

US Patent & Trademark Office

Patent Public Search | Text View

United States Patent Application Publication
Kind Code
Publication Date
Inventor(s)

20250253976
A1
August 07, 2025
LIANG; Chulong et al.

CHANNEL CODING AND MODULATION

Abstract

The present application relates to methods, systems, and devices related to digital wireless communication, and more specifically, to techniques related to channel coding and modulation. In one exemplary aspect, a method for wireless communication is disclosed. The method includes performing multiple operations that include a first operation related to pre-channel coding and/or a second operation comprising attaching redundancy check bits in an order. The method can also include generating a transmission waveform by a first node using one or more bits for transmission to a second node over a wireless channel using information derived from the multiple operations.

Inventors: LIANG; Chulong (Shenzhen, CN), GUO; Qiujin (Shenzhen, CN), XU; Jin (Shenzhen, CN), XU; Jun (Shenzhen, CN), KANG; Jian (Shenzhen, CN), FU; Qiang (Shenzhen, CN)

Applicant: ZTE Corporation (Shenzhen, CN)

Family ID: 79555984

Appl. No.: 19/186358

Filed: April 22, 2025

Related U.S. Application Data

parent US continuation 17664189 20220519 parent-grant-document US 12308961 child US 19186358
parent WO continuation PCT/CN2020/102067 20200715 PENDING child US 17664189

Publication Classification

Int. Cl.: H04L1/00 (20060101); H04L27/18 (20060101)

U.S. Cl.:

CPC H04L1/0058 (20130101); H04L1/0071 (20130101); H04L27/18 (20130101);

Background/Summary

CROSS-REFERENCE TO RELATED APPLICATION [0001] This application is a continuation of U.S. patent application Ser. No. 17/664,189, filed May 19, 2022, which is a continuation and claims priority to International Application No. PCT/CN 2020/102067, filed on Jul. 15, 2020, the disclosure of which is hereby incorporated by reference herein in its entirety.

TECHNICAL FIELD

[0002] This patent document is directed generally to wireless communications.

BACKGROUND

[0003] Mobile communication technologies are moving the world toward an increasingly connected and networked society. The rapid growth of mobile communications and advances in technology have led to greater demand for capacity and connectivity. Other aspects, such as energy consumption, device cost, spectral efficiency, and latency are also important to meeting the needs of various communication scenarios. Various techniques, including new ways to provide higher quality of service, are being discussed.

SUMMARY

[0004] This document discloses methods, systems, and devices related to digital wireless communication, and more specifically, to techniques related to channel coding and modulation.

[0005] In one exemplary aspect, a method for wireless communication is disclosed. The method includes performing multiple operations that include a first operation related to pre-channel coding and/or a second operation comprising attaching redundancy check bits in an order. The method can also include generating a transmission waveform by a first node using one or more bits for transmission to a second node over a wireless channel using information derived from the multiple operations.

[0006] In another exemplary aspect, a wireless communications apparatus comprising a processor is disclosed. The processor is configured to implement a method described herein.

[0007] In yet another exemplary aspect, the various techniques described herein may be embodied as processor-executable code and stored on a computer-readable program medium.

Description

BRIEF DESCRIPTION OF THE DRAWINGS

[0009] FIG. 1A provides a first example graph depicting a set of constellation points.

[0010] FIG. 1B provides a second example graph depicting a set of constellation points.

[0011] FIG. 1C provides a third example graph depicting a set of constellation points.

[0012] FIG. 2A provides a first example for the pre-channel coding before a transport block CRC attachment.

[0013] FIG. 2B provides a second example for the pre-channel coding before a transport block CRC attachment.

[0014] FIG. 3A provides a first example for the pre-channel coding is after a transport block CRC attachment.

[0015] FIG. 3B provides a second example for the pre-channel coding is after a transport block CRC attachment.

[0016] FIG. 3C provides a third example for the pre-channel coding is after a transport block CRC attachment.

[0017] FIG. 3D provides a fourth example for the pre-channel coding is after a transport block CRC attachment.

[0018] FIG. 4A provides a first example for the pre-channel coding after a code block CRC attachment.

[0019] FIG. 4B provides a second example for the pre-channel coding after a code block CRC attachment.

[0020] FIG. 5 is a block diagram of an example method for a control channel and modulation.

[0021] FIG. 6 shows an example of a wireless communication system where techniques in accordance with one or more embodiments of the present technology can be applied.

[0022] FIG. 7 is a block diagram representation of a portion of a hardware platform.

DETAILED DESCRIPTION

[0023] The development of the new generation of wireless communication—5G New Radio (NR) communication—is a part of a continuous mobile broadband evolution process to meet the requirements of increasing network demand. NR will provide greater throughput to allow more users connected at the same time. Other aspects, such as energy consumption, device cost, spectral efficiency, and latency are also important to meeting the needs of various communication scenarios.

[0024] In the fifth generation (5G) wireless and new radio (NR) communication systems of 3GPP (3GPP 5G), a first node (such as a user equipment (UE) or a base station (BS)), can transmit data bits to a second node (such as a BS or a UE) by partitioning data bits into transport blocks (TBs) for channel coding and modulation. The number of data bits of the transport block may be referred to as a transport block size (TBS). The data bits of a transport block may also be referred to as the payload. Channel coding and modulation schemes may be used to encode the payload into a modulation symbol sequence. A modulation symbol sequence may be a sequence of modulation symbols output from a modulation mapper. A modulation mapper may include a high-order modulation including a labeling of a constellation with the number of constellation points larger than 1. A constellation may include a set of complex values and each complex value is a constellation point. A modulation mapper with a modulation order Q_m may take as input Q_m bits and output a modulation symbol chosen from the constellation. The modulation order may also be the modulation order of the constellation. There are various types of constellations, such as M-ary quadrature amplitude modulation (MQAM), quadrature phase shift keying (QPSK), M-ary phase shift keying (MPSK), M-ary amplitude shift keying (MASK), and M-ary amplitude phase shift keying (MAPSK), as non-limiting examples.

[0025] A labeling may include a mapping between a set of bit sequences and a set of complex values. The bits associated to a complex value may be called the label of the complex value. The number of bits in the label of the complex value may be called the length of the label or the length of the labeling. A labeling of a constellation may be a mapping between a set of bit sequences and the set of constellation points. The modulation order of a constellation may also be the length of the labeling of the constellation. Table 1 may provide specific examples for labeling, for example, 00 is the label of the value $1+j$ in the labeling 1 and 1 is the label of the value -1 in the labeling 4. For a given constellation, there can be various types of labeling of the given constellation, such as Gray labeling and natural labeling, as non-limiting examples. Table 1 may provide two specific examples for the labeling of the set $\{\alpha, 3\alpha, 5\alpha, 7\alpha\}$, i.e., the labeling 2 and the labeling 3, where α is a positive real constant.

[0026] There may be a symbol set associated to a given modulation mapper. A symbol set may be a set of non-negative real values. An element in a symbol set may be called a symbol. For a modulation mapper with an MQAM constellation, the symbol set may contain all possible absolute values of the real part or the imaginary part of the constellation points in the MQAM constellation. The symbol set for an MQAM constellation with modulation order Q_m may contain all odd integers between 0 and $2 \cdot \sup Q_m/2$ multiplying a positive real constant α . For example, the symbol set for 16QAM is the set $\{\alpha, 3\alpha\}$; the symbol set for 64QAM is the set $\{\alpha, 3\alpha, 5\alpha, 7\alpha\}$; the symbol set for 256QAM is the set $\{\alpha, 3\alpha, 5\alpha, 7\alpha, 9\alpha, 11\alpha, 13\alpha, 15\alpha\}$; the symbol set for 1024QAM is the set $\{\alpha, 3\alpha, 5\alpha, 7\alpha, 9\alpha, 11\alpha, 13\alpha, 15\alpha, 17\alpha, 19\alpha, 21\alpha, 23\alpha, 25\alpha, 27\alpha, 29\alpha, 31\alpha\}$, where α is a positive real constant. For a modulation mapper with an MAPSK constellation, the symbol set may contain all possible complex modulus of constellation points in the MAPSK constellation. For example, the symbol set for the 16APSK constellation in Table 3(b) is $\{\alpha_{\text{sub}.0}, \alpha_{\text{sub}.1}, \alpha_{\text{sub}.2}, \alpha_{\text{sub}.3}\}$, where $0 < \alpha_{\text{sub}.0} < \alpha_{\text{sub}.1} < \alpha_{\text{sub}.2} < \alpha_{\text{sub}.3}$ are any positive real values; the symbol set for the 16APSK constellation in Table 3(c) is $\{\beta_{\text{sub}.0}, \beta_{\text{sub}.1}, \beta_{\text{sub}.2}, \beta_{\text{sub}.3}\}$, where $0 < \beta_{\text{sub}.0} < \beta_{\text{sub}.1} < \beta_{\text{sub}.2} < \beta_{\text{sub}.3}$ are any positive real values.

[0027] The definition of labeling may also be applied to symbol sets. This may not be limited to one type of labeling of a given symbol set. Table 1 may provide two specific examples for labeling of the symbol set $\{\alpha, 3\alpha, 5\alpha, 7\alpha\}$, i.e., the labeling 2 and the labeling 3. The label of a symbol in a constellation point may include a sub bit sequence of the label of the constellation point. Specifically, for MQAM constellations with modulation order Q_m , $Q_m/2-1$ bits among the Q_m bits of the label of a constellation point may be the label of the symbol of the real part of the constellation point and other $Q_m/2-1$ bits among the Q_m bits of the label of a constellation point is the label of the symbol of the imaginary part of the constellation point, where $Q_m \geq 4$. For MAPSK constellations with modulation order Q_m , Q_a bits among the Q_m bits of the label of a constellation point may be the label of the symbol of the constellation point, where $0 \leq Q_a < Q_m$. Table 2 may provide the symbol set as well as the corresponding labeling for 16QAM, 64QAM, and 256QAM of 3GPP 5G. Tables 3A-3C may provide examples for the relation between symbols, constellation points, the labeling of the symbol set and the labeling of the constellation with input bits ($b_{\text{sub}.0}, b_{\text{sub}.1}, \dots, b_{\text{sub}.Q_m-1}$) for a modulation mapper with a modulation order Q_m . FIGS. 1A-1C provide example graphs 100a-c depicting constellation points for Tables 3A-3C, respectively.

[0028] A symbol sequence may include a sequence with each element chosen from a symbol set. The energy of a symbol sequence may be defined as the sum of the weight of each symbol in the symbol sequence. The weight of a symbol e may include the function value $f(e)$ with $f(\cdot)$ being a monotonic increasing real-value function that maps non-negative real value into non-negative real value. Specific examples for $f(e)$ are $f(e)=e$, $f(e)=e \cdot \sup$, and $f(e)=2e+1$. The power of a modulation symbol may include the square of the complex modulus of the modulation symbol. The power of a modulation symbol sequence may include the sum of the power of each modulation symbol in the modulation symbol sequence.

[0029] Theoretically, a channel coding and modulation scheme outputs modulation symbols chosen from a constellation with equal probability may not be efficient. An efficient scheme can output different constellation points with different probabilities. Specifically, a constellation point with smaller power may appear more frequently than a constellation point with larger power in the output of the channel coding and modulation scheme.

[0030] In the present embodiments, channel coding and modulation schemes can encode a payload into a modulation symbol sequence with an unequal probability constellation points.

[0031] In the following, the size of a set, or the set size, may include the number of elements in the set. The notation L_z^L for real values may stand

for $\lceil z \rceil$ for real values may stand for the nearest integer to z . The notation $\exp(\cdot)$ may include the natural exponential function. The notation “mod” may include the modulo operation. The notation π is the Pi.

TABLE-US-00001 TABLE 1 Examples for labeling ($j = \{\text{square root over } (-1)\}$) is the imaginary unit) labeling 2 labeling 3 labeling 1 com- labeling 4 com- bit plex bit plex bit com- bit plex se- value se- plex se- value quence ($\alpha > 0$) quence ($\alpha > 0$) quence value quence $1 + j$ 00 α 11 α 01 $-1 - j$ $1 - j$ 01 3α 10 3α 00 $-1 + j$ 10 5α 00 5α 10 $1 - j$ 11 7α 01 7α 11

TABLE-US-00002 TABLE 2 symbol sets and labeling for MQAM constellation with input bits (b.sub.0, b.sub.1, ..., b.sub.Qm-1), Qm is the modulation order. symbol in symbol set 16QAM [00001]($\alpha = \frac{1}{\sqrt{10}}$) 64QAM [00002]($\alpha = \frac{1}{\sqrt{32}}$) Qm = 6 [00003]256QAM($\alpha = \frac{1}{\sqrt{170}}$) Qm = 8 for real part Qm = 4 label of symbol label of symbol or imaginary label of symbol (b.sub.2, b.sub.4) or (b.sub.2, b.sub.4, b.sub.6) or part (b.sub.2) or (b.sub.3) (b.sub.3, b.sub.5) (b.sub.3, b.sub.5, b.sub.7) α 0 01 011 3α 1 00 010 5α N/A 10 000 7α N/A 11 001 9α N/A N/A 101 11α N/A N/A 100 13α N/A N/A 110 15α N/A N/A 111

TABLE-US-00003 TABLE 3(a) Examples for the relation between symbols, constellation points, the labeling of the symbol set and the labeling of the 16QAM constellation with modulation order Qm = 4 for input bits (b.sub.0, b.sub.1, b.sub.2, b.sub.3) ($j = \{\text{square root over } (-1)\}$ is the imaginary unit) label of constellation point symbol (b.sub.0, b.sub.1, b.sub.2, b.sub.3) with label of symbol for the the symbol in the real part constellation the real imaginary of a constellation point being point in the part in the part in (b.sub.0) and label of the symbol constellation symbol set symbol set in the imaginary part of a (x) {+1, +3} {+1, +3} constellation point being (b.sub.1) $1 + j$ +1 +1 1100 $1 - j$ +1 +1 1101 $-1 + j$ +1 +1 1110 $-1 - j$ +1 +1 1111 $1 + 3j$ +1 +3 1000 $1 - 3j$ +1 +3 1001 $-1 + 3j$ +1 +3 1010 $-1 - 3j$ +1 +3 1011 $3 + j$ +3 +1 0100 $3 - j$ +3 +1 0101 $-3 + j$ +3 +1 0110 $-3 - j$ +3 +1 0111 $3 + 3j$ +3 +3 0000 $3 - 3j$ +3 +3 0001 $-3 + 3j$ +3 +3 0010 $-3 - 3j$ +3 +3 0011

TABLE-US-00004 TABLE 3(b) Example for relation between symbols, constellation points, the labeling of the symbol set and the labeling of a 16APSK with modulation order Qm = 4 for input bits (b.sub.0, b.sub.1, b.sub.2, b.sub.3) with Qa = 2 ($j = \{\text{square root over } (-1)\}$ imaginary unit) symbol in the constellation symbol set label label of point in the { α .sub.0, α .sub.1, α .sub.2, α .sub.3} of constellation constellation with symbol point (x) $0 < \alpha$.sub.0 $< \alpha$.sub.1 $< \alpha$.sub.2 $< \alpha$.sub.3 (b.sub.0, b.sub.1) (b.sub.0, b.sub.1, b.sub.2, b.sub.3) α .sub.0 .Math. exp(j .Math. $\pi/4$) α .sub.0 11 1100 α .sub.0 .Math. exp(j .Math. $3\pi/4$) α .sub.0 11 1101 α .sub.0 .Math. exp(j .Math. $5\pi/4$) α .sub.0 11 1111 α .sub.0 .Math. exp(j .Math. $7\pi/4$) α .sub.0 11 1110 α .sub.1 .Math. exp(j .Math. $\pi/4$) α .sub.1 10 1000 α .sub.1 .Math. exp(j .Math. $3\pi/4$) α .sub.1 10 1001 α .sub.1 .Math. exp(j .Math. $5\pi/4$) α .sub.1 10 1011 α .sub.1 .Math. exp(j .Math. $7\pi/4$) α .sub.1 10 1010 α .sub.2 .Math. exp(j .Math. $\pi/4$) α .sub.2 00 0000 α .sub.2 .Math. exp(j .Math. $3\pi/4$) α .sub.2 00 0001 α .sub.2 .Math. exp(j .Math. $5\pi/4$) α .sub.2 00 0011 α .sub.2 .Math. exp(j .Math. $7\pi/4$) α .sub.2 00 0010 α .sub.3 .Math. exp(j .Math. $\pi/4$) α .sub.3 01 0100 α .sub.3 .Math. exp(j .Math. $3\pi/4$) α .sub.3 01 0101 α .sub.3 .Math. exp(j .Math. $5\pi/4$) α .sub.3 01 0111 α .sub.3 .Math. exp(j .Math. $7\pi/4$) α .sub.3 01 0110

TABLE-US-00005 TABLE 3(c) Examples for the relation symbols, constellation points, the labeling of the symbol set and the labeling of a 16APSK constellation with modulation order Qm = 4 for input bits (b.sub.0, b.sub.1, b.sub.2, b.sub.3) with Qa = 2 ($j = \{\text{square root over } (-1)\}$ is the imaginary unit) symbol in the constellation symbol set label label of point in the { β .sub.0, β .sub.1, β .sub.2, β .sub.3} of constellation constellation (x) with symbol point with $\Delta = \pi/4$ $0 < \beta$.sub.0 $< \beta$.sub.1 $< \beta$.sub.2 $< \beta$.sub.3 (b.sub.0, b.sub.1) (b.sub.0, b.sub.1, b.sub.2, b.sub.3) β .sub.0 .Math. exp(j .Math. $\pi/4$) β .sub.0 11 1100 β .sub.0 .Math. exp(j .Math. $3\pi/4$) β .sub.0 11 1101 β .sub.0 .Math. exp(j .Math. $5\pi/4$) β .sub.0 11 1111 β .sub.0 .Math. exp(j .Math. $7\pi/4$) β .sub.0 11 1110 β .sub.1 .Math. exp(j .Math. $\pi/4 + j$.Math. Δ) β .sub.1 10 1000 β .sub.1 .Math. exp(j .Math. $3\pi/4 + j$.Math. Δ) β .sub.1 10 1001 β .sub.1 .Math. exp(j .Math. $5\pi/4 + j$.Math. Δ) β .sub.1 10 1011 β .sub.1 .Math. exp(j .Math. $7\pi/4 + j$.Math. Δ) β .sub.1 10 1010 β .sub.2 .Math. exp(j .Math. $\pi/4$) β .sub.2 00 0000 β .sub.2 .Math. exp(j .Math. $3\pi/4$) β .sub.2 00 0001 β .sub.2 .Math. exp(j .Math. $5\pi/4$) β .sub.2 00 0011 β .sub.2 .Math. exp(j .Math. $7\pi/4$) β .sub.2 00 0010 β .sub.3 .Math. exp(j .Math. $\pi/4 + j$.Math. Δ) β .sub.3 01 0100 β .sub.3 .Math. exp(j .Math. $3\pi/4 + j$.Math. Δ) β .sub.3 01 0101 β .sub.3 .Math. exp(j .Math. $5\pi/4 + j$.Math. Δ) β .sub.3 01 0111 β .sub.3 .Math. exp(j .Math. $7\pi/4 + j$.Math. Δ) β .sub.3 01 0110

[0032] High-order M-ary quadrature amplitude modulation (MQAM) constellation may be used to improve spectral efficiency in the 3rd Generation Partnership Project (3GPP). However, because of the non-linearity of transmission medium, the marginal benefits obtained in higher order MQAM constellations may be gradually disappearing. New channel coding and modulation techniques may need be used to further improve efficiency for higher order MQAM. For improving efficiency for higher order MQAM, channel coding and modulation schemes are provided in the present embodiments.

System Overview

[0033] The present embodiments relate to channel coding and modulation. Tables 4A-C provide examples for labeling a symbol set.

TABLE-US-00006 TABLE 4A Example for labeling of a symbol set with the symbol set $\Phi = \{1, 2, 6\}$ Labeling 1 Labeling 2 Labeling 3 Labeling 4 Labeling 5 Labeling 6 Labeling 7 Labeling 8 Symbol (Qa = 2) (Qa = 2) (Qa = 2) (Qa = 2) (Qa = 3) (Qa = 3) (Qa = 3) (Qa = 3) in Φ (b.sub.0, b.sub.1) (b.sub.0, b.sub.1) (b.sub.0, b.sub.1) (b.sub.0, b.sub.1) (b.sub.0, b.sub.1, b.sub.2) (b.sub.0, b.sub.1, b.sub.2) (b.sub.0, b.sub.1, b.sub.2) (b.sub.0, b.sub.1, b.sub.2) $1 + j$ 00 α 11 α 01 $-1 - j$ $1 - j$ 01 3α 10 3α 00 $-1 + j$ 10 5α 00 5α 10 $1 - j$ 11 7α 01 7α 11

TABLE-US-00007 TABLE 4B Example for labeling of a symbol set with the symbol set $\Phi = \{1, 3, 5, 7\}$ Labeling 1 Labeling 2 Labeling 3 Labeling 4 Labeling 5 Labeling 6 Labeling 7 Labeling 8 symbol (Qa = 2) (Qa = 2) (Qa = 2) (Qa = 2) (Qa = 3) (Qa = 3) (Qa = 3) (Qa = 3) in Φ (b.sub.0, b.sub.1) (b.sub.0, b.sub.1) (b.sub.0, b.sub.1) (b.sub.0, b.sub.1) (b.sub.0, b.sub.1, b.sub.2) (b.sub.0, b.sub.1, b.sub.2) (b.sub.0, b.sub.1, b.sub.2) (b.sub.0, b.sub.1, b.sub.2) $1 + j$ 00 α 11 α 01 $-1 - j$ $1 - j$ 01 3α 10 3α 00 $-1 + j$ 10 5α 00 5α 10 $1 - j$ 11 7α 01 7α 11

TABLE-US-00008 TABLE 4C Example for labeling of a symbol set with the symbol set $\Phi = \{1, 3, 5, 7, 9, 11, 13, 15\}$ Labeling 1 Labeling 2 Labeling 3 Labeling 4 Labeling 5 Labeling 6 Labeling 7 Labeling 8 symbol (Qa = 3) (Qa = 3) (Qa = 3) (Qa = 3) (Qa = 3) (Qa = 3) (Qa = 3) (Qa = 3) in Φ (b.sub.0, b.sub.1, b.sub.2) (b.sub.0, b.sub.1, b.sub.2) (b.sub.0, b.sub.1, b.sub.2) (b.sub.0, b.sub.1, b.sub.2) (b.sub.0, b.sub.1, b.sub.2) (b.sub.0, b.sub.1, b.sub.2) (b.sub.0, b.sub.1, b.sub.2) (b.sub.0, b.sub.1, b.sub.2) $1 + j$ 00 α 11 α 01 $-1 - j$ $1 - j$ 01 3α 10 3α 00 $-1 + j$ 10 5α 00 5α 10 $1 - j$ 11 7α 01 7α 11

Example Embodiment 1

[0034] Example Embodiment 1 can provide a detailed description of a pre-channel coding. In this embodiment, channel coding and modulation for a transport block comprises the following information: channel coding information, pre-channel coding information, transport block CRC attachment information, code block segmentation information, code block CRC attachment information, etc. The channel coding can include a low-density parity-check coding, a polar coding, a turbo coding, a convolutional coding, as non-limited examples. The pre-channel coding may include a process comprising any of a bit-to-symbol encoding and a symbol-to-bit conversion.

[0035] In a specific example of this embodiment, the bit-to-symbol encoding can encode a first bit sequence u into a first symbol sequence A using a symbol set Φ . Denote the length of the first bit sequence u as K_d . Denote the length of the first symbol sequence A as N_a . Denote the set size of the symbol set Φ as M_a . The length of the labeling of the symbol set Φ , denoted as Q_a , may be an integer such that $2^{\text{sup}}.Q_a \geq M_a$.

[0036] Further, in a specific example of this embodiment, the symbol-to-bit conversion converts the first symbol sequence $A = [A_{\text{sub}.0}, A_{\text{sub}.1}, A_{\text{sub}.2}, \dots, A_{\text{sub}.N_a-1}]$ into a second bit sequence $v = [v_{\text{sub}.0}, v_{\text{sub}.1}, v_{\text{sub}.2}, \dots, v_{\text{sub}.N_v-1}]$ according to a labeling of the symbol set Φ by converting each symbol $A_{\text{sub}.i}$ into bits equal to the length of the labeling of the symbol set Φ . One of the specific implementation of the symbol-to-bit conversion is that the bits ($V_{\text{sub}.i} \cdot \text{Math}.Q_a, V_{\text{sub}.i} \cdot \text{Math}.Q_a + 1, \dots, V_{\text{sub}.i} \cdot \text{Math}.Q_a + Q_a - 1$) of the second bit sequence v is the label of the symbol $A_{\text{sub}.i}$ for $i = 0, 1, 2, \dots, N_a - 1$.

[0037] Further, in a specific example of this embodiment, Table 2 may provide one of the specific implementation of the labeling of the symbol set Φ

for 16QAM, 64QAM and 256QAM in the 3GPP 5G.

[0038] Further, in an example of this embodiment, one of the specific implementation of the labeling of the symbol set Φ is that the label of the smallest symbol in the symbol set Φ has bit "1" more than the label of the largest symbol in the symbol set Φ . Table 4B gives a specific example that the symbol "1" is the smallest symbol with the label "11" having 2 bit "1" while the largest symbol "7" with the label "10" having 1 bit "1" in the labeling 2. Table 4A-C gives many specific examples of the label of symbol for three symbol sets {1,2, 6}, {1,3,5,7}, {1,3,5,7,9,11,13,15} with $Q_a=2$ and 3.

[0039] Further, in an example of this embodiment, Table 4C can provide specific implementations of the labeling of the symbol set Φ is that the label of the smallest symbol in the symbol set Φ is the all bit 1's sequence.

[0040] Further, in a specific example of this embodiment, Table 3 can provide one of specific implementations of the labeling of the symbol set Φ that the labeling of the associated constellation of the modulation mapper have the following property: the label with all bits being "1" is associated to the constellation point with the smallest power. the constellation point $-1-j$ with the smallest power in Table 3A is associated to the label "1111", the constellation point $\alpha \cdot \text{sub.0} \cdot \text{Math.exp}(j \cdot \text{Math.5}\pi/4)$ with the smallest power in Table 3B is associated to the label "1111", the constellation point $\beta \cdot \text{sub.0} \cdot \text{Math.exp}(j \cdot \text{Math.5}\pi/4)$ with the smallest power in Table 3C is associated to the label "1111", where $j=\sqrt{-1}$ is the imaginary unit.

[0041] Further, in a specific example of this embodiment, the bit-to-symbol encoding may include any of source coding related encoding, energy threshold encoding, minimum energy encoding, variable-length encoding, non-linear coding, etc.

[0042] Further, in a specific example of this embodiment, the source coding related encoding can encode the first bit sequence u into the first symbol sequence A according to a probability related parameter.

[0043] One of the specific implementations for the probability related parameter may include a probability mass function (PMF) of the symbol set Φ . One of the specific implementations for using the probability mass function of the symbol set $\Phi=\{\phi \cdot \text{sub.0}, \phi \cdot \text{sub.1}, \dots, \phi \cdot \text{sub.Ma-1}\}$ may be to calculate the number of symbol ϕ_i appears in the first symbol sequence A as

$$[00004] \text{Num}(\phi_i) = \text{Math. Na} \cdot \text{Math. } \sum_{j=0}^i \text{Pr}\{\phi_j\} \cdot \text{Math. } - \text{Math. Na} \cdot \text{Math. } \sum_{j=0}^{i-1} \text{Pr}\{\phi_j\} \cdot \text{Math. },$$

[0044] where the summation in the right may be zero for $i=0$. Take $\Phi=\{1, 3, 5\}$ with the probability mass function $\text{Pr}\{1\}=0.4, \text{Pr}\{3\}=0.3, \text{Pr}\{5\}=0.3$ and $\text{Na}=5$ as an example as follows.

$$[00005] \text{Num}(1) = \text{Math. 5} \cdot \text{Math. Pr}\{1\} \cdot \text{Math. } - \text{Math. 5} \cdot \text{Math. 0.4} \cdot \text{Math. } = 2,$$

$$\text{Num}(2) = \text{Math. 5} \cdot \text{Math. Pr}\{1\} + 5 \cdot \text{Math. Pr}\{3\} \cdot \text{Math. } - \text{Math. 5} \cdot \text{Math. Pr}\{1\} \cdot \text{Math. } = \text{Math. } 5 \times 0.4 + 5 \times 0.3 \cdot \text{Math. } - 2 = \text{Math. 3.5} \cdot \text{Math. } - \text{Math. 2} = \text{Math. 1.5},$$

$$\text{Num}(3) = \text{Math. 5} \cdot \text{Math. Pr}\{1\} + 5 \cdot \text{Math. Pr}\{3\} + 5 \cdot \text{Math. Pr}\{5\} \cdot \text{Math. } - \text{Math. 5} \cdot \text{Math. Pr}\{1\} + 5 \cdot \text{Math. Pr}\{3\} \cdot \text{Math. } = \text{Math. } 5 \times 0.4 + 5 \times 0.3 + 5 \times 0.3 - \text{Math. 3.5} = \text{Math. 2.5},$$

[0045] One of the specific implementations for using the probability mass function of the symbol set $\Phi=\{\phi \cdot \text{sub.0}, \phi \cdot \text{sub.1}, \dots, \phi \cdot \text{sub.Ma-1}\}$ may be to calculate the number of symbol $\phi \cdot \text{sub.i}$ appears in the first symbol sequence A

$$[00006] \text{Num}(\phi_i) = \text{round}(\text{Na} \cdot \text{Math. } \sum_{j=0}^i \text{Pr}\{\phi_j\}) - \text{round}(\text{Na} \cdot \text{Math. } \sum_{j=0}^{i-1} \text{Pr}\{\phi_j\}),$$

[0046] where the summation in the right may be zero for $i=0$. Take $\Phi=\{1, 3, 5\}$ with the probability mass function $\text{Pr}\{1\}=0.4, \text{Pr}\{3\}=0.3, \text{Pr}\{5\}=0.3$ and $\text{Na}=5$ as an example as follows.

$$[00007] \text{Num}(1) = \text{round}(5 \cdot \text{Math. Pr}\{1\}) = \text{round}(5 \cdot \text{Math. 0.4}) = 2,$$

$$\text{Num}(3) = \text{round}(5 \cdot \text{Math. Pr}\{1\} + 5 \cdot \text{Math. Pr}\{3\}) - \text{round}(5 \cdot \text{Math. Pr}\{1\}) = \text{round}(3.5) - 2 = 2,$$

$$\text{Num}(5) = \text{round}(5 \cdot \text{Math. Pr}\{1\} + 5 \cdot \text{Math. Pr}\{3\} + 5 \cdot \text{Math. Pr}\{5\}) - \text{round}(5 \cdot \text{Math. Pr}\{1\} + 5 \cdot \text{Math. Pr}\{3\}) = 5 - \text{round}(3.5) = 1.$$

[0047] The number of symbol $\phi \cdot \text{sub.i}$ appearing in the first symbol sequence A , the first symbol sequence A may have the number of 1, 3, and 5 being $\text{Num}(1), \text{Num}(3)$ and $\text{Num}(5)$. Using the $\text{round}(\cdot)$ operation, Table 5 may provide one of the specific example for $\Phi=\{1, 3, 5\}$ with the probability mass function $\text{Pr}\{1\}=0.4, \text{Pr}\{3\}=0.3, \text{Pr}\{5\}=0.3, \text{Kd}=4$ and $\text{Na}=5$.

TABLE-US-00009 TABLE 5 The source coding related encoding with a single PMF as parameters using the $\text{round}(\cdot)$ operation for $\text{Kd} = 4$ and $\text{Na} = 5$ number of the first bit The first PMF for the appearances sequence symbol sequence symbol set of $u = A = \Phi = \{1, 3, 5\}$ each symbol
[u.sub.0, u.sub.1, u.sub.2, u.sub.3] [A.sub.0, A.sub.1, A.sub.2, A.sub.3, A.sub.4] $\text{Pr}\{1\} = 0.4, \text{Num}(1) = 2, 0000 \ 11335 \ \text{Pr}\{3\} = 0.3, \text{Num}(3) = 2, 0001 \ 11353 \ \text{Pr}\{5\} = 0.3, \text{Num}(5) = 1. \ 0010 \ 13135 \ 0011 \ 13315 \ 0100 \ 13513 \ 0101 \ 15133 \ 0110 \ 15331 \ 0111 \ 31153 \ 1000 \ 31351 \ 1001 \ 31513 \ 1010 \ 33115 \ 1011 \ 33511 \ 1100 \ 35131 \ 1101 \ 51131 \ 1110 \ 51331 \ 1111 \ 53131$

[0048] One of the specific implementations for the probability related parameter may include Q_a probability mass functions for the set $\{0, 1\}$, where Q_a is the length of the labeling of the symbol set Φ . Denote the Q_a PMFs as $\text{PMF.sub.0}, \text{PMF.sub.1}, \dots, \text{PMF.sub.Qa-1}$. One of the specific implementations for using the Q_a PMFs is as follows.

[0049] Dividing the first bit sequence u into Q_a parts $u \cdot \text{sub.0}, u \cdot \text{sub.1}, \dots, u \cdot \text{sub.Qa-1}$. Regarding the set $\{0, 1\}$ as a symbol set and using PMF.sub.i to encode $u \cdot \text{sub.i}$ into a third bit sequence $y \cdot \text{sub.i}$ of Na bits as the examples in Table 6. Taking the i -th bit from each of the third bit sequences $y \cdot \text{sub.0}, y \cdot \text{sub.1}, \dots$, and $y \cdot \text{sub.Qa-1}$ to form a label $(y \cdot \text{sub.0}, i, y \cdot \text{sub.1}, i, \dots, y \cdot \text{sub.Qa-1}, i)$ and map to the symbol $A \cdot \text{sub.i}$ of the first symbol sequence A according to the labeling of the symbol set Φ such that $A \cdot \text{sub.i}$ has a label $(y \cdot \text{sub.0}, i, y \cdot \text{sub.1}, i, \dots, y \cdot \text{sub.Qa-1}, i)$, where $i=0, 1, 2, \dots, \text{Na}-1$. Using the $\text{round}(\cdot)$ operation, Table 6 may provide one of the specific example for $\Phi=\{1, 3, 5, 7\}$ using the labeling 3 in Table 4B, where other parameters are as follows.

$$[0050] Q_a=2, \text{Kd}=4, \text{Na}=5;$$

$$[0051] \text{The PMF.sub.0 is } \text{Pr}\{0\}=0.35, \text{Pr}\{1\}=0.65; \text{The PMF.sub.1 is } \text{Pr}\{0\}=0.2, \text{Pr}\{1\}=0.8;$$

[0052] The first bit sequence $u=[u \cdot \text{sub.0}, u \cdot \text{sub.1}, u \cdot \text{sub.2}, u \cdot \text{sub.3}]$ is divided into $Q_a=2$ parts with $u \cdot \text{sub.0}=[u \cdot \text{sub.0}, u \cdot \text{sub.1}, u \cdot \text{sub.2}]$ of length 3 bits and $u \cdot \text{sub.1}=[u \cdot \text{sub.3}]$ of length 1 bit.

$$[0053] \text{For } u \cdot \text{sub.0}, \text{Num}(0)=\text{round}(5 \times 0.35)=2, \text{Num}(1)=\text{round}(5 \times 0.35+5 \times 0.65)-2=3;$$

$$[0054] \text{For } u \cdot \text{sub.1}, \text{Num}(0)=\text{round}(5 \times 0.20)=1, \text{Num}(1)=\text{round}(5 \times 0.20+5 \times 0.80)-1=4.$$

TABLE-US-00010 TABLE 6 The source coding related encoding with two PMFs $\text{Pr}\{0\} = 0.35, \text{Pr}\{1\} = 0.65$ for $y \cdot \text{sub.0}$ and $\text{Pr}\{0\} = 0.2, \text{Pr}\{1\} = 0.8$ for $y \cdot \text{sub.1}$ for the set $\{0, 1\}$ as parameters. $\text{Kd} = 4, \text{Na} = 5$. $\text{Num}(0) = 2, \text{Num}(1) = 3$ for $y \cdot \text{sub.0}$; $\text{Num}(0) = 1, \text{Num}(1) = 4$ for $y \cdot \text{sub.1}$. The labeling of the symbol set $\Phi = \{1, 3, 5, 7\}$ is $\{11,10,01,00\}$. the first bit sequence $u = [u \cdot \text{sub.0}, u \cdot \text{sub.1}, u \cdot \text{sub.2}, u \cdot \text{sub.3}]$ [00008]

$$\begin{pmatrix} u_0 \\ u_1 \end{pmatrix} = \begin{pmatrix} u_0, u_1, u_2 \\ u_3 \end{pmatrix} \text{ sequences } [00009] \begin{pmatrix} y_0 \\ y_1 \end{pmatrix} = \begin{pmatrix} y_{0,0}, y_{0,1}, y_{0,2}, y_{0,3}, y_{0,4} \\ y_{1,0}, y_{1,1}, y_{1,2}, y_{1,3}, y_{1,4} \end{pmatrix} \text{ the first symbol sequence } A = [A \cdot \text{sub.0}, A \cdot \text{sub.1}, A \cdot \text{sub.2},$$

$$A \cdot \text{sub.3}, A \cdot \text{sub.4}] [00010] \begin{pmatrix} 000 \\ 0 \end{pmatrix} 00111; 11110 \ 55113 [00011] \begin{pmatrix} 000 \\ 1 \end{pmatrix} 00111; 11011 \ 55311 [00012] \begin{pmatrix} 001 \\ 0 \end{pmatrix} 01011; 11110 \ 51513 [00013] \begin{pmatrix} 001 \\ 1 \end{pmatrix}$$

$$01011; 11011 \ 51711 [00014] \begin{pmatrix} 010 \\ 0 \end{pmatrix} 01101; 11110 \ 51153 [00015] \begin{pmatrix} 010 \\ 1 \end{pmatrix} 01101; 11011 \ 51351 [00016] \begin{pmatrix} 011 \\ 0 \end{pmatrix} 01110; 11110 \ 51117 [00017] \begin{pmatrix} 011 \\ 1 \end{pmatrix}$$

$$01110; 11011 \ 51315 [00018] \begin{pmatrix} 100 \\ 0 \end{pmatrix} 10101; 11110 \ 15153 [00019] \begin{pmatrix} 100 \\ 1 \end{pmatrix} 10101; 11011 \ 15351 [00020] \begin{pmatrix} 101 \\ 0 \end{pmatrix} 10110; 11110 \ 15117 [00021] \begin{pmatrix} 101 \\ 1 \end{pmatrix}$$

$$10110; 11011 \ 15315 [00022] \begin{pmatrix} 110 \\ 0 \end{pmatrix} 11001; 11110 \ 11553 [00023] \begin{pmatrix} 110 \\ 1 \end{pmatrix} 11001; 11011 \ 11751 [00024] \begin{pmatrix} 111 \\ 0 \end{pmatrix} 11010; 11110 \ 11517 [00025] \begin{pmatrix} 111 \\ 1 \end{pmatrix}$$

$$11010; 11011 \ 11715$$

[0055] One of the specific implementations for the probability related parameter may be R positive integer sets with a one-to-one mapping to the symbol set and the sum of the elements in the non-negative integer set is equal to the length of the first symbol sequence A. One of the specific implementations may be that the non-negative integer set has elements being the number of each symbol $\phi_{\text{sub}.i}$ appearing in the first symbol sequence A. The encoding may be the same as Table 5.

[0056] One of the specific implementations for the probability related parameter may be R positive integer sets with $R > 1$. One of the specific implementations may be that the R positive integer sets are the number of 0's and 1's in each $y_{\text{sub}.i}$ of Table 6, where R equals to the length of the labeling of the symbol set Q_a and the encoding is the same as Table 6. One of the specific implementations may be that each of R positive integer sets is the set $\{\text{Num}(\phi_{\text{sub}.0}), \text{Num}(\phi_{\text{sub}.1}), \dots, \text{Num}(\phi_{\text{sub}.Ma-1})\}$ with encoding as follows.

[0057] Construct a one-to-one mapping between a prefix-free code of R codewords and the R positive integer sets. One of the specific implementations is $R=3$ and the prefix-free code may be $\{0, 10, 11\}$ and the R positive integer sets are $\{3,2\}$, $\{4,1\}$ and $\{1,4\}$ for the symbol set $\{1, 3\}$ with one-to-one mapping $0 \rightarrow \{3,2\}$; $10 \rightarrow \{4,1\}$; $11 \rightarrow \{1,4\}$.

[0058] Select the positive integer set that corresponding to the prefix bits of the first bit sequence u and encode as Table 5. For example, $u=[0,0,1,1]$, the prefix bit 0 is a codeword of the prefix-free code and the corresponding positive integer set $\{3,2\}$ is used as the parameter $\{\text{Num}(1), \text{Num}(3)\}$ for the encoding in Table 5. The rest bits in u as input, i.e., $[0,1,1]$. Another example is $u=[1,0,1,1]$, the prefix bits 10 is a codeword of the prefix-free code the corresponding positive integer set $\{4,1\}$ is used as the parameter $\{\text{Num}(1), \text{Num}(3)\}$ for the encoding with the rest bits in u as input, i.e., $[1,1]$. Table 7 gives the details for the encoding.

TABLE-US-00011 TABLE 7 The source coding related encoding with multiple positive integer sets as parameter for $K_d = 4$ and $N_a = 5$ for the symbol set $\Phi = \{1, 3\}$ Positive integer the first bit sequence The first set, i.e., number $u = [u_{\text{sub}.0}, u_{\text{sub}.1}, u_{\text{sub}.2}, u_{\text{sub}.3}]$, the bold symbol sequence of appearances of bit(s) is used for selecting A = each symbol the positive integer set $[A_{\text{sub}.0}, A_{\text{sub}.1}, A_{\text{sub}.2}, A_{\text{sub}.3}, A_{\text{sub}.4}]$ $\{3, 2\}$ with **0000** 11133 Num(1) = 3, **0001** 11313 Num(3) = 2. **0010** 11331 **0011** 13113 **0100** 13311 **0101** 31113 **0110** 31131 **0111** 31311 (4, 1) with **1000** 11113 Num(1) = 4, **1001** 11131 Num(3) = 1 **1010** 11311 **1011** 13111 (1, 4) with **1100** 13333 Num(1) = 1, **1101** 31333 Num(3) = 4. **1110** 33133 **1111** 33313

[0059] Further, in a specific example of this embodiment, the energy threshold encoding encodes the first bit sequence u into the first symbol sequence A according to a pre-defined value $E_{\text{sub}.m} > 0$ such that the sequence energy of any of the $2^{\text{sup}.K_d}$ possible output of the energy threshold encoding is less than or equal to $E_{\text{sub}.m}$. One of the specific implementation is discussed as follows.

[0060] The sequence energy may be defined as the square sum of the symbols in the first symbol sequence as example.

[0061] Among all the $M_a^{\text{sup}.N_a}$ symbol sequences of length N_a for the symbol set Φ , denote $T_{\text{sub}.i}(a_{\text{sub}.0}, a_{\text{sub}.1}, \dots, a_{\text{sub}.i})$ as the number of symbol sequences with the first i symbols being $a_{\text{sub}.0}, a_{\text{sub}.1}, \dots, a_{\text{sub}.i}$ and the sequence energy less than or equal to $E_{\text{sub}.m}$. Take the pre-defined value $E_{\text{sub}.m}=28$, the symbol set $\phi=\{1, 3, 5\}$, $N_a=4$ as an example. Then,

[0062] $T_{\text{sub}.3}(1,3,3,3)=1$ since $[1,3,3,3]$ itself has an energy less than or equal to 28;

[0063] $T_{\text{sub}.2}(1,3,5)=0$ since $1^{\text{sup}.2}+3^{\text{sup}.2}+5^{\text{sup}.2}=35>28$;

[0064] $T_{\text{sub}.2}(1,3,3)=2$ since both $[1,3,3,1]$ and $[1,3,3,3]$ have energy not greater than 28.

[0065] Take $K_d=4$ as an example. The encoding from the first bit sequence $u=[u_{\text{sub}.0}, u_{\text{sub}.1}, u_{\text{sub}.2}, u_{\text{sub}.3}]$ to the first symbol sequence A = $[A_{\text{sub}.0}, A_{\text{sub}.1}, A_{\text{sub}.2}, A_{\text{sub}.3}]$ is in Algorithm 1.

[0066] Algorithm 1 may relate to one of the specific implementation of the energy threshold encoding. This can include computing u into an integer

$$[00026] I = \sum_{i=0}^3 \text{Math. } u_i \cdot \text{Math. } 2^i$$

and set $I_{\text{SUB}.0}=1$.

[0067] For $i=0, 1, 2, 3$, the algorithm can include finding $A_{\text{sub}.i}$ in the symbol set $\Phi=\{1, 3, 5\}$ such that

$$[00027] \text{Math. } T_i(A_0, A_1, A_{i-1}, a) \leq I_i < \text{Math. } T_i(A_0, A_1, A_{i-1}, a),$$

and if $i < N_a - 1$, compute

$$[00028] I_{i+1} = I_i - \text{Math. } T_i(A_0, A_1, A_{i-1}, a),$$

output $[A_{\text{sub}.0}, A_{\text{sub}.1}, A_{\text{sub}.2}, A_{\text{sub}.3}]$ as the first symbol sequence A.

[0068] Further, in a specific example of this embodiment, the minimum energy encoding may encode the first bit sequence u into the first symbol sequence A such that the $2^{\text{sup}.K_d}$ possible output of the minimum energy encoding is the $2^{\text{sup}.K_d}$ symbol sequences with the lowest sequence energy among all the $M_a^{\text{sup}.N_a}$ symbol sequences with each symbol chosen from the symbol set.

[0069] Table 8 may provide one of the specific implementation of the minimum energy encoding is that converting the first bit sequence u into an integer

$$[00029] I = \sum_{i=0}^{K_d-1} \text{Math. } u_i \cdot \text{Math. } 2^i$$

such that the I-th minimum energy sequence is the output of the minimum energy encoding for the symbol set $\Phi=\{1,3\}$, $K_d=3$, $N_a=4$.

TABLE-US-00012 TABLE 8 One of the specific implementation of the minimum energy encoding for $K_d = 3$ and $N_a = 4$ for the symbol set $\Phi = \{1, 3\}$ the first bit sequence The first symbol sequence $u = [u_{\text{sub}.0}, u_{\text{sub}.1}, u_{\text{sub}.2}]$ A = $[A_{\text{sub}.0}, A_{\text{sub}.1}, A_{\text{sub}.2}, A_{\text{sub}.3}, A_{\text{sub}.4}]$ 000 1111 100 1113 010 1131 110 1311 001 3111 101 1133 011 1313 111 1331

[0070] Further, in a specific example of this embodiment, the variable-length encoding may include a variable-to-variable-length encoding or a fix-to-variable-length encoding, where a variable-to-variable-length may encode the first bit sequence u into the first symbol sequence using a one-to-one mapping between a first prefix-free code and a second prefix-free code. A fix-to-variable-length encoding may encode the first bit sequence u into the first symbol sequence using a one-to-one mapping between a first fix-length bit sequence set and a second prefix-free code. A prefix-free code may have the following two properties: (i) codewords have different lengths; (ii) any codeword is not the prefix of any other codewords such that if $[a_{\text{sub}.0}, a_{\text{sub}.1}, \dots, a_{\text{sub}.n-1}]$ is a codeword of length n, there is no any other codewords of the form $[a_{\text{sub}.0}, a_{\text{sub}.1}, \dots, a_{\text{sub}.n-1}, b_{\text{sub}.0}, b_{\text{sub}.1}, \dots, b_{\text{sub}.m-1}]$ with $m > 0$. The first prefix-free code is with bits as elements in a codeword. The second prefix-free code may be with symbols as elements in a codeword. One of the specific example of the first prefix-free code is $\{0, 10, 110, 111\}$ with elements in codewords being bits. One of the specific example of the second prefix-free code is $\{1111, 1113, 33, 53\}$ with elements in codewords chosen from the symbol set $\Phi=\{1, 3, 5\}$. One of the specific example of the variable-to-variable-length encoding is the one-to-one mapping between the prefix-free code $\{0, 10, 110, 111\}$ and the prefix-free code $\{1111, 1113, 33, 53\}$ as follows. [0071] $0 \rightarrow 1111$ [0072] $10 \rightarrow 1113$ [0073] $110 \rightarrow 33$ [0074] $111 \rightarrow 53$

[0075] Table 9 may provide one of the specific example of the variable-to-variable-length encoding for the symbol set $\Phi=\{1, 3\}$. One of the specific example of the fix-to-variable-length is the one-to-one mapping between the fix-length bit sequence set $\{11, 10, 01\}$ and the prefix-free code $\{1111, 1113, 33\}$ as follows. [0076] $11 \rightarrow 1111$ [0077] $10 \rightarrow 1113$ [0078] $01 \rightarrow 33$

[0079] One of the specific example of the fix-to-variable-length is the one-to-one mapping between the fix-length bit sequence $\{11, 10, 01, 00\}$ and the prefix-free code $\{1111, 1113, 33, 53\}$ as follows. [0080] $11 \rightarrow 1111$ [0081] $10 \rightarrow 1113$ [0082] $01 \rightarrow 33$ [0083] $00 \rightarrow 53$

TABLE-US-00013 TABLE 9 One of the specific example of the variable-to-variable-length encoding for the symbol set $\Phi = \{1, 3\}$ the first prefix-free code the second prefix-free code 0 111111 100 113 1010 111113 1011 11113 1100 1113 1101 1311 1110 3111 111100 133 111101 3113 1111100 1313 11111101 3131 1111110 3311 11111110 3133 111111110 3313 111111111 3331

[0084] Further, in a specific example of this embodiment, the non-linear coding can encode the first bit sequence into the first symbol sequence such that there exist at least three different first bit sequences $[u_{\text{sub}.0,0}, u_{\text{sub}.0,1}, \dots, u_{\text{sub}.0,K_d-1}]$, $[u_{\text{sub}.1,0}, u_{\text{sub}.1,1}, \dots, u_{\text{sub}.1,K_d-1}]$, and

[u.sub.2,0, u.sub.2,1, . . . , u.sub.2,Kd-1] and their corresponding second bit sequences are [v.sub.2,0, v.sub.2,1, . . . , v.sub.2,Nv-1], [v.sub.1,0, v.sub.1,1, . . . , v.sub.1,Nv-1], and [v.sub.2,0, v.sub.2,1, . . . , v.sub.2,Nv-1], respectively fulfilling all the following properties. A first property may include $u_{\text{sub}.2,i} = u_{\text{sub}.1,i} + u_{\text{sub}.0,i} \bmod 2$ for $i=0, 1, 2, \dots, Kd-1$, where $\bmod 2$ is the modulo-2 operation. There may exist at least an integer i such that $A_{\text{sub}.2,i} \neq A_{\text{sub}.1,i} + A_{\text{sub}.0,i} \bmod 2$, where $\bmod 2$ is the modulo-2 operation. There may exist at least an integer i such that $A_{\text{sub}.2,i} \neq A_{\text{sub}.1,i} + A_{\text{sub}.0,i}$, where the addition is real addition.

[0085] Further, in a specific example of this embodiment, the pre-channel coding may include a non-linear coding such that there exist at least three different first bit sequences [u.sub.0,0, u.sub.0,1, . . . , u.sub.0,Kd-1], [u.sub.1,0, u.sub.1,1, . . . , u.sub.1,Kd-1], and [u.sub.2,0, u.sub.2,1, . . . , u.sub.2,Kd-1] and their corresponding second bit sequences are [v.sub.0,0, v.sub.0,1, . . . , v.sub.0,Nv-1], [v.sub.1,0, v.sub.1,1, . . . , v.sub.1,Nv-1], and [v.sub.2,0, v.sub.2,1, . . . , v.sub.2,Nv-1], respectively fulfilling all the following properties. A property may include $u_{\text{sub}.2,i} = u_{\text{sub}.1,i} + u_{\text{sub}.0,i} \bmod 2$ for $i=0, 1, 2, \dots, Kd-1$, where $\bmod 2$ is the modulo-2 operation. There may exist at least an integer i such that $v_{\text{sub}.2,i} \neq v_{\text{sub}.1,i} + v_{\text{sub}.0,i} \bmod 2$, where $\bmod 2$ is the modulo-2 operation.

[0086] Further, in a specific example of this embodiment, the first bit sequence u can be divided into $C_{\text{sub}.p}$ sub-sequences with each sub-sequence performing the pre-channel coding separately.

[0087] The labeling of the symbol set can have effect on the error correction performance of channel coding and modulation. If the input of the channel coding has more bit 1's, the output of the channel coding will have more bit 1's and have a better error correction performance. Thus, a symbol may appear more frequently in the output of the pre-channel coding should have a label with more bit 1's, i.e., a labeling with one of the following properties.

[0088] The label of the smallest symbol in the symbol set Φ has bit "1" more than the label of the largest symbol in the symbol set. The all 1's labeling is corresponding to one of the constellation points with smallest power. The all 1's labeling is corresponding to the smallest symbol in the symbol set Φ .

[0089] FIGS. 2A-2B provide examples **200A**, **200B** for the pre-channel coding before the transport block CRC attachment.

Example Embodiment 2

[0090] Example Embodiment 2 relates to pre-channel coding before a transport block CRC attachment. This embodiment may be obtained on the basis of the Embodiment 1.

[0091] In a specific example of this embodiment, the pre-channel coding is before the transport block CRC attachment. FIGS. 2A-2B provide two of the specific examples that can include at least one portion of the payload is the input of the pre-channel coding and/or both the output of the pre-channel coding and the bits excluding the at least one portion of the payload are the input of the transport block CRC attachment.

[0092] One of the benefits of the pre-channel coding before the transport block CRC attachment may be that if errors are detected by the CRC attached to the transport block, the decoding of the pre-channel coding is not needed which reduces receive complexity. Another benefit of the pre-channel coding before the transport block CRC attachment may be that the TBS is larger than that of the pre-channel coding after the transport block CRC attachment for a given number of resource elements.

[0093] FIGS. 3A-D provide examples **300A**, **300B**, **300C**, **300D** for the pre-channel coding after a transport block CRC attachment.

Example Embodiment 3

[0094] Example Embodiment 3 relates to a pre-channel coding is after a transport block CRC attachment. This embodiment may be obtained on the basis of the Embodiment 1.

[0095] In a specific example of this embodiment, the pre-channel coding is after the transport block CRC attachment.

[0096] Further, in a specific example of this embodiment, FIG. 3A and FIG. 3B may provide two of the specific example that at least one portion of the output of the transport block CRC attachment is the input of the pre-channel coding, and both the output of the pre-channel coding and the output of the transport block CRC attachment excluding the at least one portion of the output of the transport block CRC attachment are the input of the code block segmentation, and the output of the code block segmentation is the input of the code block CRC attachment, and the output of the code block CRC attachment may be the input of the channel coding.

[0097] Further, in a specific example of this embodiment, FIG. 3C and FIG. 3D may provide two of the specific example that the output of the transport block CRC attachment is the input of the code block segmentation, and at least one portion of the output of the code block segmentation is the input of the pre-channel coding, and both the output of the pre-channel coding and the output of the code block segmentation excluding the at least one portion of the output of the code block segmentation are the input of the code block CRC attachment, and the output of the code block CRC attachment may be the input of the channel coding.

[0098] Further, in a specific example of this embodiment, FIGS. 4A-4B may give two of the specific example that the output of the transport block CRC attachment is the input of the code block segmentation, and the output of the code block segmentation is the input of the code block CRC attachment, and the at least one portion of the output of the code block CRC attachment is the input of the pre-channel coding.

[0099] Both the output of the pre-channel coding and the output of the code block CRC attachment excluding the at least one portion of the output of the code block CRC attachment may be the input of the channel coding.

[0100] One of the benefits of the pre-channel coding after the transport block CRC attachment may be that if errors are detected by the CRC attached to the code blocks, the decoding of the pre-channel coding is not needed which reduces receive complexity. Another benefit of the pre-channel coding after the transport block CRC attachment may be that the TBS is larger than that of the pre-channel coding after both the transport block CRC attachment and the code block CRC attachment for a given number of resource elements.

[0101] FIGS. 4A-4B provide examples **400A**, **400B** for the pre-channel coding after the code block CRC attachment.

Example Embodiment 4

[0102] Example Embodiment 4 relates to a pre-channel coding is after a code block CRC attachment. This embodiment may be obtained on the basis of the Embodiment 1. In a specific example of this embodiment, the pre-channel coding is after the code block CRC attachment.

[0103] FIG. 4A-4B may provide specific examples that the output of the transport block CRC attachment is the input of the code block segmentation, and the output of the code block segmentation is the input of the code block CRC attachment, and at least one portion of the output of the code block CRC attachment is the input of the pre-channel coding.

[0104] Both the output of the pre-channel coding and the output of the code block CRC attachment excluding the at least one portion of the output of the code block CRC attachment may be the input of the channel coding.

[0105] One of the benefits of the pre-channel coding after the code block CRC attachment may be that the error correction compatibility of the pre-channel coding can be utilized when the transport block is received.

Example Embodiment 5

[0106] Example Embodiment 5 relates to the output of the channel coding. This embodiment may be obtained on the basis of one of the above embodiments.

[0107] In a specific example of this embodiment, the output of the channel coding comprises two parts: a first part of the output of the channel coding, denoted as $c_{\text{sub}.1}$, is the output of the symbol-to-bit conversion of the pre-channel coding, i.e., the second bit sequence v ; a second part of the output of the channel coding, denoted as $c_{\text{sub}.2}$, is the output of the channel coding excluding the first part of the output of the channel coding.

[0108] Further, in a specific example of this embodiment, if $c = [c_{\text{sub}.0}, c_{\text{sub}.1}, c_{\text{sub}.2}, \dots, c_{\text{sub}.N-1}]$ is the output of the channel coding, the first part of the output of the channel coding $c_{\text{sub}.1}$ can be $[c_{\text{sub}.K0}, c_{\text{sub}.K0+1}, c_{\text{sub}.K0+2}, \dots, c_{\text{sub}.K1-1}]$, and the second part the output of the channel coding $c_{\text{sub}.2}$ can be $[c_{\text{sub}.0}, c_{\text{sub}.1}, \dots, c_{\text{sub}.K0-1}, c_{\text{sub}.K1}, c_{\text{sub}.K1+1}, \dots, c_{\text{sub}.N-1}]$, where $0 \leq K0 < K1 < N$ and N is the length of

the output of the channel coding.

[0109] Further, in a specific example of this embodiment, the second part of the output of the channel coding comprises the parity bits computed by the channel coding.

[0110] Further, in a specific example of this embodiment, the second part of the output of the channel coding further comprises at least one of the following: The bits input to the channel coding excluding the output of the symbol-to-bit conversion of the pre-channel coding, CRC parity bits computed by the code block CRC attachment, CRC parity bits computed by the transport block CRC attachment.

Example Embodiment 6

[0111] Example Embodiment 6 relates to at least one portion of the output of the channel coding is interleaved for MQAM. This embodiment may be obtained on the basis of Example Embodiment 5.

[0112] In this embodiment, channel coding and modulation for the transport block further comprises a modulation mapper, where the modulation order of the modulation mapper is Q_m .

[0113] Further, in a specific example of this embodiment, the constellation of the modulation mapper includes an MQAM constellation.

[0114] Further, in a specific example of this embodiment, the at least one portion of the output of the channel coding is interleaved into an interleaved channel coding sequence w with every consecutive Q_m bits of the interleaved channel coding sequence w being a label for the modulation mapper by putting together the following three parts into Q_m consecutive bits of the interleaved channel coding sequence w .

[0115] Every $Q_m/2-1$ bits in the first part of the output of the channel coding being a label of a symbol from the bit-to-symbol encoding. Every $Q_m/2-1$ bits in the first part of the output of the channel coding being another label of another symbol from the bit-to-symbol encoding;

[0116] Every 2 bits in at least one portion of the second part of the output of the channel coding.

[0117] Table 10(a) may provide one of the specific implementations as follows.

[00030] $w_{i \cdot \text{Math. } Q_m} = c_{2,2i} \text{ for } i = 0, 1, 2, \dots, \text{Na} / 2 - 1$ $w_{i \cdot \text{Math. } Q_m + 1} = c_{2,2i+1} \text{ for } i = 0, 1, 2, \dots, \text{Na} / 2 - 1$

$w_{i \cdot \text{Math. } Q_m + 2 \cdot \text{Math. } j} = c_{1,2i \cdot \text{Math. } Q_a + j - 1} \text{ for } i = 0, 1, 2, \dots, \text{Na} / 2 - 1 \text{ and } j = 1, 2, \dots, Q_a$

$w_{i \cdot \text{Math. } Q_m + 2 \cdot \text{Math. } j + 1} = c_{1,(2i+1) \cdot \text{Math. } Q_a + j - 1} \text{ for } i = 0, 1, 2, \dots, \text{Na} / 2 - 1 \text{ and } j = 1, 2, \dots, Q_a$

[0118] Table 10(b) may provide one of the specific implementations as follows.

[00031] $w_{i \cdot \text{Math. } Q_m} = c_{2,2i} \text{ for } i = 0, 1, 2, \dots, \text{Na} / 2 - 1$ $w_{i \cdot \text{Math. } Q_m + 1} = c_{2,2i+1} \text{ for } i = 0, 1, 2, \dots, \text{Na} / 2 - 1$

$w_{i \cdot \text{Math. } Q_m + 2 \cdot \text{Math. } j} = c_{1,(2i+1) \cdot \text{Math. } Q_a + j - 1} \text{ for } i = 0, 1, 2, \dots, \text{Na} / 2 - 1 \text{ and } j = 1, 2, \dots, Q_a$

$w_{i \cdot \text{Math. } Q_m + 2 \cdot \text{Math. } j + 1} = c_{1,2i+1 \cdot \text{Math. } Q_a + j - 1} \text{ for } i = 0, 1, 2, \dots, \text{Na} / 2 - 1 \text{ and } j = 1, 2, \dots, Q_a$

[0119] Table 10(c) may provide one of the specific implementations as follows.

[00032] $w_{i \cdot \text{Math. } Q_m + 2 \cdot \text{Math. } j} = c_{1,2i \cdot \text{Math. } Q_a + j} \text{ for } i = 0, 1, 2, \dots, \text{Na} / 2 - 1 \text{ and } j = 0, 1, \dots, Q_a - 1$

$w_{i \cdot \text{Math. } Q_m + 2 \cdot \text{Math. } j + 1} = c_{1,(2i+1) \cdot \text{Math. } Q_a + j} \text{ for } i = 0, 1, 2, \dots, \text{Na} / 2 - 1 \text{ and } j = 0, 1, \dots, Q_a - 1$

$w_{i \cdot \text{Math. } Q_m + 2Q_a} = c_{2,2i} \text{ for } i = 0, 1, 2, \dots, \text{Na} / 2 - 1$ $w_{i \cdot \text{Math. } Q_m + 2Q_a + 1} = c_{2,2i+1} \text{ for } i = 0, 1, 2, \dots, \text{Na} / 2 - 1$

[0120] Table 10(d) may provide one of the specific implementations as follows.

[00033] $w_{i \cdot \text{Math. } Q_m + 2 \cdot \text{Math. } j} = c_{1,(2i+1) \cdot \text{Math. } Q_a + j} \text{ for } i = 0, 1, 2, \dots, \text{Na} / 2 - 1 \text{ and } j = 0, 1, \dots, Q_a - 1$

$w_{i \cdot \text{Math. } Q_m + 2 \cdot \text{Math. } j + 1} = c_{1,2i \cdot \text{Math. } Q_a + j} \text{ for } i = 0, 1, 2, \dots, \text{Na} / 2 - 1 \text{ and } j = 0, 1, \dots, Q_a - 1$

$w_{i \cdot \text{Math. } Q_m + 2Q_a} = c_{2,2i} \text{ for } i = 0, 1, 2, \dots, \text{Na} / 2 - 1$ $w_{i \cdot \text{Math. } Q_m + 2Q_a + 1} = c_{2,2i+1} \text{ for } i = 0, 1, 2, \dots, \text{Na} / 2 - 1$

[0121] Further, in a specific example of this embodiment, Tables 10(a)-10(d) may provide one of the specific implementations that $Q_a = Q_m/2 - 1$ may be the length of the label of the symbol set, (c.sub.1,2i.Math.Qa, c.sub.1,2i.Math.Qa+1, ..., c.sub.1,2i.Math.Qa+Qa-1) may be the label of the symbol A.sub.2i in the first symbol sequence A, (c.sub.1,(2i+1).Math.Qa, c.sub.1,(2i+1).Math.Qa+1, ..., c.sub.1,(2i+1).Math.Qa+Qa-1) may be the label of the symbol A.sub.2i+1 in the first symbol sequence A and Na may be the length of the first symbol sequence A.

TABLE-US-00014 TABLE 10 Examples for interleaving the output of the channel coding for $Q_m = 6$, $Q_a = 2$, and $Na = 8$. (a) c.sub.2, 0 c.sub.2, 2 c.sub.2, 4 c.sub.2, 6 .fwdarw. w.sub.0 w.sub.6 w.sub.12 w.sub.18 c.sub.2, 1 c.sub.2, 3 c.sub.2, 5 c.sub.2, 7 w.sub.1 w.sub.7 w.sub.13 w.sub.19 c.sub.1, 0 c.sub.1, 4 c.sub.1, 8 c.sub.1, 12 w.sub.2 w.sub.8 w.sub.14 w.sub.20 c.sub.1, 2 c.sub.1, 6 c.sub.1, 10 c.sub.1, 14 w.sub.3 w.sub.9 w.sub.15 w.sub.21 c.sub.1, 1 c.sub.1, 5 c.sub.1, 9 c.sub.1, 13 w.sub.4 w.sub.10 w.sub.16 w.sub.22 c.sub.1, 3 c.sub.1, 7 c.sub.1, 11 c.sub.1, 15 w.sub.5 w.sub.11 w.sub.17 w.sub.23 (b) c.sub.2, 0 c.sub.2, 2 c.sub.2, 4 c.sub.2, 6 .fwdarw. w.sub.0 w.sub.6 w.sub.12 w.sub.18 c.sub.2, 1 c.sub.2, 3 c.sub.2, 5 c.sub.2, 7 w.sub.1 w.sub.7 w.sub.13 w.sub.19 c.sub.1, 2 c.sub.1, 6 c.sub.1, 10 c.sub.1, 14 w.sub.2 w.sub.8 w.sub.14 w.sub.20 c.sub.1, 0 c.sub.1, 4 c.sub.1, 8 c.sub.1, 12 w.sub.3 w.sub.9 w.sub.15 w.sub.21 c.sub.1, 3 c.sub.1, 7 c.sub.1, 11 c.sub.1, 15 w.sub.4 w.sub.10 w.sub.16 w.sub.22 c.sub.1, 1 c.sub.1, 5 c.sub.1, 9 c.sub.1, 13 w.sub.5 w.sub.11 w.sub.17 w.sub.23 (c) c.sub.1, 0 c.sub.1, 4 c.sub.1, 8 c.sub.1, 12 .fwdarw. w.sub.0 w.sub.6 w.sub.12 w.sub.18 c.sub.1, 2 c.sub.1, 6 c.sub.1, 10 c.sub.1, 14 w.sub.1 w.sub.7 w.sub.13 w.sub.19 c.sub.1, 1 c.sub.1, 5 c.sub.1, 9 c.sub.1, 13 w.sub.2 w.sub.8 w.sub.14 w.sub.20 c.sub.1, 3 c.sub.1, 7 c.sub.1, 11 c.sub.1, 15 w.sub.3 w.sub.9 w.sub.15 w.sub.21 c.sub.2, 0 c.sub.2, 2 c.sub.2, 4 c.sub.2, 6 w.sub.4 w.sub.10 w.sub.16 w.sub.22 c.sub.2, 1 c.sub.2, 3 c.sub.2, 5 c.sub.2, 7 w.sub.5 w.sub.11 w.sub.17 w.sub.23 (d) c.sub.1, 2 c.sub.1, 6 c.sub.1, 10 c.sub.1, 14 .fwdarw. w.sub.0 w.sub.6 w.sub.12 w.sub.18 c.sub.1, 0 c.sub.1, 4 c.sub.1, 8 c.sub.1, 12 w.sub.1 w.sub.7 w.sub.13 w.sub.19 c.sub.1, 3 c.sub.1, 7 c.sub.1, 11 c.sub.1, 15 w.sub.2 w.sub.8 w.sub.14 w.sub.20 c.sub.1, 1 c.sub.1, 5 c.sub.1, 9 c.sub.1, 13 w.sub.3 w.sub.9 w.sub.15 w.sub.21 c.sub.2, 0 c.sub.2, 2 c.sub.2, 4 c.sub.2, 6 w.sub.4 w.sub.10 w.sub.16 w.sub.22 c.sub.2, 1 c.sub.2, 3 c.sub.2, 5 c.sub.2, 7 w.sub.5 w.sub.11 w.sub.17 w.sub.23

Example Embodiment 7

[0122] Example Embodiment 7 relates to at least one portion of the output of the channel coding is interleaved for all kinds of constellations, may include MQAM. This embodiment may be obtained on the basis of Example Embodiment 5. In this embodiment, channel coding and modulation for the transport block further comprises a modulation mapper, where the modulation order of the modulation mapper is Q_m .

[0123] In a specific example of this embodiment, at least one portion of the output of the channel coding is interleaved into an interleaved channel coding sequence w with every consecutive Q_m bits of the interleaved channel coding sequence w forming a label for the modulation mapper by putting together the following two parts into Q_m consecutive bits of the interleaved channel coding sequence w : Every Q_a bits in the first part of the output of the channel coding being a label of a symbol from the bit-to-symbol encoding and every $Q_m - Q_a$ bits in at least one portion of the second part of the output of the channel coding, wherein Q_a may be the length of the labeling of the symbol set and Q_m is the modulation order of the modulation mapper. Table 11(a) may provide one of the specific implementations as follows.

[00034] $w_{i \cdot \text{Math. } Q_m + j} = c_{1,i \cdot \text{Math. } Q_a + j} \text{ for } i = 0, 1, 2, \dots, \text{Na} - 1 \text{ and } j = 0, 1, \dots, Q_a - 1$,

$w_{i \cdot \text{Math. } Q_m + Q_a + j} = c_{2,i \cdot \text{Math. } (Q_m - Q_a) + j} \text{ for } i = 0, 1, 2, \dots, \text{Na} - 1 \text{ and } j = 0, 1, \dots, Q_m - Q_a - 1$,

[0124] Table 11(b) gives one of the specific implementations as follows.

[00035] $w_{i \cdot \text{Math. } Q_m + j} = c_{2,i \cdot \text{Math. } (Q_m - Q_a) + j} \text{ for } i = 0, 1, 2, \dots, \text{Na} - 1 \text{ and } j = 0, 1, \dots, Q_m - Q_a - 1$,

$w_{i \cdot \text{Math. } Q_m + (Q_m - Q_a) + j} = c_{1,i \cdot \text{Math. } Q_a + j} \text{ for } i = 0, 1, 2, \dots, \text{Na} - 1 \text{ and } j = 0, 1, \dots, Q_a - 1$,

[0125] Tables 11(a)-(b) may provide one of the specific implementations that (c.sub.1,i.Math.Qa, c.sub.1,i.Math.Qa+1, ..., c.sub.1,i.Math.Qa+Qa-1) may be the label of the symbol A.sub.i in the first symbol sequence A and Na may be the length of the first symbol sequence A.

TABLE-US-00015 TABLE 11 Examples for interleaving the output of the channel coding for $Q_m = 5$, $Q_a = 3$, and $Na = 4$. (a) c.sub.1, 0 c.sub.1, 3 c.sub.1, 6 c.sub.1, 9 .fwdarw. w.sub.0 w.sub.5 w.sub.10 w.sub.15 c.sub.1, 1 c.sub.1, 4 c.sub.1, 7 c.sub.1, 10 w.sub.1 w.sub.6 w.sub.11 w.sub.16 c.sub.1, 2 c.sub.1, 5 c.sub.1, 8 c.sub.1, 11 w.sub.2 w.sub.7 w.sub.12 w.sub.17 c.sub.2, 0 c.sub.2, 2 c.sub.2, 4 c.sub.2, 6 w.sub.3 w.sub.8 w.sub.13

[0140] $k=[k.sub.0, k.sub.1, k.sub.2, \dots, k.sub.m.Math.Nx-1]$ is a pseudo-random bit sequence generated by a pseudo-random sequence generator and $Ng \geq Nx$.

[0141] Further, in a specific example of this embodiment, the real value set Θ may be one of the following: the set $\{0, \pi/2, \pi, 3\pi/2\}$, the set $\{0, \pi\}$, a subset of the set $\{0, \pi/2, \pi, 3\pi/2, 2\pi\}$.

[0142] Further, in a specific example of this embodiment, one of the specific implementation is that each element of the complex-value sequence $g=[g.sub.0, g.sub.1, g.sub.2, \dots, g.sub.Ng-1]$ can be 0 or 1 and the element-by-element phase rotation on the modulation symbol sequence x into the scrambled modulation symbol sequence z is one of the following:

[00041] $z_i = x_i \cdot \exp(j \cdot \delta_i)$ for $i = 0, 1, 2, \dots, Nx - 1$,

$\delta_i = 2\pi \cdot (g_{i \cdot m} + 2 \cdot g_{i \cdot m + 1} + 2^2 \cdot g_{i \cdot m + 2} + 2^3 \cdot g_{i \cdot m + 3} + \dots + 2^{m-1} \cdot g_{i \cdot m + m - 1}) / (2^m)$ for $i =$

[0143] $m > 0, Ng > m \cdot Nx$, the complex-value sequence $g=[g.sub.0, g.sub.1, g.sub.2, \dots, g.sub.Ng-1]$ can be a pseudo-random bit sequence generated by a pseudo-random sequence generator and $Ng \geq m \cdot Nx$.

[0144] In a specific example of this embodiment, the scrambling and/or interleaving is a modulation symbol scrambling element-by-element modifies the phase of each modulation symbol in the modulation symbol sequence $x=[x.sub.0, x.sub.1, x.sub.2, \dots, x.sub.Nx-1]$ according to a phase rotation angle, wherein the phase rotation angle for each modulation symbol includes a value from a set of phase rotation angles Ω and wherein the modification of the phase of the modulation symbols does not modify a power of a constellation point. One of the specific implementations can be that the phase rotation angle of each modulation symbol in the modulation symbol sequence x is determined from the set of phase rotation angles Ω . One of the specific implementations can be that the set of phase rotation angles Ω includes any of 90 degrees, 180 degrees, 270 degrees, and 360 degrees. One of the specific implementations can be that the set of phase rotation angles Ω includes a subset of the set including 0 degrees, 90 degrees, 180 degrees, 270 degrees, and 360 degrees.

[0145] The scrambled modulation symbol sequence z may be the output of the channel coding and modulation scheme.

[0146] The phase rotation can be applied to constellations of any type and any modulation order and multiplying a modulation symbol by a unit-modulus complex-value do not change the power of a modulation symbol, thus preserve the probability distribution of constellation points. The output of the phase rotation using a subset of the complex value set $\{+1, -1, +j, -j\}$ is still the constellation points in an MQAM constellation and can be applied to modulation mapper of any constellation and any modulation order. The output of the phase rotation using the set of phase rotation angles Ω can include a subset of the set including 0 degrees, 90 degrees, 180 degrees, 270 degrees, and 360 degrees is still the constellation points in an MQAM constellation and can be applied to modulation mapper of any constellation and any modulation order.

Example Embodiment 10

[0147] Example Embodiment 10 may relate to interleaving after the modulation mapper. This embodiment may be obtained on the basis of one of the above example embodiments. In this embodiment, the scrambling and/or interleaving is an interleaving to permute modulation symbols in the modulation symbol sequence $x=[x.sub.0, x.sub.1, x.sub.2, \dots, x.sub.Nx-1]$ and output an interleaved modulation symbol sequence $z=[z.sub.0, z.sub.1, z.sub.2, \dots, z.sub.Nx-1]$, wherein Nx is the length of both the modulation symbol sequence x and the interleaved modulation symbol sequence z .

Example Embodiment 11

[0148] Example Embodiment 11 may relate to swap interleaving after the modulation mapper. This embodiment may be obtained on the basis of one of the above example embodiments. In a specific example of this embodiment, the scrambling and/or interleaving is an interleaving to permute two neighboring modulation symbols in the modulation symbol sequence $x=[x.sub.0, x.sub.1, x.sub.2, \dots, x.sub.Nx-1]$ according to a fourth bit sequence $s=[s(0), s(1), s(2), \dots, s(Ns-1)]$ known to both the first node and the second node and output an interleaved modulation symbol sequence $z=[z.sub.0, z.sub.1, z.sub.2, \dots, z.sub.Nx-1]$ according to one of the following: $s(i)=1$, two neighboring modulation symbols are swapped; otherwise, the two neighboring modulation symbols are not swapped, or $s(i)=0$, two neighboring modulation symbols are swapped; otherwise, the two neighboring modulation symbols are not swapped.

[0149] One of the specific implemented is as follows:

[00042] $z_{2i} = x_{2i + s(i)}$ for $i = 0, 1, 2, \dots, Nx / 2 - 1$, $z_{2i+1} = x_{2i+1 - s(i)}$ for $i = 0, 1, 2, \dots, Nx / 2 - 1$.

[0150] $s(i)=1$, the two neighboring modulation symbols may be swapped; otherwise, the two neighboring modulation symbols are not swapped and $Ns \geq Nx/2$.

[0151] One of the specific implemented is as follows:

[00043] $z_{2i} = x_{2i+1 - s(i)}$ for $i = 0, 1, 2, \dots, Nx / 2 - 1$, $z_{2i+1} = x_{2i + s(i)}$ for $i = 0, 1, 2, \dots, Nx / 2 - 1$,

[0152] $s(i)=0$, the two neighboring modulation symbols are swapped; otherwise, the two neighboring modulation symbols are not swapped and $Ns > Nx/2$.

[0153] One of the specific implemented is as follows:

[00044] $z_{2i} = x_{2i + s(2i)}$ for $i = 0, 1, 2, \dots, Nx / 2 - 1$, $z_{2i+1} = x_{2i+1 - s(2i)}$ for $i = 0, 1, 2, \dots, Nx / 2 - 1$,

[0154] $s(2i)=1$, the two neighboring modulation symbols are swapped; otherwise, the two neighboring modulation symbols are not swapped and $Ns \geq Nx$.

[0155] One of the specific implemented is as follows:

[00045] $z_{2i} = x_{2i+1 - s(2i)}$ for $i = 0, 1, 2, \dots, Nx / 2 - 1$, $z_{2i+1} = x_{2i + s(2i)}$ for $i = 0, 1, 2, \dots, Nx / 2 - 1$,

[0156] $s(2i)=0$, the two neighboring modulation symbols are swapped; otherwise, the two neighboring modulation symbols are not swapped and $Ns \geq Nx$.

[0157] One of the specific implemented is as follows:

[00046] $z_i = x_{i + (1 - 2(i \bmod 2)) \cdot s(\lfloor i/2 \rfloor)}$ for $i = 0, 1, 2, \dots, Nx - 1$,

[0158] $s(\lfloor i/2 \rfloor)=1$, the two neighboring modulation symbols are swapped; otherwise, the two neighboring modulation symbols are not swapped and $Ns \geq Nx$.

[0159] One of the specific implemented is as follows:

[00047] $z_i = x_{i + (1 - 2(i \bmod 2)) \cdot s(1 - \lfloor i/2 \rfloor)}$ for $i = 0, 1, 2, \dots, Nx - 1$,

[0160] $s(\lfloor i/2 \rfloor)=0$, the two neighboring modulation symbols are swapped; otherwise, the two neighboring modulation symbols are not swapped and $Ns \geq Nx$.

[0161] One of the specific implemented is that the fourth bit sequence $s=[s(0), s(1), s(2), \dots, s(Ns-1)]$ is a pseudo-random bit sequence generated by a pseudo-random sequence generator.

[0162] The interleaved modulation symbol sequence z may be the output of the channel coding and modulation scheme.

Example Embodiment 12

[0163] Example Embodiment 12 relates to shift interleaving after the modulation mapper. This embodiment may be obtained on the basis of one of the above example embodiments.

[0164] In a specific example of this embodiment, the scrambling and/or interleaving is an interleaving to cyclic shift the modulation symbol sequence $x=[x.sub.0, x.sub.1, x.sub.2, \dots, x.sub.Nx-1]$ into a shifted modulation symbol sequence $z=[z.sub.0, z.sub.1, z.sub.2, \dots, z.sub.Nx-1]$ according to a first integer Ls known to both the first node and the second node by one of the following.

[0165] Cyclic shift the modulation symbol sequence x to the left by Ls modulation symbols into the shifted modulation symbol sequence z .

[0166] Cyclic shift the modulation symbol sequence x to the right by Ls modulation symbols into the shifted modulation symbol sequence z.

[0167] One of the specific implementations can be as follows.

[00048] $z_i = x_{k(i)}$ for $i = 0, 1, 2, \dots, N_x - 1$, $k(i) = i + L \bmod N_x$ for $i = 0, 1, 2, \dots, N_x - 1$,

[0168] The first integer Ls is determined according to the high layer parameters. If Ls>0, the cyclic shift is a left cyclic shift. If Ls<0, the cyclic shift is a right cyclic shift. If Ls=0, there is no shift.

[0169] One example specific implementation is as follows.

[00049] $z_{k(i)} = x_i$ for $i = 0, 1, 2, \dots, N_x - 1$, $k(i) = i + L \bmod N_x$ for $i = 0, 1, 2, \dots, N_x - 1$,

[0170] The first integer Ls is determined according to the high layer parameters. If Ls>0, the cyclic shift is a right cyclic shift. If Ls<0, the cyclic shift is a left cyclic shift. If Ls=0, there is no shift.

[0171] The shifted modulation symbol sequence z may be the output of the channel coding and modulation scheme.

Example Embodiment 13

[0172] Example Embodiment 13 relates to swap and shift interleaving after the modulation mapper. This embodiment may be obtained on the basis of one of the above example embodiments.

[0173] In a specific example of this embodiment, the scrambling and/or interleaving is an interleaving both permutes two neighboring modulation symbols in the modulation symbol sequence $x=[x_{\text{sub},0}, x_{\text{sub},1}, x_{\text{sub},2}, \dots, x_{\text{sub},N_x-1}]$ according to the fourth bit sequence $s=[s(0), s(1), s(2), \dots, s(N_s-1)]$ known to both the first node and the second node and cyclic shifts the modulation symbol sequence $x=[x_{\text{sub},0}, x_{\text{sub},1}, x_{\text{sub},2}, \dots, x_{\text{sub},N_x-1}]$ according to the first integer Ls known to both the first node and the second node and output an interleaved modulation symbol sequence $z=[z_{\text{sub},0}, z_{\text{sub},1}, z_{\text{sub},2}, \dots, z_{\text{sub},N_s-1}]$ according to at least one of the following: [0174] $s(i)=1$, the two neighboring modulation symbols are swapped; otherwise, the two neighboring modulation symbols are not swapped; [0175] $s(i)=0$, the two neighboring modulation symbols are swapped; otherwise, the two neighboring modulation symbols are not swapped; [0176] Cyclic shift the modulation symbol sequence x to the left by Ls modulation symbols into the shifted modulation symbol sequence z; [0177] Cyclic shift the modulation symbol sequence x to the right by Ls modulation symbols into the shifted modulation symbol sequence z.

[0178] One example specific implemented is as follows:

[00050] $z_{2i} = x_{k(2i+s(i))}$ for $i = 0, 1, 2, \dots, N_x/2 - 1$, $z_{2i+1} = x_{k(2i+1-s(i))}$ for $i = 0, 1, 2, \dots, N_x/2 - 1$,

$k(i) = i + L \bmod N_x$ for $i = 0, 1, 2, \dots, N_x - 1$,

[0179] The first integer Ls is determined from the high layer parameters. If Ls>0, the cyclic shift is a left cyclic shift. If Ls<0, the cyclic shift is a right cyclic shift. If Ls=0, there is no shift. $N_s > N_x/2$. $s(i)=1$, the two neighboring modulation symbols are swapped; otherwise, the two neighboring modulation symbols are not swapped.

[0180] One of the specific implemented can be as follows:

[00051] $z_{2i} = x_{k(2i+1-s(i))}$ for $i = 0, 1, 2, \dots, N_x/2 - 1$, $z_{2i+1} = x_{k(2i+s(i))}$ for $i = 0, 1, 2, \dots, N_x/2 - 1$,

$k(i) = i + L \bmod N_x$ for $i = 0, 1, 2, \dots, N_x - 1$,

[0181] The first integer Ls is determined from the high layer parameters. If Ls>0, the cyclic shift is a left cyclic shift. If Ls<0, the cyclic shift is a right cyclic shift. If Ls=0, there is no shift. $s(i)=0$, the two neighboring modulation symbols are swapped; otherwise, the two neighboring modulation symbols are not swapped. $N_s \geq N_x/2$.

[0182] One of the specific implemented can be as follows:

[00052] $z_{k(i)} = x_{i+(1-2(i \bmod 2))}$, $s(i) = \lfloor i/2 \rfloor$ for $i = 0, 1, 2, \dots, N_x - 1$, $k(i) = i + L \bmod N_x$ for $i = 0, 1, 2, \dots, N_x - 1$,

[0183] The first integer Ls is determined from the high layer parameters. If Ls>0, the cyclic shift is a right cyclic shift. If Ls<0, the cyclic shift is a left cyclic shift. If Ls=0, there is no shift. $s(\lfloor i/2 \rfloor)=1$, the two neighboring modulation symbols are swapped; otherwise, the two neighboring modulation symbols are not swapped. $N_s \geq N_x$.

[0184] One of the specific implemented can be as follows:

[00053] $z_{k(i)} = x_{i+(1-2(i \bmod 2))}$, $s(i) = 1 - s(\lfloor i/2 \rfloor)$ for $i = 0, 1, 2, \dots, N_x - 1$, $k(i) = i + L \bmod N_x$ for $i = 0, 1, 2, \dots, N_x - 1$,

[0185] The first integer Ls is determined from the high layer parameters. If Ls>0, the cyclic shift is a right cyclic shift. If Ls<0, the cyclic shift is a left cyclic shift. If Ls=0, there is no shift. $s(\lfloor i/2 \rfloor)=0$, the two neighboring modulation symbols are swapped; otherwise, the two neighboring modulation symbols are not swapped. $N_s \geq N_x$.

[0186] The interleaved modulation symbol sequence z may be the output of the channel coding and modulation scheme.

[0187] In some embodiments, interleaving does not change the power of the modulation symbols, thus preserve the power of the modulation symbol and keep the probability distribution of the output modulation symbols.

Example Embodiment 14

[0188] Example Embodiment 14 relates to the scrambling and/or interleaving before the modulation mapper. This embodiment may be obtained on the basis of one of the above example embodiments.

[0189] In this embodiment, channel coding and modulation for the transport block further comprises the scrambling and/or interleaving.

[0190] In a specific example of this embodiment, the scrambling and/or interleaving is before the modulation mapper. FIGS. 2A, 3A, 3C and 4A may provide specific examples that the output of the scrambling and/or interleaving is the input of the modulation mapper.

[0191] The interleaved channel coding sequence w can include two parts: a first part of the interleaved channel coding sequence is the first part of the output of the channel coding; a second part of the interleaved channel coding sequence is the at least one portion of the second part of the output of the channel coding.

Example Embodiment 15

[0192] Example Embodiment 15 relates to scrambling on the at least one portion of the second part of the output of the channel coding. This embodiment may be obtained on the basis of one of the above example embodiments.

[0193] In a specific example of this embodiment, the scrambling and/or interleaving is a bit scrambling performs bit-by-bit modulo-2 operation to the bits of the at least one portion of the second part of the output of the channel coding in the interleaved channel coding sequence w using the fourth bit sequence s known to both the first node and the second node and outputs a scrambled sequence y. One of the specific examples of the fourth bit sequence s is a pseudo-random bit sequence generated by a pseudo-random sequence generator. One of the implementation of the bit scrambling with the first Qb bits in every Qm consecutive bits of the interleaved channel coding sequence w being the at least one portion of the second part of the output of the channel coding is as follows:

[00054] $y_{i \cdot \text{Math. } Q_m + j} = s_{i \cdot \text{Math. } Q_b + j} + w_{i \cdot \text{Math. } Q_m + j} \bmod 2$ for $i = 0, 1, 2, \dots, N_x - 1$ and $j = 0, 1, \dots, Q_b - 1$,

$y_{i \cdot \text{Math. } Q_m + j} = w_{i \cdot \text{Math. } Q_m + j}$ for $i = 0, 1, 2, \dots, N_x - 1$ and $j = Q_b, Q_b + 1, \dots, Q_m - 1$,

[0194] wherein $N_x = N_a/2$ for MQAM constellations; otherwise, $N_x = N_a$ and N_a is the length of the first symbol sequence A. $Q_b=2$ for MQAM constellations; otherwise, $Q_b>0$; the four bit sequence s with length not less than $N_x \cdot \text{Math. } Q_b$ is a pseudo-random bit sequence generated by a pseudo-random sequence generator.

[0195] One of the implementation of the bit scrambling with the last Qb bits in every Qm consecutive bits of the interleaved channel coding sequence w being the at least one portion of the second part of the output of the channel coding is as follows:

[00055] $y_{i \cdot \text{Math. } Q_m + j} = w_{i \cdot \text{Math. } Q_m + j}$ for $i = 0, 1, 2, \dots, N_x - 1$ and $j = 0, 1, \dots, Q_m - Q_b - 1$,

$y_{i, \text{Math. } Q_m + j} = s_{i, \text{Math. } Q_b + j - Q_m - Q_b} + w_{i, \text{Math. } Q_m + j} \bmod 2$ for $i = 0, 1, 2, \dots, \text{Math. } N_x - 1$ and $j = Q_m - Q_b, Q_m - Q_b + 1, \dots, \text{Math. } Q_m - 1$, [0196] wherein $N_x = N_a/2$ for MQAM constellations; otherwise, $N_x = N_a$ and N_a is the length of the first symbol sequence A. $Q_b = 2$ for MQAM constellations; otherwise, $Q_b > 0$; the fourth bit sequence s with length not less than $N_x \cdot \text{Math. } Q_b$ is a pseudo-random bit sequence generated by a pseudo-random sequence generator.

[0197] In a specific example of this embodiment, the scrambled sequence y may be the input of the modulation mapper.

Example Embodiment 16

[0198] Example Embodiment 16 relates to scrambling on the at least one portion of the second part of the output of the channel coding and interleaving the first part of the channel coding. This embodiment may be obtained on the basis of one of the above example embodiments.

[0199] In this embodiment, the modulation mapper includes an MQAM constellation.

[0200] In a specific example of this embodiment, the scrambling and/or interleaving is a combination of bit scrambling and bit interleaving to the interleaved channel coding sequence $w = [w_{\text{sub.0}}, w_{\text{sub.1}}, w_{\text{sub.2}}, \dots, w_{\text{sub.Nw}-1}]$ according to the fourth bit sequence $s = [s_{\text{sub.0}}, s_{\text{sub.1}}, s_{\text{sub.2}}, \dots, s_{\text{sub.Ns}-1}]$ known to both the first node and the second node and output a scrambled and interleaved sequence $y = [y_{\text{sub.0}}, y_{\text{sub.1}}, y_{\text{sub.2}}, \dots, y_{\text{sub.Nw}-1}]$ as follows.

[0201] bit scrambling is performed on the first 2 bits in every Q_m consecutive bits of the interleaved channel coding sequence w from the at least one portion of the second part of the output of the channel coding.

[0202] bit interleaving is performed on the rest $Q_m - 2$ bits in every Q_m consecutive bits of the interleaved channel coding sequence w from the first part of the output of the channel coding, where using an indication bit from the fourth bit sequence s , if the indication bit is of value d , the bits in the odd position (3, 5, \dots , $Q_m - 1$) of every Q_m bits is swapped with the bits in the even position (2, 4, \dots , $Q_m - 2$) of every Q_m bits; otherwise, no swapping is performed, wherein the value d is 0 or 1.

[0203] One of the specific implementation can be as follows:

[00056] $y_{i, \text{Math. } Q_m + j} = s_{3i + j} + w_{i, \text{Math. } Q_m + j} \bmod 2$ for $i = 0, 1, 2, \dots, \text{Math. } N_x - 1$ and $j = 0, 1$

$y_{i, \text{Math. } Q_m + j} = w_{i, \text{Math. } Q_m + j + s_{3i + 2}}$ for $i = 0, 1, 2, \dots, \text{Math. } N_x - 1$ and $j = 2, 4, \dots, \text{Math. } Q_m - 2$

$y_{i, \text{Math. } Q_m + j} = w_{i, \text{Math. } Q_m + j - s_{3i + 2}}$ for $i = 0, 1, 2, \dots, \text{Math. } N_x - 1$ and $j = 3, 5, \dots, \text{Math. } Q_m - 1$

[0204] where the fourth bit sequence $s = [s_{\text{sub.0}}, s_{\text{sub.1}}, s_{\text{sub.2}}, \dots, s_{\text{sub.Ns}-1}]$ is a pseudo-random bit sequence generated by a pseudo-random sequence generator, $N_s \geq 3N_w$ is the length of the fourth bit sequence s , $N_x = N_w/Q_m$, N_w is the length of the interleaved channel coding sequence w , and $d = 1$ in this example. One of the specific implementation can be as follows:

[00057] $y_{i, \text{Math. } Q_m + j} = s_{3i + j} + w_{i, \text{Math. } Q_m + j} \bmod 2$ for $i = 0, 1, 2, \dots, \text{Math. } N_x - 1$ and $j = 0, 1$

$y_{i, \text{Math. } Q_m + j} = w_{i, \text{Math. } Q_m + j + (1 - s_{3i + 2})}$ for $i = 0, 1, 2, \dots, \text{Math. } N_x - 1$ and $j = 2, 4, \dots, \text{Math. } Q_m - 2$

$y_{i, \text{Math. } Q_m + j} = w_{i, \text{Math. } Q_m + j - (1 - s_{3i + 2})}$ for $i = 0, 1, 2, \dots, \text{Math. } N_x - 1$ and $j = 3, 5, \dots, \text{Math. } Q_m - 1$

[0205] where the fourth bit sequence $s = [s_{\text{sub.0}}, s_{\text{sub.1}}, s_{\text{sub.2}}, \dots, s_{\text{sub.Ns}-1}]$ is a pseudo-random bit sequence generated by a pseudo-random sequence generator, $N_s \geq 3N_w$ is the length of the fourth bit sequence s , $N_x = N_w/Q_m$, N_w is the length of the interleaved channel coding sequence w , and $d = 0$ in this example.

[0206] In a specific example of this embodiment, the scrambled and interleaved sequence y may be the input of the modulation mapper.

[0207] In some embodiments, the scrambling to the second part of the output of the channel coding is needed in a wireless communication system with multiple devices to make the transmit powers of different devices using the same frequency and time resource random, resulting in suppressing interference between these devices. If not careful design, the scrambling may make the modulation sequence output from the channel coding and modulation scheme having constellation points uniformly and randomly distributed. However, non-uniformly distributed constellation points are need for efficient transmission. The scrambling to the second part of the output of the channel coding are designed to retain the non-uniform property of constellation points in the modulation sequence. Specifically, if there is scrambling before the modulation mapper, the scrambling can not operate on the output of the pre-channel coding. To achieving this, the scrambling is only to the second part of the output of the channel coding, while interleaving is operated to the first part of the output of the channel coding.

Example Embodiment 17

[0208] Example Embodiment 17 relates to interleaving to the interleaved channel coding sequence by swapping. This embodiment may be obtained on the basis of one of the above example embodiments.

[0209] In a specific example of this embodiment, the scrambling and/or interleaving is an interleaving to permute the interleaved channel coding sequence $w = [w_{\text{sub.0}}, w_{\text{sub.1}}, w_{\text{sub.2}}, \dots, w_{\text{sub.Nw}-1}]$ according to the fourth bit sequence $s = [s(0), s(1), s(2), \dots, s(N_s - 1)]$ known to both the first node and the second node and output an interleaved sequence $y = [y_{\text{sub.0}}, y_{\text{sub.1}}, y_{\text{sub.2}}, \dots, y_{\text{sub.Nw}-1}]$ as follows.

[0210] every Q_m consecutive bits in the interleaved channel coding sequence w is grouped as an element, where Q_m is the modulation order of the modulation mapper.

[0211] permute these Q_m -bit elements based on the fourth bit sequence $s = [s(0), s(1), s(2), \dots, s(N_s - 1)]$ known to both the first node and the second node.

[0212] One of the specific implementations is that the sub sequence ($w_{\text{sub.i.Math. } Q_m}$, $w_{\text{sub.i.Math. } Q_m + 1}$, \dots , $w_{\text{sub.i.Math. } Q_m + Q_m - 1}$) in the interleaved channel coding sequence w is the i -th Q_m -bit element of the interleaved channel coding sequence w .

[0213] One of the specific examples for the interleaving on the interleaved channel coding sequence w is that the adjacent two Q_m -bit elements are swapped if an indication bit from the fourth bit sequence s being of the value d ; otherwise, the adjacent two Q_m -bit elements are not swapped, where $d = 0$ or 1. One of the specific implemented can be as follows:

[00058] $y_{2i, \text{Math. } Q_m + j} = w_{(2i + s(i)), \text{Math. } Q_m + j}$ for $i = 0, 1, 2, \dots, \text{Math. } N_x / 2 - 1$ and $j = 0, 1, 2, \dots, \text{Math. } Q_m - 1$,

$y_{(2i + 1), \text{Math. } Q_m + j} = w_{(2i + 1 - s(i)), \text{Math. } Q_m + j}$ for $i = 0, 1, 2, \dots, \text{Math. } N_x / 2 - 1$ and $j = 0, 1, 2, \dots, \text{Math. } Q_m - 1$,

[0214] where the indication bit $s(i) = d$ with $d = 1$, the adjacent two Q_m -bit elements are swapped; otherwise, the adjacent two Q_m -bit elements are not swapped. $N_s \geq N_x/2$.

[0215] One of the specific examples can be as follows:

[00059] $y_{2i, \text{Math. } Q_m + j} = w_{(2i + 1 - s(i)), \text{Math. } Q_m + j}$ for $i = 0, 1, 2, \dots, \text{Math. } N_x / 2 - 1$ and $j = 0, 1, 2, \dots, \text{Math. } Q_m - 1$,

$y_{(2i + 1), \text{Math. } Q_m + j} = w_{(2i - s(i)), \text{Math. } Q_m + j}$ for $i = 0, 1, 2, \dots, \text{Math. } N_x / 2 - 1$ and $j = 0, 1, 2, \dots, \text{Math. } Q_m - 1$,

[0216] where the indication bit $s(i) = d$ with $d = 0$, the adjacent two Q_m -bit elements are swapped; otherwise, the adjacent two Q_m -bit elements are not swapped. $N_s \geq N_x/2$.

[0217] One of the specific examples can be as follows:

[00060] $y_{2i, \text{Math. } Q_m + j} = w_{(2i + s(2i)), \text{Math. } Q_m + j}$ for $i = 0, 1, 2, \dots, \text{Math. } N_x / 2 - 1$ and $j = 0, 1, 2, \dots, \text{Math. } Q_m - 1$,

$y_{(2i + 1), \text{Math. } Q_m + j} = w_{(2i + 1 - s(2i)), \text{Math. } Q_m + j}$ for $i = 0, 1, 2, \dots, \text{Math. } N_x / 2 - 1$ and $j = 0, 1, 2, \dots, \text{Math. } Q_m - 1$,

[0218] where the indication bit $s(2i) = d$ with $d = 1$, the adjacent two Q_m -bit elements are swapped; otherwise, the adjacent two Q_m -bit elements are not swapped. $N_s \geq N_x$.

[0219] One of the specific examples can be as follows:

[00061] $y_{2i \cdot \text{Math. Qm} + j} = w_{(2i+1-s(2i)) \cdot \text{Math. Qm} + j}$ for $i = 0, 1, 2, \dots, \text{Math. , Nx} / 2 - 1$ and $j = 0, 1, 2, \dots, \text{Math. , Qm} - 1$,

$y_{(2i+1) \cdot \text{Math. Qm} + j} = w_{(2i-s(2i)) \cdot \text{Math. Qm} + j}$ for $i = 0, 1, 2, \dots, \text{Math. , Nx} / 2 - 1$ and $j = 0, 1, 2, \dots, \text{Math. , Qm} - 1$,

[0220] where the indication bit $s(2i)=d$ with $d=0$, the adjacent two Qm-bit elements are swapped; otherwise, the adjacent two Qm-bit elements are not swapped. $Ns \geq Nx$.

[0221] One of the specific examples can be as follows:

[00062] $y_{i \cdot \text{Math. Qm} + j} = w_{(i+(1-2(\text{imod}2)) \cdot \text{Math. s}(\text{Math. i} / 2 \cdot \text{Math. })) \cdot \text{Math. Qm} + j}$ for $i = 0, 1, 2, \dots, \text{Math. , Nx} - 1$ and $j = 0, 1, 2, \dots, \text{Math. , Qm} - 1$,

[0222] where the indication bit $s(\lfloor i/2 \rfloor)=d$ with $d=1$, the adjacent two Qm-bit elements are swapped; otherwise, the adjacent two Qm-bit elements are not swapped. $Ns \geq Nx/2$.

[0223] One of the specific implemented can be as follows:

[00063] $y_{i \cdot \text{Math. Qm} + j} = w_{(i+(1-2(\text{imod}2)) \cdot \text{Math. (1-s}(\text{Math. i} / 2 \cdot \text{Math. })) \cdot \text{Math. Qm} + j}$ for $i = 0, 1, \dots, \text{Math. , Nx} - 1$ and $j = 0, 1, \dots, \text{Math. , Qm} - 1$,

[0224] where the indication bit $s(\lfloor i/2 \rfloor)=d$ with $d=0$, the adjacent two Qm-bit elements are swapped; otherwise, the adjacent two Qm-bit elements are not swapped. $Ns \geq Nx/2$.

[0225] One of the specific implementations can be that the fourth bit sequence s is generated by a pseudo-random sequence generator.

[0226] In a specific example of this embodiment, the interleaved sequence y is the input of the modulation mapper.

Example Embodiment 18

[0227] Example Embodiment 18 relates to interleaving to the interleaved channel coding sequence by cyclic shifting. This embodiment may be obtained on the basis of one of the above example embodiments.

[0228] In a specific example of this embodiment, the scrambling and/or interleaving is an interleaving to permute the interleaved channel coding sequence $w=[w_{\text{sub.0}}, w_{\text{sub.1}}, w_{\text{sub.2}}, \dots, w_{\text{sub.Nw-1}}]$ and output an interleaved sequence $y=[y_{\text{sub.0}}, y_{\text{sub.1}}, y_{\text{sub.2}}, \dots, y_{\text{sub.Nw-1}}]$ as follows: every Qm consecutive bits in the interleaved channel coding sequence w is grouped as an element, where Qm is the modulation order of the modulation mapper, and permute these Qm-bit elements based on a second integer Lb known to both the first node and the second node.

[0229] One of the specific example for the interleaving on the interleaved channel coding sequence w is that the interleaved sequence y is a cyclic shift to the interleaved channel coding sequence w according to the second integer Lb known to both the first node and the second by one of the following.

[0230] Cyclic shift the interleaved channel coding sequence w to the left by $Lb \cdot \text{Math. Qm}$ bits into the interleaved sequence y .

[0231] Cyclic shift the interleaved channel coding sequence w to the right by $Lb \cdot \text{Math. Qm}$ bits into the interleaved sequence y .

[0232] One of the specific implementations is that the sub sequence $(w_{\text{sub.i} \cdot \text{Math. Qm}}, w_{\text{sub.i} \cdot \text{Math. Qm}+1}, \dots, w_{\text{sub.i} \cdot \text{Math. Qm}+Qm-1})$ in the interleaved channel coding sequence w is the i -th Qm-bit element of the interleaved channel coding sequence w .

[0233] One of the specific implementation can be as follows:

[00064] $y_i = w_{r(i)}$ for $i = 0, 1, 2, \dots, \text{Math. , Nw} - 1$, $r(i) = i + Lb \cdot \text{Math. Qm} \bmod Nw$ for $i = 0, 1, 2, \dots, \text{Math. , Nw} - 1$,

[0234] One of the specific implementation can be as follows:

[00065] $y_{r(i)} = w_i$ for $i = 0, 1, 2, \dots, \text{Math. , Nw} - 1$, $r(i) = i + Lb \cdot \text{Math. Qm} \bmod Nw$ for $i = 0, 1, 2, \dots, \text{Math. , Nw} - 1$,

[0235] One of the specific implementations is that the second integer Lb may be determined according to the high layer parameters. If $Lb > 0$, the cyclic shift is a right cyclic shift. If $Lb < 0$, the cyclic shift is a left cyclic shift. If $Lb = 0$, there is no shift.

[0236] In a specific example of this embodiment, the interleaved sequence y is the input of the modulation mapper.

Example Embodiment 19

[0237] Example Embodiment 19 relates to interleaving to the interleaved channel coding sequence by swapping and cyclic shifting. This embodiment may be obtained on the basis of one of the above example embodiments.

[0238] In a specific example of this embodiment, the scrambling and/or interleaving is an interleaving to permute the interleaved channel coding sequence $w=[w_{\text{sub.0}}, w_{\text{sub.1}}, w_{\text{sub.2}}, \dots, w_{\text{sub.Nw-1}}]$ according to the fourth bit sequence $s=[s(0), s(1), s(2), \dots, s(Ns-1)]$ and the second integer Lb both known to the first node and the second node and output an interleaved sequence $y=[y_{\text{sub.0}}, y_{\text{sub.1}}, y_{\text{sub.2}}, \dots, y_{\text{sub.Nw-1}}]$ as follows: every Qm consecutive bits in the interleaved channel coding sequence w is grouped as an element, where Qm is the modulation order of the modulation mapper, and permute these Qm-bit elements based on both the fourth bit sequence $s=[s(0), s(1), s(2), \dots, s(Ns-1)]$ and the second integer Lb known to both the first node and the second node.

[0239] One of the specific implementations is that the sub sequence $(w_{\text{sub.i} \cdot \text{Math. Qm}}, w_{\text{sub.i} \cdot \text{Math. Qm}+1}, \dots, w_{\text{sub.i} \cdot \text{Math. Qm}+Qm-1})$ in the interleaved channel coding sequence w is the i -th Qm-bit element of the interleaved channel coding sequence w .

[0240] Further, in a specific example of this embodiment, one of the specific example for the interleaving on the interleaved channel coding sequence w is at least one of the following: the adjacent two Qm-bit elements are swapped if an indication bit from the fourth bit sequence s being of value d ; otherwise, the adjacent two Qm-bit elements are not swapped, where $d=0$ or 1 , and cyclic shift the Qm-bit elements of the interleaved channel coding sequence w to the left or right by $Lb \cdot \text{Qm}$ -bit elements into the interleaved sequence y .

[0241] One of the specific implementations can be as follows:

[00066] $y_{2i \cdot \text{Math. Qm} + j} = w_{r(2i+s(i)) \cdot \text{Math. Qm} + j}$ for $i = 0, 1, 2, \dots, \text{Math. , Nx} / 2 - 1$ and $j = 0, 1, \dots, \text{Math. , Qm} - 1$,

$y_{(2i+1) \cdot \text{Math. Qm} + j} = w_{r(2i+1-s(i)) \cdot \text{Math. Qm} + j}$ for $i = 0, 1, 2, \dots, \text{Math. , Nx} / 2 - 1$ and $j = 0, 1, \dots, \text{Math. , Qm} - 1$,

$r(i) = i + Lb \bmod Nx$ for $i = 0, 1, 2, \dots, \text{Math. , Nx} - 1$,

[0242] the value $d=1$.

[0243] One of the specific implementations can be as follows:

[00067] $y_{2i \cdot \text{Math. Qm} + j} = w_{r(2i+1-s(i)) \cdot \text{Math. Qm} + j}$ for $i = 0, 1, 2, \dots, \text{Math. , Nx} / 2 - 1$ and $j = 0, 1, \dots, \text{Math. , Qm} - 1$,

$y_{(2i+1) \cdot \text{Math. Qm} + j} = w_{r(2i-s(i)) \cdot \text{Math. Qm} + j}$ for $i = 0, 1, 2, \dots, \text{Math. , Nx} / 2 - 1$ and $j = 0, 1, \dots, \text{Math. , Qm} - 1$,

$r(i) = i + Lb \bmod Nx$ for $i = 0, 1, 2, \dots, \text{Math. , Nx} - 1$,

[0244] the value $d=1$.

[0245] One of the specific implementations can be as follows:

[00068] $y_{r(2i) \cdot \text{Math. Qm} + j} = w_{(2i+s(i)) \cdot \text{Math. Qm} + j}$ for $i = 0, 1, 2, \dots, \text{Math. , Nx} / 2 - 1$ and $j = 0, 1, \dots, \text{Math. , Qm} - 1$,

$y_{r(2i+1) \cdot \text{Math. Qm} + j} = w_{(2i+1-s(i)) \cdot \text{Math. Qm} + j}$ for $i = 0, 1, 2, \dots, \text{Math. , Nx} / 2 - 1$ and $j = 0, 1, \dots, \text{Math. , Qm} - 1$,

$r(i) = i + Lb \bmod Nx$ for $i = 0, 1, 2, \dots, \text{Math. , Nx} - 1$,

[0246] the value $d=1$.

[0247] One of the specific implementations can be as follows:

[00069] $y_{r(2i) \cdot \text{Math. Qm} + j} = w_{(2i+s(i)) \cdot \text{Math. Qm} + j}$ for $i = 0, 1, 2, \dots, \text{Math. , Nx} / 2 - 1$ and $j = 0, 1, \dots, \text{Math. , Qm} - 1$,

$y_{r(2i+1) \cdot \text{Math. Qm} + j} = w_{(2i-s(i)) \cdot \text{Math. Qm} + j}$ for $i = 0, 1, 2, \dots, \text{Math. , Nx} / 2 - 1$ and $j = 0, 1, \dots, \text{Math. , Qm} - 1$,

$r(i) = i + Lb \bmod Nx$ for $i = 0, 1, 2, \dots, \text{Math. , Nx} - 1$,

[0248] the value $d=1$.

[0249] One of the specific implementations can be that Lb may be an integer determined according to the high layer parameters. If $Lb > 0$, the cyclic

shift is a right cyclic shift. If $L_b < 0$, the cyclic shift is a left cyclic shift. If $L_b = 0$, there is no shift, $N_x = N_w / Q_m$.

[0250] In a specific example of this embodiment, the interleaved sequence y is the input of the modulation mapper.

[0251] In some embodiments, the interleaving is based on Q_m -bit elements, thus the number of different symbols output from the bit-to-symbol encoding is not changed since the interleaving does not permute bits belong to the same symbol output from the symbol-to-bit conversion. As a result, the modulation mapper uses interleaved sequence y as input will preserve the probability of the symbols from the bit-to-symbol encoding, i.e., the probability of the constellation points.

Example Embodiment 20

[0252] Example Embodiment 20 relates to modulation mapper after the scrambling and/or interleaving. This embodiment may be obtained on the basis of one of the above example embodiments.

[0253] In a specific example of this embodiment, the modulation mapper converts every Q_m consecutive bits of the input of the modulation mapper, denoted as $y' = [y'.sub.0, y'.sub.1, y'.sub.2, \dots, y'.sub.N_y-1]$, into a modulation symbol according to the labeling of the constellation of the modulation mapper and outputs a modulation symbol sequence $x = [x.sub.0, x.sub.1, x.sub.2, \dots, x.sub.N_x-1]$. One of the specific implementation is that the label of the modulation symbol $x.sub.i$ is $(y'.sub.i.Math.Q_m, y'.sub.i.Math.Q_m+1, y'.sub.i.Math.Q_m+2, \dots, y'.sub.i.Math.Q_m+Q_m-1)$ for $i=0, 1, 2, \dots, N_x-1$ with $N_y=Q_m.Math.N_x$, wherein N_x is the length of the modulation symbol sequence x , N_y is the length of the input of the modulation mapper y' and Q_m is the modulation order of the modulation mapper.

[0254] The modulation symbol sequence x may be the output of the channel coding and modulation scheme.

Example Embodiment 21

[0255] Example Embodiment 21 relates to the pseudo-random sequence generator and high layer parameters. This embodiment may be obtained on the basis of one of the above example embodiments.

[0256] In this embodiment, the high layer parameters can be any parameters in protocol layer excluding layer 1 (L1). In this embodiment, the high layer parameters can be parameters in protocol layers excluding physical layer.

[0257] Further, in this embodiment, the pseudo-random sequence generator can be initialed by a seed determined by at least one of the following: an identity of a user equipment, an identity of a base station, an identity of a cell, an identity of a virtual cell, a cell radio network temporary identifier (C-RNTI), a modulation coding scheme C-RNTI (MCS-C-RNTI), a semi-persistent channel state information radio network temporary identifier (SP-CSI-RNTI), a coding scheme RNTI (CS-RNTI), a RNTI, a random access RNTI (RA-RNTI), the higher-layer parameter `dataScramblingIdentityPUSCH`, the higher-layer parameter `msgA-dataScramblingIdentity`, the index of the random-access preamble transmitted for `msgA`, the RA-RNTI for `msgA`, the RNTI associated with the PUSCH transmission, the higher-layer parameter `dataScramblingIdentityPDSCH`, `AdditionaldataScramblingIdentityPDSCH`, and/or physical layer cell identity.

[0258] FIG. 5 is a block diagram 500 of an example method for channel coding and modulation. The method can include performing multiple operations that include a first operation related to pre-channel coding and/or a second operation comprising a transport block cyclic redundancy check (CRC) attachment operation and/or a code block CRC attachment operation in an order (block 502).

[0259] The method can also include generating a transmission waveform by a first node using one or more bits for transmission to a second node over a wireless channel using information derived from the multiple operations (block 504). The first node can include a terminal or a network node (e.g., a base station, eNB, gNB) and the second node can include a terminal or a network node (e.g., a base station, eNB, gNB).

[0260] In some embodiments, the order includes the first operation being performed prior to the second operation including a transport block cyclic redundancy check (CRC) attachment operation.

[0261] In some embodiments, the order includes the first operation being performed after the second operation including a transport block CRC attachment operation.

[0262] In some embodiments, the order includes the first operation being performed after the second operation including a code block CRC attachment operation.

[0263] In some embodiments, the method includes performing, by the first node, a scrambling operation and an interleaving operation after a modulation mapping operation.

[0264] In some embodiments, the scrambling operation comprises a modulation symbol scrambling operation to modify a phase of modulation symbols in a modulation symbol sequence, wherein a phase rotation angle for each modulation symbol includes a value from a set of phase rotation angles, and wherein the modification of the phase of the modulation symbols does not modify a power of a constellation point.

[0265] In some embodiments, the phase of each modulation symbol in the modulation symbol sequence is a value from the set of phase rotation angles.

[0266] In some embodiments, the phase of each modulation symbol in the modulation symbol sequence is determined based on a complex-value sequence that is known by both the first node and the second node.

[0267] In some embodiments, the set of phase rotation angles includes any of 90 degrees, 180 degrees, 270 degrees, and 360 degrees.

[0268] In some embodiments, the set of phase rotation angles includes a subset of the set including 0 degrees, 90 degrees, 180 degrees, 270 degrees, and 360 degrees.

[0269] In some embodiments, the interleaving operation is performed using the modulation symbol sequence, wherein a result of the interleaving operation comprises an interleaved modulation sequence.

[0270] In some embodiments, the interleaving operation includes any of: swapping every two consecutive modulation symbols in the modulation symbol sequence based on a fourth bit sequence, performing a cyclic left or right shift to the modulation symbol sequence based on a first integer, performing the cyclic left or right shift based on the first integer and swapping every two consecutive modulation symbols based on the fourth bit sequence to the modulation symbol sequence, wherein both the fourth bit sequence and the first integer are known by both the first node and the second node.

[0271] In some embodiments, the method includes performing, by the first node, a scrambling operation and/or an interleaving operation before a modulation mapping operation.

[0272] In some embodiments, the scrambling operation comprises a bit scrambling operation that is performed to a first set of bits that excludes a set of output bits of the first operation relating to pre-channel coding.

[0273] In some embodiments, the interleaving operation is performed on the set of output bits of the first operation relating to pre-channel coding.

[0274] In some embodiments, the interleaving operation is performed on a set of at least one portion of output bits of the first operation.

[0275] In some embodiments, the interleaving operation permutes bit elements of the first set of bits, wherein a bit length of the bit elements includes a length of labeling of a symbol set included in the modulation mapping operation.

[0276] In some embodiments, the interleaving operation permutes Q_a -bit elements of the output bits of the first operation, wherein a bit length of the Q_a -bit elements includes a length of labeling of a symbol set included in the modulation mapping operation.

[0277] In some embodiments, labelling of a symbol set included in the modulation mapping operation includes a first label of a smallest symbol in the symbol set including more 1 values than that of a second label of a largest symbol in the symbol set.

[0278] In some embodiments, labelling of a symbol set included in the modulation mapping operation comprises a label for all 1 values being mapping to a constellation point with a smallest power value.

[0279] In some embodiments, the constellation point included in the modulation mapping operation relates to any of M-ary Quadrature Amplitude Modulation (MQAM), M-ary amplitude shift keying (MASK), M-ary amplitude phase shift keying (MAPSK), and M-ary Phase-Shift Keying

(MPSK). wherein M is the number of constellation points in a constellation.

[0280] In some embodiments, the first operation includes any of: source coding related encoding, energy threshold encoding, minimum energy encoding, variable-length encoding, and non-linear coding.

[0281] In some embodiments, the second operation further comprises any of: low-density parity-check coding, polar coding, turbo coding, and convolutional coding.

[0282] In some embodiments, the modulation mapping operation comprises a high-order modulation operation.

[0283] In some embodiments, the scrambling operation does not scramble output bits of the first operation and the interleaving operation does not permute consecutive bits relating to output bits that correspond to a symbol in a symbol-to-bit conversion included in the first operation.

[0284] In some embodiments, the interleaving operation includes any of: swapping every two consecutive bit elements in an input bit sequence based on a fourth bit sequence, performing a cyclic left or right shift of the bit elements of the input bit sequence based on a second integer, performing the cyclic left or right shift based on the second integer and swapping every two consecutive bit elements to the input bit sequence based on the fourth bit sequence, wherein both the fourth bit sequence and the second integer are known to both the first node and the second node.

[0285] In some embodiments, the method includes performing, by the first node, a scrambling operation and/or an interleaving operation after performance of a modulation mapping operation, wherein any of the scrambling operation and/or the interleaving operation includes any of: a modulation symbol scrambling comprising a phase rotation operation for a complex-value modulation symbol sequence that does not modify a complex-modulus of modulation symbols and permuting a modulation symbol sequence output from the modulation mapping operation.

[0286] In some embodiments, the second operation relating to CRC attachment of blocks relates to any of: transport block CRC attachment, code block segmentation, and code block CRC attachment, wherein the first operation is performed before or after the transport block CRC attachment, or the first operation is performed after the code block CRC attachment.

[0287] In some embodiments, the modulation mapping operation relates to M-ary Quadrature Amplitude Modulation (QAM) constellations and a first bit among each modulation order (Q_m) bit determines a sign of the real part of a modulation symbol, a second bit relating to a set of Q_m bits determines a sign of the imaginary part of the modulation symbol, a first $Q_m/2-1$ bit of the set of Q_m bits determines an absolute value of the real part of the modulation symbol, and a second $Q_m/2-1$ bit of the set of Q_m bits determines an absolute value of the imaginary part of the modulation symbol.

[0288] In some embodiments, the first bit among each modulation order (Q_m) bit is scrambled by a first indication bit from the fourth bit sequence, the second bit relating to the set of Q_m bits is scrambled by a second indication bit from the fourth bit sequence, and the first $Q_m/2-1$ bit of the set of Q_m bits and the second $Q_m/2-1$ bit of the set of Q_m bits are swapped if a third indication bit from the fourth bit sequence is of value d; otherwise the first $Q_m/2-1$ bit of the set of Q_m bits and the second $Q_m/2-1$ bit of the set of Q_m bits are not swapped, wherein d is 0 or 1.

[0289] In some embodiments, a part of a set of Q_a bits among the set of Q_m bits determines a complex modulus of the modulation symbol, and wherein a part of a set of Q_m-Q_a bits among the set of Q_m bits determines a phase of the modulation symbol.

[0290] In some embodiments, any of the complex-value sequence, the fourth bit sequence, the first integer and the second integer is generated by a pseudo-random sequence generator that is initialed by a seed that is determined by any of: an identity of the first node, an identity of the second node, an identity of a cell, an identity of a virtual cell, a cell radio network temporary identifier (C-RNTI), a modulation coding scheme C-RNTI (MCS-C-RNTI), a semi-persistent channel state information radio network temporary identifier (SP-CSI-RNTI), a coding scheme RNTI (CS-RNTI), a RNTI, a random access RNTI (RA-RNTI), a higher-layer parameter dataScramblingIdentityPUSCH, a higher-layer parameter msgA-dataScramblingIdentity, an index of a random-access preamble transmitted for a first message, an RA-RNTI for the first message, an RNTI associated with a PUSCH transmission, a higher-layer parameter dataScramblingIdentityPDSCH, a higher-layer parameter AdditionaldataScramblingIdentityPDSCH, and a physical layer cell identity.

Example Wireless System

[0291] FIG. 6 shows an example of a wireless communication system where techniques in accordance with one or more embodiments of the present technology can be applied. A wireless communication system 600 can include one or more base stations (B Ss) 605a, 605b, one or more wireless devices or terminals 610a, 610b, 610c, 610d, and a core network 625. A base station 605a, 605b can provide wireless service to wireless devices 610a, 610b, 610c and 610d in one or more wireless sectors. In some implementations, a base station 605a, 605b includes directional antennas to produce two or more directional beams to provide wireless coverage in different sectors. The base station may implement functionalities of a scheduling cell or a candidate cell, as described in the present document.

[0292] The core network 625 can communicate with one or more base stations 605a, 605b. The core network 625 provides connectivity with other wireless communication systems and wired communication systems. The core network may include one or more service subscription databases to store information related to the subscribed wireless devices 610a, 610b, 610c, and 610d. A first base station 605a can provide wireless service based on a first radio access technology, whereas a second base station 605b can provide wireless service based on a second radio access technology. The base stations 605a and 605b may be co-located or may be separately installed in the field according to the deployment scenario. The wireless devices 610a, 610b, 610c, and 610d can support multiple different radio access technologies.

[0293] In some implementations, a wireless communication system can include multiple networks using different wireless technologies. A dual-mode or multi-mode wireless device includes two or more wireless technologies that could be used to connect to different wireless networks.

[0294] FIG. 7 is a block diagram representation of a portion of a hardware platform. A hardware platform 705 such as a network node or a base station or a terminal or a wireless device (or UE) can include processor electronics 710 such as a microprocessor that implements one or more of the techniques presented in this document. The hardware platform 705 can include transceiver electronics 715 to send and/or receive wired or wireless signals over one or more communication interfaces such as antenna 720 or a wireline interface. The hardware platform 705 can implement other communication interfaces with defined protocols for transmitting and receiving data. The hardware platform 705 can include one or more memories (not explicitly shown) configured to store information such as data and/or instructions. In some implementations, the processor electronics 710 can include at least a portion of the transceiver electronics 715. In some embodiments, at least some of the disclosed techniques, modules or functions are implemented using the hardware platform 705.

CONCLUSION

[0295] The disclosed and other embodiments, modules and the functional operations described in this document can be implemented in digital electronic circuitry, or in computer software, firmware, or hardware, including the structures disclosed in this document and their structural equivalents, or in combinations of one or more of them. The disclosed and other embodiments can be implemented as one or more computer program products, i.e., one or more modules of computer program instructions encoded on a computer readable medium for execution by, or to control the operation of, data processing apparatus. The computer readable medium can be a machine-readable storage device, a machine-readable storage substrate, a memory device, a composition of matter effecting a machine-readable propagated signal, or a combination of one or more of them. The term "data processing apparatus" encompasses all apparatus, devices, and machines for processing data, including by way of example a programmable processor, a computer, or multiple processors or computers. The apparatus can include, in addition to hardware, code that creates an execution environment for the computer program in question, e.g., code that constitutes processor firmware, a protocol stack, a database management system, an operating system, or a combination of one or more of them. A propagated signal is an artificially generated signal, e.g., a machine-generated electrical, optical, or electromagnetic signal, that is generated to encode information for transmission to suitable receiver apparatus.

[0296] A computer program (also known as a program, software, software application, script, or code) can be written in any form of programming language, including compiled or interpreted languages, and it can be deployed in any form, including as a stand-alone program or as a module,

component, subroutine, or other unit suitable for use in a computing environment. A computer program does not necessarily correspond to a file in a file system. A program can be stored in a portion of a file that holds other programs or data (e.g., one or more scripts stored in a markup language document), in a single file dedicated to the program in question, or in multiple coordinated files (e.g., files that store one or more modules, subprograms, or portions of code). A computer program can be deployed to be executed on one computer or on multiple computers that are located at one site or distributed across multiple sites and interconnected by a communication network.

[0297] The processes and logic flows described in this document can be performed by one or more programmable processors executing one or more computer programs to perform functions by operating on input data and generating output. The processes and logic flows can also be performed by, and apparatus can also be implemented as, special purpose logic circuitry, e.g., an FPGA (field programmable gate array) or an ASIC (application specific integrated circuit).

[0298] Processors suitable for the execution of a computer program include, by way of example, both general and special purpose microprocessors, and any one or more processors of any kind of digital computer. Generally, a processor will receive instructions and data from a read only memory or a random-access memory or both. The essential elements of a computer are a processor for performing instructions and one or more memory devices for storing instructions and data. Generally, a computer will also include, or be operatively coupled to receive data from or transfer data to, or both, one or more mass storage devices for storing data, e.g., magnetic, magneto optical disks, or optical disks. However, a computer need not have such devices. Computer readable media suitable for storing computer program instructions and data include all forms of non-volatile memory, media and memory devices, including by way of example semiconductor memory devices, e.g., EPROM, EEPROM, and flash memory devices; magnetic disks, e.g., internal hard disks or removable disks; magneto optical disks; and CD ROM and DVD-ROM disks. The processor and the memory can be supplemented by, or incorporated in, special purpose logic circuitry.

[0299] While this patent document contains many specifics, these should not be construed as limitations on the scope of any invention or of what may be claimed, but rather as descriptions of features that may be specific to particular embodiments of particular inventions. Certain features that are described in this patent document in the context of separate embodiments can also be implemented in combination in a single embodiment. Conversely, various features that are described in the context of a single embodiment can also be implemented in multiple embodiments separately or in any suitable sub combination. Moreover, although features may be described above as acting in certain combinations and even initially claimed as such, one or more features from a claimed combination can in some cases be excised from the combination, and the claimed combination may be directed to a sub combination or variation of a sub combination.

[0300] Similarly, while operations are depicted in the drawings in a particular order, this should not be understood as requiring that such operations be performed in the particular order shown or in sequential order, or that all illustrated operations be performed, to achieve desirable results. Moreover, the separation of various system components in the embodiments described in this patent document should not be understood as requiring such separation in all embodiments.

[0301] Only a few implementations and examples are described, and other implementations, enhancements and variations can be made based on what is described and illustrated in this patent document.

Claims

1. A method of wireless communication, comprising: performing, at a first node, multiple operations that include a pre-channel coding operation, a transport block cyclic redundancy check (CRC) attachment operation, and a code block CRC attachment operation in an order that includes the pre-channel coding operation being performed after the transport block CRC attachment operation and prior to the code block CRC attachment operation; and generating, at the first node, a transmission waveform using one or more bits for transmission to a second node over a wireless channel using information derived from the multiple operations.
2. The method of claim 1, wherein the pre-channel coding operation comprises at least one of a bit-to-symbol encoding or a symbol-to-bit conversion.
3. The method of claim 1, wherein the information derived from the multiple operations includes a modulation symbol sequence with constellation points associated with unequal probabilities.
4. The method of claim 1, wherein the pre-channel coding operation is performed with one or more probability mass functions to calculate a number of symbols in a symbol sequence.
5. The method of claim 1, wherein the pre-channel coding operation is performed using a non-negative integer set having elements that are a number of appearances for different symbols in a symbol sequence.
6. The method of claim 1, further comprising performing a code block segmentation operation after the pre-channel coding operation and prior to the code block CRC attachment operation.
7. The method of claim 1, wherein the pre-channel coding operation is performed based on no errors being detected in an output of the transport block CRC attachment operation.
8. An apparatus for wireless communication method of wireless communication, comprising: perform, at a first node, multiple operations that include a pre-channel coding operation, a transport block cyclic redundancy check (CRC) attachment operation, and a code block CRC attachment operation in an order that includes the pre-channel coding operation being performed after the transport block CRC attachment operation and prior to the code block CRC attachment operation; and generate, at the first node, a transmission waveform using one or more bits for transmission to a second node over a wireless channel using information derived from the multiple operations.
9. The apparatus of claim 8, wherein the pre-channel coding operation comprises at least one of a bit-to-symbol encoding or a symbol-to-bit conversion.
10. The apparatus of claim 8, wherein the information derived from the multiple operations includes a modulation symbol sequence with constellation points associated with unequal probabilities.
11. The apparatus of claim 8, wherein the pre-channel coding operation is performed with one or more probability mass functions to calculate a number of symbols in a symbol sequence.
12. The apparatus of claim 8, wherein the pre-channel coding operation is performed using a non-negative integer set having elements that are a number of appearances for different symbols in a symbol sequence.
13. The apparatus of claim 8, further comprising performing a code block segmentation operation after the pre-channel coding operation and prior to the code block CRC attachment operation.
14. The apparatus of claim 8, wherein the pre-channel coding operation is performed based on no errors being detected in an output of the transport block CRC attachment operation.
15. A non-transitory computer readable medium having code stored thereon, the code when executed by a processor, causing the processor to: perform, at a first node, multiple operations that include a pre-channel coding operation, a transport block cyclic redundancy check (CRC) attachment operation, and a code block CRC attachment operation in an order that includes the pre-channel coding operation being performed after the transport block CRC attachment operation and prior to the code block CRC attachment operation; and generate, at the first node, a transmission waveform using one or more bits for transmission to a second node over a wireless channel using information derived from the multiple operations.
16. The non-transitory computer readable medium of claim 15, wherein the pre-channel coding operation comprises at least one of a bit-to-symbol encoding or a symbol-to-bit conversion.

- 17.** The non-transitory computer readable medium of claim 15, wherein the information derived from the multiple operations includes a modulation symbol sequence with constellation points associated with unequal probabilities.
- 18.** The non-transitory computer readable medium of claim 15, wherein the pre-channel coding operation is performed with one or more probability mass functions to calculate a number of symbols in a symbol sequence.
- 19.** The non-transitory computer readable medium of claim 15, wherein the pre-channel coding operation is performed using a non-negative integer set having elements that are a number of appearances for different symbols in a symbol sequence.
- 20.** The non-transitory computer readable medium of claim 15, wherein the code further causes the processor to perform a code block segmentation operation after the pre-channel coding operation and prior to the code block CRC attachment operation.
-