_i} = \ln\frac{e^{v_{n_i\to k}+\gamma_k}-1}{e^{v_{n_i\to k}-\gamma_k}-1} - \gamma_k$$

The ln(.) term of the above equation can be obtained using a look-up table LUT_x that represents the function $\ln|e^x-1|$ (step **1313**). Unlike the other look-up tables LUT_f or LUT_g, the table LUT_x would likely requires as many entries as the number of quantization levels. Once γ_k is obtained, the calculation of $w_{k\to n_i}$ for all n_i can occur in parallel using the above equation, per step **1315**.

The computational latency of γ_k is advantageously $\log_2(d_c)$.

FIGS. **14**A–**14**C are graphs showing simulation results of LDPC codes generated in accordance with various embodiments of the present invention. In particular, FIGS. **14**A–**14**C show the performance of LDPC codes with higher order modulation and code rates of ¾ (QPSK, 1.485 bits/symbol), ⅔ (8-PSK, 1.980 bits/symbol), and ⅚ (8-PSK, 2.474 bits/symbol).

Two general approaches exist to realize the interconnections between check nodes and bit nodes: (1) a fully parallel approach, and (2) a partially parallel approach. In fully parallel architecture, all of the nodes and their interconnections are physically implemented. The advantage of this architecture is speed.

The fully parallel architecture, however, may involve greater complexity in realizing all of the nodes and their connections. Therefore with fully parallel architecture, a smaller block size may be required to reduce the complexity. In that case, for the same clock frequency, a proportional reduction in throughput and some degradation in FER versus Es/No performance may result.

The second approach to implementing LDPC codes is to physically realize only a subset of the total number of the nodes and use only these limited number of "physical" nodes to process all of the "functional" nodes of the code. Even though the LDPC decoder operations can be made extremely simple and can be performed in parallel, the further challenge in the design is how the communication is established between "randomly" distributed bit nodes and check nodes. The decoder **305** (of FIG. **3**), according to one embodiment of the present invention, addresses this problem by accessing memory in a structured way, as to realize a seemingly random code. This approach is explained with respect to FIGS. **15**A and **15**B.

FIGS. **15**A and **15**B are diagrams of the top edge and bottom edge, respectively, of memory organized to support structured access as to realize randomness in LDPC coding, according to an embodiment of the present invention. Structured access can be achieved without compromising the performance of a truly random code by focusing on the generation of the parity check matrix. In general, a parity check matrix can be specified by the connections of the check nodes with the bit nodes. For example, the bit nodes are divided into groups of 392 (392 is provided for the purposes of illustration). Additionally, assuming the check nodes connected to the first bit node of degree **3**, for instance, are numbered as a,b and c, then the check nodes connected to the second bit node are numbered as a+p, b+p and c+p, the check nodes connected to the third bit node are numbered as a+2p, b+2p and c+2p etc. For the next group of 392 bit nodes, the check nodes connected to the first bit node are different from a, b, c so that with a suitable choice of p, all the check nodes have the same degree. A random search is performed over the free constants such that the resulting LDPC code is cycle-4 and cycle-6 free.

The above arrangement facilitates memory access during check node and bit-node processing. The values of the edges in the bipartite graph can be stored in a storage medium, such as random access memory (RAM). It is noted that for a truly random LDPC code during check node and bit node processing, the values of the edges would need to be accessed one by one in a random fashion. However, such an access scheme would be too slow for a high data rate application. The RAM of FIGS. **15**A and **15**B are organized in a manner, whereby a large group of relevant edges can be fetched in one clock cycle; accordingly, these values are placed "together" in memory. It is observed that, in actuality, even with a truly random code, for a group of check nodes (and respectively bit nodes), the relevant edges can be placed next to one another in RAM, but then the relevant edges adjacent to a group of bit nodes (respectively check nodes) will be randomly scattered in RAM. Therefore, the "togetherness," under the present invention, stems from the design of the parity check matrices themselves. That is, the check matrix design ensures that the relevant edges for a group of bit nodes and check nodes are simultaneously placed together in RAM.

As seen in FIGS. **15**A and **15**B, each box contains the value of an edge, which is multiple bits (e.g., 6). Edge RAM, according to one embodiment of the present invention, is divided into two parts: top edge RAM (FIG. **15**A) and bottom edge RAM (FIG. **15**B). Bottom edge RAM contains the edges between bit nodes of degree **2**, for example, and check nodes. Top edge RAM contains the edges between bit nodes of degree greater than 2 and check nodes. Therefore, for every check node, 2 adjacent edges are stored in the bottom RAM, and the rest of the edges are stored in the top edge RAM.

Continuing with the above example, a group of 392 bit nodes and 392 check nodes are selected for processing at a time. For 392 check node processing, q consecutive rows are accessed from the top edge RAM, and 2 consecutive rows from the bottom edge RAM. In this instance, q+2 is the degree of each check node. For bit node processing, if the group of 392 bit nodes has degree **2**, their edges are located in 2 consecutive rows of the bottom edge RAM. If the bit nodes have degree d>2, their edges are located in some d rows of the top edge RAM. The address of these d rows can be stored in non-volatile memory, such as Read-Only Memory (ROM). The edges in one of the rows correspond to the first edges of 392 bit nodes, the edges in another row correspond to the second edges of 392 bit nodes, etc. Moreover for each row, the column index of the edge that belongs to the first bit node in the group of 392 can also be stored in ROM. The edges that correspond to the second, third, etc. bit nodes follow the starting column index in a "wrapped around" fashion. For example, if the j^{th} edge in the row belongs to the first bit node, then the (j+1)st edge belongs to the second bit node, (j+2)nd edge belongs to the third bit node, . . . , and (j−1)st edge belongs to the 392^{th} bit node.

With the above organization (shown in FIGS. **15**A and **15**B), speed of memory access is greatly enhanced during LDPC coding.

FIG. **16** illustrates a computer system **1600** upon which an embodiment according to the present invention can be implemented. The computer system **1600** includes a bus **1601** or other communication mechanism for communicating information, and a processor **1603** coupled to the bus **1601** for processing information. The computer system **1600** also includes main memory **1605**, such as a random access memory (RAM) or other dynamic storage device, coupled to the bus **1601** for storing information and instructions to be executed by the processor **1603**. Main memory **1605** can also be used for storing temporary variables or other intermediate information during execution of instructions to be executed by the processor **1603**. The computer system **1600** further includes a read only memory (ROM) **1607** or other static storage device coupled to the bus **1601** for storing static information and instructions for the processor **1603**. A storage device **1609**, such as a magnetic disk or optical disk, is additionally coupled to the bus **1601** for storing information and instructions.

The computer system **1600** may be coupled via the bus **1601** to a display **1611**, such as a cathode ray tube (CRT), liquid crystal display, active matrix display, or plasma display, for displaying information to a computer user. An input device **1613**, such as a keyboard including alphanumeric and other keys, is coupled to the bus **1601** for communicating information and command selections to the processor **1603**. Another type of user input device is cursor control **1615**, such as a mouse, a trackball, or cursor direction keys for communicating direction information and command selections to the processor **1603** and for controlling cursor movement on the display **1611**.

According to one embodiment of the invention, generation of LDPC codes is provided by the computer system **1600** in response to the processor **1603** executing an arrangement of instructions contained in main memory **1605**. Such instructions can be read into main memory **1605** from another computer-readable medium, such as the storage device **1609**. Execution of the arrangement of instructions contained in main memory **1605** causes the processor **1603** to perform the process steps described herein. One or more processors in a multi-processing arrangement may also be employed to execute the instructions contained in main memory **1605**. In alternative embodiments, hard-wired circuitry may be used in place of or in combination with software instructions to implement the embodiment of the present invention. Thus, embodiments of the present invention are not limited to any specific combination of hardware circuitry and software.

The computer system **1600** also includes a communication interface **1617** coupled to bus **1601**. The communication interface **1617** provides a two-way data communication coupling to a network link **1619** connected to a local network **1621**. For example, the communication interface **1617** may be a digital subscriber line (DSL) card or modem, an integrated services digital network (ISDN) card, a cable modem, or a telephone modem to provide a data communication connection to a corresponding type of telephone line. As another example, communication interface **1617** may be a local area network (LAN) card (e.g. for Ethernet™ or an Asynchronous Transfer Model (ATM) network) to provide a data communication connection to a compatible LAN. Wireless links can also be implemented. In any such implementation, communication interface **1617** sends and receives electrical, electromagnetic, or optical signals that carry digital data streams representing various types of information. Further, the communication interface **1617** can include peripheral interface devices, such as a Universal Serial Bus (USB) interface, a PCMCIA (Personal Computer Memory Card International Association) interface, etc.

The network link **1619** typically provides data communication through one or more networks to other data devices. For example, the network link **1619** may provide a connection through local network **1621** to a host computer **1623**, which has connectivity to a network **1625** (e.g. a wide area network (WAN) or the global packet data communication network now commonly referred to as the "Internet") or to data equipment operated by service provider. The local network **1621** and network **1625** both use electrical, electromagnetic, or optical signals to convey information and instructions. The signals through the various networks and the signals on network link **1619** and through communication interface **1617**, which communicate digital data with computer system **1600**, are exemplary forms of carrier waves bearing the information and instructions.

The computer system **1600** can send messages and receive data, including program code, through the network(s), network link **1619**, and communication interface **1617**. In the Internet example, a server (not shown) might transmit requested code belonging to an application program for implementing an embodiment of the present invention through the network **1625**, local network **1621** and communication interface **1617**. The processor **1603** may execute the transmitted code while being received and/or store the code in storage device **169**, or other non-volatile storage for later execution. In this manner, computer system **1600** may obtain application code in the form of a carrier wave.

The term "computer-readable medium" as used herein refers to any medium that participates in providing instructions to the processor **1603** for execution. Such a medium may take many forms, including but not limited to non-volatile media, volatile media, and transmission media. Non-volatile media include, for example, optical or magnetic disks, such as storage device **1609**. Volatile media include dynamic memory, such as main memory **1605**. Transmission media include coaxial cables, copper wire and fiber optics, including the wires that comprise bus **1601**. Transmission media can also take the form of acoustic, optical, or electromagnetic waves, such as those generated during radio frequency (RF) and infrared (IR) data communications. Common forms of computer-readable media include, for example, a floppy disk, a flexible disk, hard disk, magnetic tape, any other magnetic medium, a CD-ROM, CDRW, DVD, any other optical medium, punch cards, paper tape, optical mark sheets, any other physical medium with patterns of holes or other optically recognizable indicia, a RAM, a PROM, and EPROM, a FLASH-EPROM, any other memory chip or cartridge, a carrier wave, or any other medium from which a computer can read.

Various forms of computer-readable media may be involved in providing instructions to a processor for execution. For example, the instructions for carrying out at least part of the present invention may initially be borne on a magnetic disk of a remote computer. In such a scenario, the remote computer loads the instructions into main memory and sends the instructions over a telephone line using a modem. A modem of a local computer system receives the data on the telephone line and uses an infrared transmitter to convert the data to an infrared signal and transmit the infrared signal to a portable computing device, such as a personal digital assistance (PDA) and a laptop. An infrared detector on the portable computing device receives the information and instructions borne by the infrared signal and places the data on a bus. The bus conveys the data to main memory, from which a processor retrieves and executes the instructions. The instructions received by main memory may optionally be stored on storage device either before or after execution by processor.

Accordingly, the various embodiments of the present invention provide an approach for generating structured Low Density Parity Check (LDPC) codes, as to simplify the encoder and decoder. Structure of the LDPC codes is provided by restricting the parity check matrix to be lower triangular. Also, the approach can advantageously exploit the unequal error protecting capability of LDPC codes on transmitted bits to provide extra error protection to more vulnerable bits of high order modulation constellations (such as 8-PSK (Phase Shift Keying)). The decoding process involves iteratively regenerating signal constellation bit metrics into an LDPC decoder after each decoder iteration or several decoder iterations. The above approach advantageously yields reduced complexity without sacrificing performance.

While the present invention has been described in connection with a number of embodiments and implementations, the present invention is not so limited but covers various obvious modifications and equivalent arrangements, which fall within the purview of the appended claims.

What is claimed is:

1. A method for decoding low density parity check (LDPC) codes, the method comprising:

receiving a priori probability information based an distance vector information comprising information on distances between received noisy symbol points end symbol points of a signal constellation associated with the LDPC codes, wherein signal constellation metrics are determined based on the distance vector information;

transmitting a posteriori probability information based on the a priori probability information;

determining whether parity check equations associated with the LDPC codes are satisfied according to the a priori probability and the a posteriori probability information;

selectively regenerating the signal constellation bit metrics based on the determining step; and

outputting decoded messages based on the regenerated signal constellation bit metrics.

2. A method according to claim **1**, further comprising:

determining extrinsic information based on the a posteriori probability information and a priori probability information; and

outputting symbol probabilities associated with the signal constellation according to the extrinsic information.

3. A method according to claim **1**, wherein symbols of the signal constellation are Gray coded, whereby more vulnerable bits of Gray coded symbol constellation are assigned at least as many parity checks as less vulnerable bits of Gray coded symbol constellation.

4. A method according to claim **1**, further comprising:

storing information regarding bit nodes and check nodes of the LDPC codes in contiguous physical memory locations.

5. A method according to claim **1**, wherein the LDPC codes are encoded using a structured parity check matrix that imposes restrictions on a sub-matrix of the parity check matrix.

6. A method according to claims **1**, wherein the signal constellation includes one of 8-PSK((Phase Shift Keying), 16-QAM (Quadrature Amplitude Modulation), and QPSK (Quadrature Phase Shift Keying).

7. A computer-readable medium bearing instructions for decoding low density parity check (LDPC) codes, said instruction, being arranged, upon execution, to cause one or more processors to perform the method of claim **1**.

8. A system for decoding low density parity check (LDPC) codes, the system comprising:

means for receiving a priori probability information based on distance vector information comprising information on distances between received noisy symbol points and symbol points of a signal constellation associated with the LDPC codes, wherein signal constellation metrics are determined based on the distance vector information;

means for transmitting a posteriori probability information based on the a priori probability information;

means for determining whether parity check equations associated with the LDPC codes are satisfied according to the a priori probability and the a posteriori probability information;

means for selectively regenerating the signal constellation bit metrics based on the determination; and

means for outputting decoded messages based on the regenerated signal constellation bit metrics.

9. A system according to claim **8**, further comprising:

means for determining extrinsic information based on the a posteriori probability information and a priori probability information; and

means for outputting symbol probabilities associated with the signal constellation according to the extrinsic information.

10. A system according to claim **8**, wherein symbols of the signal constellation are Gray coded, whereby more vulnerable bits of Gray coded symbol constellation are assigned at least as many parity checks as less vulnerable bits of Gray coded symbol constellation.

11. A system according to claim **8**, further comprising:

means for storing information regarding bit nodes and check nodes of the LDPC codes in contiguous physical locations.

12. A system according to claim **8**, wherein the LDPC codes are encoded using a structured parity check matrix that imposes restrictions on a sub-matrix of the parity check matrix.

13. A system according to claims **8**, wherein the signal constellation includes one of 8-PSK (Phase Shift Keying), 16-QAM (Quadrature Amplitude Modulation), and QPSK (Quadrature Phase Shift Keying).

14. A receiver for decoding low density parity check (LDPC) codes, the receiver comprising:

a bit metric generator configured to generate a priori probability information based on distance vector information comprising information on distances between received noisy symbol points and symbol points of a signal constellation associated with the LDPC codes, wherein signal constellation metrics are determined based on the distance vector information; and

a decoder configured to output a posteriori probability information based on the a priori probability information received from the bit metric generator, wherein the decoder is further configured to determine whether parity check equations associated with the LDPC codes are satisfied according to the a priori probability and the a posteriori probability information, the decoder outputting decoded messages based on a regenerated signal constellation bit metrics if the parity check equations are not satisfied.

15. A receiver according to claim **14**, wherein the bit metric generator is further configured to determine extrinsic information based on the a posteriori probability information and a priori probability information, and to output symbol probabilities associated with the signal constellation according to the extrinsic information.

16. A receiver according to claim **14**, wherein symbols of the signal constellation are Gray coded, whereby more vulnerable bits of Gray coded symbol constellation are assigned at least as many parity checks as less vulnerable bits of Gray coded symbol constellation.

17. A receiver according to claim **14**, further comprising: memory configured to contiguously storing information regarding bit nodes and check nodes at the LDPC.

18. A receiver according to claim **14**, wherein the LDPC codes are encoded using a structured parity check matrix that imposes restrictions on a sub-matrix of the parity check matrix.

19. A receiver according to claims **14**, wherein the signal constellation includes one of 8-PSK (Phase Shift Keying), 16-QAM (Quadrature Amplitude Modulation), and QPSK (Quadrature Phase Shift Keying).

* * * * *