



República Federativa do Brasil
Ministério da Economia
Instituto Nacional da Propriedade Industrial

(11) PI 0811445-5 B1

(22) Data do Depósito: 09/05/2008

(45) Data de Concessão: 15/09/2020



(54) Título: MÉTODO PARA TRANSMITIR INFORMAÇÕES DE CONTROLE POR UMA ESTAÇÃO BASE EM UM SISTEMA DE COMUNICAÇÃO SEM FIO E MÉTODO PARA RECEBER INFORMAÇÕES DE CONTROLE POR UM EQUIPAMENTO DE USUÁRIO EM UM SISTEMA DE COMUNICAÇÃO SEM FIO

(51) Int.Cl.: H04L 27/26; H04W 72/04; H04L 1/00; H04J 13/10.

(52) CPC: H04L 27/2602; H04W 72/042; H04L 1/0057; H04J 13/10.

(30) Prioridade Unionista: 10/05/2007 US 60/917.161; 11/06/2007 US 60/943.293; 25/10/2007 KR 10-2007-0107595.

(73) Titular(es): LG ELECTRONICS INC..

(72) Inventor(es): DONG WOOK ROH; BONG HOE KIM; YOUNG WOO YUN; DONG YOUN SEO; DAE WON LEE; JUNG HOON LEE; JOON KUI AHN; KI JUN KIM.

(86) Pedido PCT: PCT KR2008002621 de 09/05/2008

(87) Publicação PCT: WO 2008/140221 de 20/11/2008

(85) Data do Início da Fase Nacional: 06/11/2009

(57) Resumo: MÉTODO PARA A TRANSMISSÃO DE INFORMAÇÕES DE CONTROLE, E MÉTODO PARA A GERAÇÃO DE PALAVRA CÓDIGO PARA ELAS. Trata-se de um método para a transmissão de informações de controle de enlace descendente e um método para a geração de uma palavra código para elas. Ao gerar um código longo com uma baixa taxa de código, um código básico cujo distância mínima entre os códigos é maximizada é repetido por um número recomendado de vezes e os bits do código repetido são ajustados. Portanto, uma condição de distância mínima entre os códigos de um códigos longo é satisfeita e simultaneamente o código pode ser simplesmente gerado. Ademais, as informações de controle podem ser transmitidas com uma baixa de erros utilizando a códigos gerado.

“MÉTODO PARA TRANSMITIR INFORMAÇÕES DE CONTROLE POR UMA ESTAÇÃO BASE EM UM SISTEMA DE COMUNICAÇÃO SEM FIO E MÉTODO PARA RECEBER INFORMAÇÕES DE CONTROLE POR UM EQUIPAMENTO DE USUÁRIO EM UM SISTEMA DE COMUNICAÇÃO SEM FIO”

5 Campo da Técnica

A presente invenção refere-se a um método para a transmissão de informações de controle de enlace descendente e um método para a geração de uma palavra código para elas.

Antecedentes da Técnica

10 Entre os fundamentos de teorias básicas de código, algumas coisas necessárias para a descrição da presente invenção serão explicadas agora.

Quando um código binário de correção de erros típico for representado como $[n, k, d]$, “n” significa o número de bits de uma palavra código codificada, “k” significa o número de bits de informação ante da codificação, e “d” significa um valor mínimo entre distâncias entre palavras código. Aqui, uma vez que a palavra código é um código binário, o comprimento da palavra código é 2^n e o número total das palavras código codificadas é 2^k . Por outro lado, o código binário de correção de erros pode ser simplesmente expresso como um código $[n, k]$ para conveniência.

20 Mais adiante neste documento, “n”, “k” e “d” terão os significados mencionados acima, exceto onde exposto em contrário.

Uma taxa de código R é definida como um valor que divide o número de bits de informação pelo número de bits de uma palavra código (ou seja, $R=k/n$).

25 Uma distância de Hamming é o número de bits cujos valores de bit correspondentes são diferentes em códigos binários que possuem o mesmo número de bits. Se uma distância de Hamming “d” for igual ou maior que $(2a+1)$, erros de até “a” podem ser corrigidos. Por exemplo, duas palavras código são “101011” e “110010”, uma distância de Hamming entre as duas palavras código é 3 e um erro pode ser corrigido.

30 Um valor mínimo entre distâncias entre quaisquer duas palavras código pertencentes a um código é definido como uma distância mínima. A distância mínima é uma das métricas importantes para avaliar o desempenho de um código. A distância de Hamming anteriormente mencionada pode ser usada como uma distância entre duas palavras código. Uma vez que uma distância entre palavras código geradas através de um processo de codificação se torna maior, visto que a probabilidade de uma palavra código correspondente ser julgada como uma palavra código diferente se torna menor, o desempenho de codificação é aumentado. O desempenho de um código é determinado por uma distância entre palavras código que possuem o pior desempenho, ou seja, uma distância mínima entre palavras código. Consequentemente, um código onde a distância mínima é maximizada mostra bom

desempenho.

Entretanto, um sistema 3GPP LTE (Projeto de Parceria de 3ª Geração de Evolução a Longo Prazo) propôs que as informações que indicam um formato de um canal de controle durante a transmissão de informações de controle sejam transmitidas através de um canal indicador de formato de controle físico (PCFICH).

O PCFICH exige a transmissão de um código que possui uma taxa de código muito baixa para reduzir a ocorrência de um erro durante a transmissão visto que as informações em um formato que transmite informações de controle são transmitidas através desse.

Entretanto, gerando-se um código de comprimento longo que possui uma taxa de código muito baixa, é difícil ajustar uma distância mínima entre palavras código a um valor máximo.

Descrição

Problema Técnico

Um objetivo da presente invenção pretende solucionar o problema proporcionando um método de construção de um código de modo que a distância mínima entre os códigos seja maximizada e gere simultaneamente um código de comprimento longo que possui uma baixa taxa código e um método para a transmissão de um sinal de controle utilizando o mesmo.

Para essa finalidade, um código de comprimento longo é gerado repetindo um código básico. Quando um código de um comprimento predeterminado não puder ser gerado simplesmente repetindo o código básico, um ajuste apropriado é realizado para gerar o código que possui o comprimento predeterminado ao mesmo tempo em que satisfaz uma condição de que uma distância mínima entre os códigos seja maximizada.

Outro objetivo da presente invenção proporciona um método para a transmissão de informações de controle de enlace descendente utilizando um código gerado pelo método descrito acima.

[Solução Técnica]

O objetivo da presente invenção pode ser atingido proporcionando-se um método para a transmissão de informações de controle através de um enlace descendente por um sistema. O método inclui: transmitir as informações de controle através de um canal de controle de enlace descendente; e transmitir informações sobre o número de símbolos de OFDM (multiplexação por divisão de frequência ortogonal) ocupados pelas informações de controle sobre o canal de controle de enlace descendente utilizando um código prescrito, onde o código prescritos é um entre

$(0, 1, 1, 0, 1, 1, 0, 1, 1, 0, 1, 1, 0, 1, 1, 0, 1, 1, 0, 1, 1, 0, 1, 1, 0, 1, 1, 0, 1, 1, 0, 1),$

$(1, 0, 1, 1, 0, 1, 1, 0, 1, 1, 0, 1, 1, 0, 1, 1, 0, 1, 1, 0, 1, 1, 0, 1, 1, 0, 1, 1, 0),$

$(1, 1, 0, 1, 1, 0, 1, 1, 0, 1, 1, 0, 1, 1, 0, 1, 1, 0, 1, 1, 0, 1, 1, 0, 1, 1, 0, 1, 1),$

[illegible]

As informações sobre o número de símbolos OFDM ocupados pelas informações de controle sobre o canal de controle de enlace descendente podem ser um indicador de formato de controle (CFI), e as informações sobre o número de símbolos OFDM ocupados pelas informações de controle sobre o canal de controle de enlace descendente podem ser transmitidas através de um canal indicador de formato de controle físico (PCFICH).

O código prescrito pode ser (0,1,1,0,1,1,0,1,1,0,1,1,0,1,1,0,1,1,0,1,1,0,1) quando o número de símbolos OFDM ocupados pelas informações de controle for 1,

(1,0,1,1,0,1,1,0,1,1,0,1,1,0,1,1,0,1,1,0,1,1,0) quando o número de símbolos OFDM ocupados pelas informações de controle for 2, e (1,1,0,1,1,0,1,1,0,1,1,0,1,1,0,1,1,0,1,1,0,1,1) quando o número de símbolos OFDM ocupados pelas informações de controle for 3.

Em outro aspecto da presente invenção, proporciona-se aqui um método para receber informações de controle através de um enlace descendente por meio de um Equipamento de Usuário (UE). O método inclui: receber informações sobre o de símbolos ocupados pelas informações de controle sobre um canal de controle de enlace descendente utilizando um código prescrito; e receber as informações de controle através do canal de controle de enlace descendente utilizando o código prescrito, onde o código prescrito é um entre (0,1,1,0,1,1,0,1,1,0,1,1,0,1,1,0,1,1,0,1,1,0,1,1,0,1),
(1,0,1,1,0,1,1,0,1,1,0,1,1,0,1,1,0,1,1,0,1,1,0,1,1,0,1,1,0),
(1,1,0,1,1,0,1,1,0,1,1,0,1,1,0,1,1,0,1,1,0,1,1,0,1,1,0,1,1), e
(0,0), de acordo com o número de símbolos OFDM ocupados pelas informações de controle.

Em um aspecto adicional da presente invenção, proporciona-se aqui um método para a geração de uma palavra código. O método inclui: obter informações sobre o número de símbolos OFDM ocupados pelas informações de controle sobre um canal de controle de enlace descendente; e gerar a palavra código de acordo com as informações obtidas, onde a palavra código é gerada repetindo qualquer um dos códigos simplex (0, 1, 1), (1, 0, 1), (1, 1, 0), e (0, 0, 0) 11 vezes e suprimindo (do inglês “puncturing”) o último bit.

Ainda em outro aspecto da presente invenção, proporciona-se aqui um método para a geração de uma palavra código. O método inclui: obter informações sobre o número de símbolos OFDM ocupados pelas informações de controle sobre um canal de controle de enlace descendente; e gerar a palavra código de acordo com as informações obtidas, onde a palavra código é gerada repetindo qualquer um dos códigos simplex (0, 1, 1), (1, 0, 1), (1, 1, 0), e (0, 0, 0) 9 vezes e inserindo qualquer um dos códigos de Hamming (0, 1, 1, 0, 1) 7 (1,

0, 1, 1, 0), (1, 1, 0, 1, 1) , e (0, 0, 0, 0, 0).

Ainda em outro aspecto da presente invenção, proporciona-se aqui um método para a geração de uma palavra código. O método inclui: obter informações sobre o número de símbolos OFDM ocupados pelas informações de controle sobre um canal de controle de enlace descendente; e gerar a palavra código de acordo com as informações obtidas, onde as informações obtidas são expressas como 2 bits de informação, e a palavra código é gerada repetindo qualquer um dos códigos simplex (0, 1, 1), (1, 0, 1), (1, 1, 0), e (0, 0, 0) 10 vezes e inserindo os bits de informação.

Efeitos Vantajosos

De acordo com os aspectos da presente invenção, um código longo que possui uma baixa taxa de código pode ser gerado por um método simples ao mesmo tempo em que a distância mínima entre os códigos é maximizada.

Ademais, as informações de controle podem ser transmitidas com uma baixa taxa de erros representando informações sobre o número de símbolos OFDM ocupados pelas informações de controle utilizando o código gerado.

Descrição dos Desenhos

Os desenhos em anexo, que são incluídos para proporcionar um entendimento adicional da invenção, ilustram as modalidades da invenção e juntamente com a descrição servem para explicar o princípio da invenção.

Nos desenhos:

A Figura 1 é um diagrama para explicar um processo de geração de um código de comprimento longo que possui uma baixa taxa de código de acordo com uma modalidade exemplificativa da presente invenção.

A Figura 2 é um fluxograma que ilustra esquematicamente um processo de geração de um código de comprimento longo que possui uma baixa taxa de código de acordo com uma modalidade exemplificativa da presente invenção.

As Figuras 3A a 3C são diagramas para explicar a relação entre uma posição de supressão de um código simplex e um bit de informação de acordo com uma modalidade exemplificativa da presente invenção.

As Figuras 4A a 4C são diagramas para explicar exemplos de conversão de linha ou coluna de um código básico em termos de mapeamento com um bit de informação de acordo com uma modalidade exemplificativa da presente invenção.

As Figuras 5A e 5B são diagramas para explicar um processo de transmissão de informações de controle utilizando um código [32,2] de acordo com uma modalidade exemplificativa da presente invenção.

Melhor Modo

Mais adiante neste documento, as modalidades preferidas da presente invenção

serão descritas em detalhes com referência aos desenhos em anexo. A descrição detalhada, que será dada abaixo com referência aos desenhos em anexo, pretende explicar as modalidades exemplificativas da presente invenção, em vez de mostrar as únicas modalidades que podem ser implementadas de acordo com a invenção. A seguinte descrição proporciona
 5 detalhes específicos para um entendimento completo das modalidades da invenção. Entretanto, um versado na técnica irá entender que a invenção pode ser praticada sem esses detalhes. Em outros casos, as estruturas e funções bem conhecidas não foram descritas em detalhes ou podem ser mostradas em forma de diagrama de bloco para evitar a ambiguidade desnecessária da descrição das modalidades da invenção. Sempre que possível, referências numéricas similares serão usadas ao longo dos desenhos para se referir às partes
 10 similares ou semelhantes.

Ao gerar um código longo que possui uma baixa taxa de código, proporciona-se um método para construir um código de modo que a distância mínima entre os códigos seja maximizada e gere simultaneamente o código longo por meio de um método simples e um
 15 método para transmitir um sinal de controle utilizando o mesmo. Para essa finalidade, uma condição de que a distância mínima entre os códigos seja maximizada será considerada em detalhes.

Na seguinte descrição da presente invenção, um conceito de uma distância mínima normalizada é introduzido como um método para comparar várias técnicas de codificação.
 20 Ou seja, em um código binário de correção de erros $[n, k, d]$, um valor que divide a distância mínima “d” pelo número “n” de bits de uma palavra código é definido como a distância mínima normalizada e a distância mínima normalizada é expressa como $\hat{d}$ para conveniência de descrição.

Ou seja, a distância mínima normalizada pode ser expressa da seguinte maneira.

Equação 1

$$\hat{d} = \frac{d}{n}$$

Mais adiante nesse documento, o limite de Plotkin será explicado para considerar as condições de modo a ajustar uma distância mínima entre os códigos a um valor máximo. Uma descrição detalhada do limite de Plotkin é descrita em detalhes em “John G. Proakis,
 30 Wiley Encyclopedia of Telecommunications, Wiley Interscience, New Jersey, 2003, vol.2, pp 929-935”.

O limite de Plotkin é resumido brevemente como se segue. Quando um código binário de correção de erros geral for expresso como $[n, k, d]$, “n” significa o número de bits de uma palavra código codificada, “k” significa o número de bits de informação ante da codificação, e “d” significa um valor mínimo entre as distâncias entre as palavras código. O limite
 35 de Plotkin descrito acima pode ser expresso da seguinte maneira.

Equação 2

$$\begin{cases} 2^k \leq 2 \left(\frac{d}{2d-n} \right) & \text{se } d = \text{par}, d > \frac{n}{2} \\ 2^k \leq 4d & \text{se } d = \text{par}, d = \frac{n}{2} \\ 2^k \leq 2 \left(\frac{d+1}{2d+1-n} \right) & \text{se } d = \text{impar}, d > \frac{n-1}{2} \\ 2^k \leq 4(d+1) & \text{se } d = \text{impar}, d = \frac{n-1}{2} \end{cases}$$

O limite de Plotkin expresso como indicado na Equação 2 pode ser novamente expresso com base na distância mínima d conforme exposto a seguir.

Equação 3

$$\begin{cases} d \leq n \times \frac{2^{k-1}}{2^k - 1} & \text{se } d = \text{par}, d > \frac{n}{2} \\ d = \frac{n}{2} \geq 2^{k-2} & \text{se } d = \text{par}, d = \frac{n}{2} \\ d \leq (n-1) \times \frac{2^{k-1}}{2^k - 1} + \frac{1}{2^k - 1} & \text{se } d = \text{impar}, d > \frac{n-1}{2} \\ d = \frac{n-1}{2} \geq 2^{k-2} - 1 & \text{se } d = \text{impar}, d = \frac{n-1}{2} \end{cases}$$

5 Um método de geração de um código expresso geralmente como $[n, k, d]$ será considerado agora com referência à Equação 3 acima de acordo com uma modalidade exemplificativa da presente invenção. Se a distância mínima for maior que metade de um comprimento de código (ou seja, $d > n/2$), deseja-se desenhar um código ótimo para satisfazer o limite de Plotkin descrito acima.

10 Aqui, supõe-se na modalidade exemplificativa da presente invenção que um código gerado possua uma baixa taxa de código (ou seja, $n > 2^k$). Quando $n=2^k$, um código ortogonal pode ser gerado e a distância mínima do código ortogonal é $n/2$. Portanto, faz sentido considerar apenas os dois casos seguintes em relação ao limite de Plotkin em uma baixa taxa de código suposta nessa modalidade exemplificativa da presente invenção.

15 Equação 4

$$\begin{cases} d \leq n \times \frac{2^{k-1}}{2^k - 1} & \text{se } d = \text{par}, d > \frac{n}{2} \\ d \leq (n-1) \times \frac{2^{k-1}}{2^k - 1} + \frac{1}{2^k - 1} & \text{se } d = \text{impar}, d > \frac{n-1}{2} \end{cases}$$

Sob as condições acima indicadas na Equação 4, os casos onde a distância mínima d é um número par e um número ímpar serão descritos agora.

Primeiro, supondo que a distância mínima d seja um número par, um valor máximo da distância mínima d é $nx \frac{2^{k-1}}{2^k - 1}$ como conhecido a partir da primeira expressão da Equação 4.

Quando considera-se a distância mínima normalizada $\hat{d}$ obtida dividindo a distância mínima pelo número de bits de uma palavra código, um valor máximo da distância mínima normalizada $\hat{d}$ é $\frac{2^{k-1}}{2^k - 1}$.

Portanto, se um código inteiro puder ser construído repetindo um código cuja distância mínima normalizada é $\frac{2^{k-1}}{2^k - 1}$, uma vez que um valor máximo de uma distância mínima é satisfeito, o código construído pode ser um código ótimo em termos da distância mínima.

Depois, supondo que a distância mínima é um número ímpar, um valor máximo da distância mínima d é $(n-1)x \frac{2^{k-1}}{2^k - 1} + \frac{1}{2^k - 1}$ como conhecido a partir da segunda expressão da Equação 4. Então, um valor máximo da distância mínima normalizada $\hat{d}$ se torna $\frac{2^{k-1}}{2^k - 1} + \frac{1}{2^k - 1}$.

Portanto, quando um código inteiro puder ser construído repetindo um código cuja distância mínima normalizada $\hat{d}$ se aproxima de $\frac{2^{k-1}}{2^k - 1}$, um valor máximo de uma distância mínima é satisfeito e o código construído se torna um código ótimo em termos da distância mínima.

Um exemplo de um código cuja distância mínima normalizada é $\frac{2^{k-1}}{2^k - 1}$ é um código simplex.

O código simplex possui uma propriedade de $[2^k-1, k, 2^{k-1}]$ e cada distância normalizada entre as palavras código é igual a $\frac{2^{k-1}}{2^k - 1}$. O código simplex pode ser geometricamente representado como vértices em um cubo unitário de uma dimensão (2^k-1) . Um código dual do código simplex é um código de Hamming que possui uma distância mínima 3.

O código simplex também é conhecido como um código de registro de deslocamento de comprimento máximo, uma sequência-m ou uma sequência de ruído pseudoaleatório (PN). Quando um código for um código binário expresso como +1 e -1 e o número de pala-

vras código for M , uma função de correlação entre palavras código diferentes é igual a $-1/(M-1)$ (onde M é um número par) ou $-1/M$ (onde M é um número ímpar). No código simplex, uma função de correlação máxima entre palavras código diferentes entre códigos binários do mesmo número de palavras código é minimizada.

5 Um código inteiro representado como $[n, k, d]$ de acordo com uma modalidade exemplificativa da presente invenção é gerado como exposto a seguir. Primeiro, um código simplex $[2^k-1, k, 2^{k-1}]$ é gerado como um código básico e repetido até um comprimento de código inteiro se tornar n . Então, um código longo cuja distância mínima normalizada é $\frac{2^{k-1}}{2^k-1}$ pode ser gerado.

10 Em alguns casos, entretanto, o comprimento de código inteiro n pode não ser um múltiplo do comprimento de código básico 2^k-1 . Em uma modalidade exemplificativa da presente invenção, propõe-se um método para a geração de um código que possui um comprimento n quando o comprimento de código inteiro n não for um múltiplo do comprimento de código básico 2^k-1 .

15 A Figura 1 é um diagrama para explicar um processo de geração de um código de comprimento longo que possui uma baixa taxa de código de acordo com uma modalidade exemplificativa da presente invenção.

Como ilustrado na Figura 1, considera-se um caso onde um comprimento de código n não é múltiplo de 2^k-1 . Ou seja, supõe-se que $n = m(2^k-1) + r$ (onde m é o número de re-
20 petições de um código básico).

Quando a relação acima for satisfeita, propõe-se um método para a geração de um código que possui um comprimento n como exposto a seguir.

Primeiro, um código simplex que satisfaz $[2^k-1, k, 2^{k-1}]$ é gerado como um código básico. Depois, gera-se um código que repete o código básico até m vezes. Por fim, r bits
25 são gerados ao suprimir o código simplex ou utilizando quaisquer outros códigos cuja distância mínima é maximizada.

Como um método para gerar os r quando k for um número pequeno, um código capaz de maximizar a distância mínima pode ser considerado testando todos os casos possíveis. Uma suposição geral para os r bits é que r é menor que um comprimento de código
30 básico 2^k-1 . Entretanto, r pode ser maior que 2^k-1 de acordo com uma modalidade exemplificativa específica da presente invenção que será descrita.

Após o código básico que possui um comprimento 2^k-1 ser repetido, um processo para ajustar os bits recomendados para gerar um código possuindo um comprimento n pode ser variado e isso será descrito mais adiante em detalhes.

35 O processo de geração de código descrito acima será descrito agora em detalhes em termos de uma distância mínima.

Primeiro, o limite de Plotkin de uma distância mínima em uma baixa taxa de código precisa considerar apenas os dois casos seguintes (onde $n = m(2^k - 1) + r$).

Equação 5

$$\left\{ \begin{aligned} d &\leq n \times \frac{2^{k-1}}{2^k - 1} && \text{se } d = \text{par}, d > \frac{n}{2} \\ &= \{m(2^k - 1) + r\} \times \frac{2^{k-1}}{2^k - 1} = m \times 2^{k-1} + r \times \frac{2^{k-1}}{2^k - 1} \\ d &\leq (n-1) \times \frac{2^{k-1}}{2^k - 1} + \frac{1}{2^k - 1} && \text{se } d = \text{impar}, d > \frac{n-1}{2} \\ &= (m(2^k - 1) + r - 1) \times \frac{2^{k-1}}{2^k - 1} + \frac{1}{2^k - 1} \\ &= m \times 2^{k-1} + \frac{(r-1)2^{k-1} + 1}{2^k - 1} \end{aligned} \right.$$

Quando um comprimento de código $n (= m(2^k-1)+r)$ não for um múltiplo de um comprimento de código básico, um método de geração de código é conforme exposto a seguir.

Ou seja, há uma parte construída repetindo um código simplex que satisfaz $[2^{k-1}, k, 2^{k-1}]$ m vezes e uma parte de código $[r, k]$ obtida adicionando r bits. O $[r, k]$ adicionado pode ser construído de muitas maneiras e deseja-se encontrar a próxima distância mínima (mais adiante nesse documento, referida como d_r para conveniência de descrição) a partir do limite de Plotkin.

Equação 6

$$[r, k, d_r] \dots d_r = \begin{cases} \leq r \times \frac{2^{k-1}}{2^k - 1} & \text{se } d = \text{par}, d > \frac{n}{2} \\ \leq \frac{(r-1)2^{k-1} + 1}{2^k - 1} & \text{se } d = \text{impar}, d > \frac{n-1}{2} \end{cases}$$

onde d_r possui um valor integral.

O código $[r, k]$ pode ser construído de muitas maneiras, por exemplo, suprimindo os bits recomendados em um código simplex usado para repetição ou utilizando outros códigos exceto para o código simplex. Entretanto, ao ajustar r bits, é importante gerar um código que se aproxima ao limite de Plotkin para uma distância mínima indicada na Equação 6.

Por outro lado, uma faixa significativa do valor r será considerada a seguir.

Conforme descrito acima, um código que possui um comprimento r pode ser construído suprimindo os bits recomendados em um código simplex usado para repetição ou

utilizando outros códigos exceto para o código simplex. Quando um código que possui um comprimento r for gerado utilizando um código diferente do código simplex de modo que uma distância mínima entre códigos seja maximizada, não há a necessidade de limitar o valor r a uma faixa específica. Em outras palavras, o valor r pode ser menor ou maior que o comprimento de código básico 2^k-1 . Se o valor r for maior que 2^k-1 , qualquer código que satisfaz uma condição de que uma distância mínima entre os códigos seja maximizada dentro do limite de Plotkin pode ser gerado. Por outro lado, quando um código diferente, exceto para o código simplex, não for usado, visto que o comprimento de código n é igual a $m(2^k-1)+r$, r é menor que (2^k-1) . Portanto, a faixa de r varia de acordo com k . Não há a necessidade de considerar um caso onde r é menor que k . Nesse caso, uma vez que r que é menor que o número k de bits a ser realmente transmitido é usado, a codificação realizada adicionando bits redundantes é insignificante. Consequentemente, uma faixa significativa de r se torna $[k, 2^k-2]$ a menos que um código, exceto para o código simplex, seja usado para gerar um código possuindo um comprimento r .

Entretanto, se r for igual a k (ou seja, $r=k$), então é desejável transmitir um determinado bit de informação (um bit sistemático) para maximizar uma distância mínima.

Mais adiante nesse documento, um método para gerar um código longo que possui uma baixa taxa de código será descrito como uma modalidade básica para conveniência de descrição.

A Figura 2 é um fluxograma que ilustra esquematicamente um processo de geração de um código de comprimento longo que possui uma baixa taxa de código de acordo com uma modalidade exemplificativa da presente invenção.

Com referência à Figura 2, um código básico cuja distância mínima entre códigos possui um valor máximo dentro do limite de Plotkin é gerado na etapa S201. Um código típico que satisfaz a condição acima pode ser um código simplex. Entretanto, se houver outros códigos cuja distância mínima entre códigos possui um valor máximo dentro do limite de Plotkin, aqueles códigos podem ser usados.

O código básico é repetido m vezes na etapa S202. Quando um código cuja distância mínima entre códigos é maximizada for repetido, uma condição de que a distância mínima seja maximizada dentro do limite de Plotkin é satisfeita como um código básico em termos de uma distância mínima normalizada. Entretanto, quando um comprimento de código n não for um múltiplo do comprimento de código básico, o outro comprimento r é gerado na etapa S203. Embora possa haver diversos métodos para a geração de um código que possui um comprimento r , é desejável ajustar o código que possui o comprimento r para possuir uma distância mínima de um valor máximo dentro do limite de Plotkin em termos da distância mínima normalizada.

Para essa finalidade, os bits recomendados no código básico usado para repetição

são suprimidos como mostrado na etapa S203-1. Se o outro comprimento r for igual ao número de bits de informação, o bit de informação é usado como o código de comprimento r como mostrado na etapa S203-2. Se o comprimento r for igual a um comprimento de qualquer código arbitrário que satisfaz uma condição de distância máxima-mínima dentro do limite de Plotkin, o código arbitrário é usado como o código de comprimento r , como mostrado na etapa S203-3. Mais adiante nesse documento, os métodos para a geração do código de comprimento r de acordo com a modalidade da presente invenção, por exemplo, será descrito um método que utiliza supressão e várias modificações de um método de geração de código. Ademais, um método para a geração de um código $[32,2]$ (ou código $(32, 2)$) aplicando o método de geração de código a um sistema 3GPP LTE, e será descrito um método para transmitir um sinal de controle utilizando o código $[32,2]$.

Primeiro, um método para a geração do código de comprimento r utilizando supressão será descrito agora de acordo com a modalidade da presente invenção.

Método para a geração de um código de comprimento r utilizando supressão

Será descrito em detalhes um método de geração de código ótimo que utiliza supressão em um caso onde um comprimento de código n não é múltiplo de (2^k-1) . Primeiro, um código básico de um comprimento curto é gerado e, então, o código básico é repetido de modo que o código básico possa ser maior do que o comprimento de código n . Os bits excessivos são suprimidos do código básico. Propõe-se a seguir um método para selecionar da melhor forma uma posição de supressão em um processo de supressão.

Ao gerar um código $[r, k]$ cuja distância mínima é maximizada dentro do limite de Plotkin, um método para construir um código simplex $[2^k-1, k, 2^k-1]$ utilizando supressão é considerado. O número de bits de supressão é 2^k-1-r e uma condição para determinar os locais de supressão de bits (2^k-1-r) é manter ao máximo uma distância mínima dentro do limite de Plotkin.

A posição de supressão pode ser determinada de diversas maneiras. Como um exemplo simples e claro, todos os locais de supressão possíveis podem ser verificados. Ou seja, se o número (2^k-1-r) de locais de supressão for pequeno e, assim, o número de possíveis locais de supressão for pequeno, todos os casos possíveis podem ser testados. Nesse caso, os locais de supressão ótimos podem ser determinados ao verificar uma condição de que uma distância mínima seja mantida ao máximo dentro do limite de Plotkin é satisfeita ao mesmo tempo em que varia os locais de supressão.

Uma posição de supressão fixa pode ser usada. Se o número de bits de supressão for 2^k-1-r , um método para suprimir sucessivamente os bits 2^k-1-r da primeira posição de bit é considerado. Em mais detalhes, os bits (2^k-1-r) correspondentes ao número de bits de supressão são suprimidos da primeira posição do código simplex $[2^k-1, k, 2^k-1]$. Entretanto, uma vez que esse método pode não garantir uma condição de que uma distância mínima seja

mantida ao máximo dentro do limite de Plotkin, é necessário verificar se um código gerado após a supressão satisfaz a condição acima.

Mais adiante nesse documento, será descrito se a condição de distância mínima foi satisfeita quando ocorrer a supressão de bits sucessivos da primeira posição suprimindo um código simplex. Se o comprimento de código n for $m(2^k-1)+r$, uma faixa significativa de r é $[k, 2^k-2]$ (de k a 2^k-2 , ambos inclusive).

Supõe-se que k seja 2. Visto que k é 2, r é menor que 3 ($=2^2-1$) e uma faixa eficaz de r é $[2, 2]$ (de 2 a 2, ambos inclusive). Portanto, apenas um caso onde $r=2$ é considerado. Nesse caso, dois bits de informação originais são usados como um código de comprimento r e esse mostra o mesmo resultado como um caso onde o primeiro bit é suprimido de um código simplex $[3, 2, 2]$.

Isso será descrito com referência às Figuras 3A a 3C.

As Figuras 3A a 3C são diagramas para explicar a relação entre uma posição de supressão de um código simplex e um bit de informação.

Uma matriz usada para gerar um código simplex $[3, 2, 2]$ é ilustrada na Figura 3A. Na Figura 3B, ilustra-se um processo para a geração de códigos simplex $(0, 0, 0)$, $(1, 0, 1)$, $(1, 1, 0)$, $(0, 1, 1)$ em relação aos bits de informação $(0, 0)$, $(0, 1)$, $(1, 0)$, $(1, 1)$, respectivamente.

Ao gerar o código de comprimento r , é eficaz utilizar apenas o bit de informação quando $r=2$. Isso mostra o mesmo resultado como um caso onde o primeiro bit do código simplex $[3, 2, 2]$ é suprimido como mostrado pela Supressão (a) na Figura 3C.

Quando o primeiro bit do código simplex $[3, 2, 2]$ for suprimido ou apenas o bit de informação for usado, o limite de Plotkin pode ser expresso da seguinte maneira.

Equação 7

$$[2, 2, d_r] \cdots d_r = \begin{cases} \leq 2 \times \frac{2^{2-1}}{2^2-1} = \frac{4}{3} & \text{se } d = \text{par}, d > \frac{n}{2} \\ \leq \frac{(2-1)2^{2-1}+1}{2^2-1} = \frac{4}{3} & \text{se } d = \text{ímpar}, d > \frac{n-1}{2} \end{cases}$$

Sob uma condição de limite de Plotkin indicada na Equação 7, é desejável que uma d_r integral satisfaça um valor máximo 1.

Entretanto, quando o primeiro bit for suprimido do código simplex $[3, 2, 2]$ ou apenas o bit de informação for usado, uma vez que a distância mínima entre os códigos é 1 como mostrado a partir da Figura 3C, a condição de que a distância mínima seja maximizada dentro do limite de Plotkin é satisfeita.

Portanto, se $k=2$ e $r=2$ ao gerar o código de comprimento r utilizando supressão, propõe-se que o primeiro bit seja suprimido do código simplex $[3, 2, 2]$ e isso mostra o mesmo resultado como um caso onde o bit de informação é usado como o código de compri-

mento r .

Depois, supõe-se que $k=3$.

Se k for 3, r é menor que 7 ($=2^3-1$) e uma faixa eficaz de r é $[3, 6]$ (de 3 a 6, ambos inclusive). Portanto, apenas os casos onde r é 3, 4, 5 e 6 são considerados.

- 5 Se r for 6, é possível gerar o código de comprimento r suprimindo o primeiro bit de um código simplex $[7, 3, 4]$. Nesse caso, o limite de Plotkin pode ser calculado como exposto a seguir.

Equação 8

$$[6, 3, d_r] \dots d_r = \begin{cases} \leq 6 \times \frac{2^{3-1}}{2^3-1} = \frac{24}{7} & \text{se } d = \text{par}, d > \frac{n}{2} \\ \leq \frac{(6-1)2^{3-1} + 1}{2^3-1} = \frac{21}{7} & \text{se } d = \text{ímpar}, d > \frac{n-1}{2} \end{cases}$$

- 10 Então, uma d_r integral possui um limite superior de $24/7$ ou $21/7$ e, portanto, um valor inteiro máximo de d_r é 3.

Um caso onde d é um número par ou um número ímpar é considerado no limite de Plotkin e então o outro caso será naturalmente derivado. Mais adiante nesse documento, apenas um caso onde d é um número par será descrito.

- 15 Quando calcula-se uma distância mínima suprimindo o primeiro bit do código simplex $[7, 3, 4]$, a distância mínima de 3 é obtida. Então, será entendido que a condição de limite de Plotkin foi satisfeita. Portanto, se $k=3$ e $r=6$ ao gerar o código de comprimento r utilizando supressão de acordo com a modalidade exemplificativa da presente invenção, propõe-se a supressão do primeiro bit do código simplex $[7, 3, 4]$.

- 20 Depois, um caso onde $k=3$ e $r=5$ é considerado. Nesse caso, descreve-se um método para suprimir dois bits sucessivos, ou seja, os primeiro e segundo bits do código simplex $[7, 3, 4]$. O limite de Plotkin é calculado como exposto a seguir.

Equação 9

$$[5, 3, d_r] \dots d_r \leq 5 \times \frac{2^{3-1}}{2^3-1} = \frac{20}{7} \quad \text{se } d = \text{par}, d > \frac{n}{2}$$

Como descrito acima, uma vez que apenas o caso onde d é um número par é considerado, um valor máximo de uma d_r integral é 2 como conhecido a partir da Equação 9.

- 25 Entretanto, uma vez que uma distância mínima quando os primeiro e segundo bits são suprimidos do código simplex $[7, 3, 4]$ é 2, a condição de limite de Plotkin acima é satisfeita. Portanto, se $k=3$ e $r=5$ ao gerar o código de comprimento r utilizando supressão, propõe-se gerar o código suprimindo dois bits sucessivos, ou seja, os primeiro e segundo bits do código simplex $[7, 3, 4]$.

- 30 Depois, um caso onde $k=3$ e $r=4$ é considerado. Nesse caso, considera-se um método para a geração de um código suprimindo três bits sucessivos, ou seja, primeiro a tercei-

ro bits do código simplex [7 3, 4]. O limite de Plotkin é calculado como exposto a seguir.

Equação 10

$$[4, 3, d_r] \dots d_r \leq 4 \times \frac{2^{3-1}}{2^3 - 1} = \frac{16}{7} \quad \text{se } d = \text{par}, d > \frac{n}{2}$$

Será entendido que um valor máximo de uma d_r integral é 2 a partir da Equação 10 acima. Quando se calcula uma distância mínima suprimindo o primeiro a terceiro bits do código simplex [7, 3, 4], a distância mínima de 2 é obtida e a condição de limite de Plotkin é satisfeita.

Portanto, se $k=3$ e $r=4$ ao gerar o código de comprimento r utilizando supressão, propõe-se gerar um a código suprimindo três bits sucessivos do primeiro a terceiro bits do código simplex [7, 3, 4].

Por fim, um caso onde $k=3$ e $r=3$ é considerado. Nesse caso, considera-se um método para a geração de um código suprimindo quatro bits sucessivos do primeiro a quarto bits do código simplex [7, 3, 4]. Nesse caso, o limite de Plotkin é calculado como exposto a seguir.

Equação 11

$$[3, 3, d_r] \dots d_r \leq 3 \times \frac{2^{3-1}}{2^3 - 1} = \frac{12}{7} \quad \text{se } d = \text{par}, d > \frac{n}{2}$$

Será entendido que um valor máximo de uma d_r integral é 1 a partir da Equação 11 acima.

Uma vez que uma distância mínima calculada suprimindo quatro bits sucessivos do código simplex [7, 3, 4] é 1, a condição de limite de Plotkin acima é satisfeita.

O caso acima onde r é igual a k também pode ser explicado como o caso onde três bits de informação (ou bits sistemáticos) são usados como o código de comprimento r .

Portanto, se $k=3$ e $r=3$ ao gerar o código de comprimento r , propõe-se suprimir quatro bits do primeiro a quarto bits do código simplex [7, 3, 4] ou utilizar o bit de informação como o código de comprimento r . Os métodos para a geração do código de comprimento r utilizando supressão quando $k=2$ e $k=3$ podem ser generalizados de modo que os métodos possam ser aplicados a casos onde k é 4 ou mais.

Ou seja, mesmo quando k for 4 ou mais, é possível gerar um código $[r, k]$ suprimindo sucessivamente os bits (2^k-1-r) da primeira posição de bit de um código simplex $[2^k-1, k, 2^k-1]$.

Modificação 1

Entretanto, as posições de supressão sucessivamente suprimidas do primeiro bit de bits (2^k-1-r) não significam as únicas posições que maximizam a distância mínima dentro do limite de Plotkin. Em alguns casos, o código de comprimento r cuja distância mínima é maximizada dentro do limite de Plotkin pode ser gerado mesmo quando os bits correspon-

tes são suprimidos em outras posições. Por exemplo, suprimir um bit quando $k=2$ e $r = 2$, o último bit pode ser suprimido em vez do primeiro bit de modo que o código de comprimento r cujo valor d_r se torne um valor máximo dentro do limite de Plotkin. Ou seja, será avaliado que d_r é 1 mesmo quando uma posição de supressão de um bit for o terceiro bit como indicado pela Supressão (b) na Figura 3C.

Modificação 2

Quando um código inteiro de um código simplex cuja distância mínima normalizada possui um valor máximo for expresso como uma matriz, uma característica de uma distância mínima não é variada mesmo que a posição de uma linha ou coluna ou posições tanto da linha como da coluna forem trocadas. Portanto, repetindo-se um código simplex m vezes, uma característica ótima de uma distância mínima não é variada mesmo que as posições de uma linha e coluna de um código que possui um comprimento pequeno sejam variadas. Consequentemente, é possível obter uma distância mínima ótima desejada mesmo que uma forma do código simplex seja variada ou fixa.

As Figuras 4A a 4C são diagramas para explicar os exemplos de conversão de linha ou coluna de um código simplex em termos de mapeamento com um bit de informação.

Como descrito acima, mesmo que a posição de uma coluna ou linha de um código simplex como um código básico usado para repetição seja trocada gerando um código de comprimento n , a característica da distância mínima não é variada. Nesse caso, a troca de uma coluna ou linha do código simplex pode significar que uma relação de mapeamento com um bit de informação foi alterada.

Ou seja, quando os bits de informação (0, 1), (1, 0), (1, 1), e (0, 0) forem mapeados até os códigos simplex #1 (1, 0, 1), (1, 1, 0), (0, 1, 1), e (0, 0, 0), respectivamente como mostrado na Figura 4A, os códigos simplex #2 cuja linha é trocada podem ser usados em vez dos códigos simplex #1.

O mesmo resultado pode ser descrito em outro aspecto como exposto a seguir. Como ilustrado na Figura 4B, quando a primeira coluna dos códigos simplex #1 for trocada pela terceira coluna desses e quando as segunda e terceira colunas dos códigos simplex #1 forem trocadas pelas primeira e segunda colunas desses, respectivamente, uma matriz de códigos que é igual aos códigos simplex #2 ilustrados na Figura 4A é gerada. A troca de uma coluna ou linha do código simplex pode corresponder a uma variação de mapeamento entre o bit de informação e o código simplex como ilustrado na Figura 4C. Ou seja, nos códigos simplex #2, os bits de informação (0, 1), (1, 0), (1, 1), e (0, 0) são mapeados até os códigos simplex (0, 1, 1), (1, 0, 1), (1, 1, 0), e (0, 0, 0), respectivamente.

Modificação 3

Os códigos cuja distância mínima é maximizada podem ser modificados de várias maneiras.

5

10

15

20

25

30

35

Primeiro, uma vez que a distância entre os códigos não é variada, mesmo quando 0 e 1 forem intercambiados, um código pode ser modificado ao intercambiar 0 e 1 de um código gerado.

Segundo, quando um código inteiro for expresso em uma matriz de modo que cada código ocupe cada linha da matriz, a característica da distância mínima não é variada mesmo que a posição de uma coluna ou uma linha ou posições da coluna e linha forem trocadas. Portanto, é possível alterar um código ao trocar uma coluna ou linha ou tanto a coluna como a linha de uma matriz de códigos de um código anteriormente gerado.

Na descrição acima, quando o comprimento de um código inteiro não for um múltiplo de um código básico ao gerar um código de comprimento longo possuindo uma baixa taxa de código utilizando o código básico possuindo uma característica de distância mínima, métodos para ajustar a outra parte do comprimento e várias modificações desses métodos foram considerados.

Mais adiante nesse documento, será descrito um exemplo de aplicação dos métodos de geração de código acima a um sistema 3GPP LTE, por exemplo, um método para gerar um código (32,2) e transmitir informações de controle utilizando o código (32,2).

Exemplo aplicado a um código (32,2)

Um indicador de formato de canal de controle (CFI) que indica o número de informações de controle entre símbolos de multiplexação por divisão de frequência ortogonal (OFDM) em um quadro OFDM em 3GPP LTE consiste em 2 bits. Uma vez que o CFI é transmitido através de um canal indicador de formato de controle físico (PCFICH) por todos os 16 símbolos de chaveamento por deslocamento de fase em quadratura (QPSK), 32 bits de codificação são necessários. Consequentemente, um código [32,2] é necessário.

Uma vez que o desempenho de um código é proporcional a uma distância entre os códigos, um código que possui desempenho excelente possui uma grande distância entre os códigos. Um código cuja distância é separada ao máximo é chamado de um código separável pela distância máxima (MDS). O código MDS pode ser um código simplex como um exemplo de um código [3,2] e um código de Hamming como um exemplo de um código [5,2].

Um código simplex [3,2] é conforme exposto a seguir:

{000, 101, 011, 110}.

No código simplex [3,2], é possível intercambiar 0 e 1. Ou seja, quando 0 e 1 forem intercambiados, o conjunto de códigos acima pode ser usado como {111, 010, 100, 001} e o código alterado também satisfaz uma característica do código simplex [3,2].

Uma posição de cada bit no código simplex [3,2] é permutável. Essa possui o mesmo resultado como um caso onde cada coluna do código simplex é alterada na descrição com referência à Figura 4B. Em outras palavras, quando os primeiro e segundo bits forem trocados a partir do conjunto de códigos acima, um conjunto de códigos {000,011,101,110} é

obtido e esse código também possui uma característica do código simplex [3,2].

O código de Hamming [5,2] é conforme exposto a seguir:

{00000, 01101, 10011, 11110} ou {00000, 01011, 10110, 11101}.

Também é possível intercambiar 0 e 1 e trocar uma posição de cada bit.

5 Entretanto, visto que é difícil gerar diretamente um código longo com uma baixa taxa de código como código [32,2], um método para a geração de um código inteiro por meio de um processo de repetição de código básico como descrito acima é considerado.

Um código básico usa o código simplex [3,2]. Após o código básico ser repetido, uma parte de bit gerada devido ao fato de o comprimento total 32 não ser um múltiplo de 3
10 pode ser processada pelos seguintes métodos, por exemplo, um método que utiliza supressão como descrito na modalidade básica e seus exemplos de modificação, um método que utiliza um bit de informação, e um método que utiliza o código de Hamming [5,2].

Primeiro, será descrito agora um método para repetir o código simplex [3,2] e suprimir um bit.

15 De acordo com a modalidade exemplificativa da presente invenção, o código simplex [3,2] é repetido 11 vezes e um bit é suprimido, desse modo gerando o código [32,2].

Uma posição de supressão de um bit pode ser qualquer posição de 33 bits. Entretanto, é desejável que um código final gerado utilizando um método de supressão satisfaça uma condição de que sua distância mínima seja maximizada dentro do limite de Plotkin.

20 O código simplex [3,2] pode utilizar uma combinação de código como {000, 101, 011, 110}. O código básico pode ser usado como formas modificadas desde que uma propriedade da distância mínima não seja variada. Por exemplo, há um método para trocar uma coluna e/ou uma linha de um código, um método para intercambiar 0 e 1 de um código, e um método para intercambiar 0 e 1 de um código e então trocar uma linha e/ou coluna do
25 código. Ao repetir o código simplex [3,2], é possível utilizar seletivamente códigos simplex 11 vezes entre vários códigos simplex [3,2].

Segundo, será descrito um método para repetir o código simplex [3,2] e inserir 2 bits de informação. De acordo com uma modalidade exemplificativa da presente invenção, o código simplex [3,2] é repetido 10 vezes e então 2 bits de informação são adicionados, desse modo gerando o código [32,2]. As posições dos bits de informação adicionados podem
30 ser a última parte do código [32,2]. Alternativamente, os bits adicionados ficam posicionados em posições arbitrárias entre 30 bits. Os 2 bits de informação podem ser sucessivamente adicionados ou separadamente adicionados. O código simplex [3,2] usado para repetição pode utilizar uma combinação de código como {000, 101, 011, 110}. Alternativamente, é
35 possível utilizar várias formas dentro da faixa de não variação de uma característica da distância mínima como descrito acima. Por exemplo, há um método para trocar uma coluna e/ou uma linha de um código, um método para intercambiar 0 e 1 de um código, e um método

do para intercambiar 0 e 1 de um código e então trocar uma linha e/ou coluna do código.

Ao repetir o código simplex [3,2], é possível utilizar seletivamente códigos simplex 10 vezes entre vários códigos simplex [3,2].

Por fim, será descrito um método para repetir o código simplex [3,2] e adicionar o
5 código de Hamming [5,2].

Nessa modalidade exemplificativa, o código simplex [3,2] é repetido 9 vezes e então o código de Hamming [5,2] é adicionado, desse modo gerando o código [32,2].

Os bits adicionados do código de Hamming [5,2] são posicionados em posições arbitrárias entre 27 bits. Os 5 bits podem ser sucessivamente adicionados ou separadamente
10 adicionados.

O código simplex [3,2] usado para repetição pode utilizar uma combinação de códigos como {000, 101, 011, 110}. Alternativamente, é possível utilizar várias formas em vez do código básico, a menos que uma característica da distância mínima seja variada como descrito acima. Por exemplo, há um método para trocar uma coluna e/ou uma linha de um código,
15 um método para intercambiar 0 e 1, e um método para intercambiar 0 e 1 e então suprimir uma linha e/ou coluna do código. O código de Hamming adicionado [5,2] pode usar qualquer um dos seguintes códigos básicos:

{00000, 01101, 10011, 11110},
{00000, 01101, 11011, 10110},
20 {00000, 10101, 01011, 11110},
{00000, 10101, 11011, 01110},
{00000, 11101, 01011, 10110},
{0000, 11101, 10011, 01110},
{01000, 00101, 10011, 11110},
25 {01000, 00101, 11011, 10110},
{01000, 10101, 00011, 11110},
{01000, 10101, 11011, 00110},
{01000, 11101, 00011, 10110},
{01000, 11101, 10011, 00110},
30 {10000, 00101, 01011, 11110},
{10000, 00101, 11011, 01110},
{10000, 01101, 00011, 11110},
{10000, 01101, 11011, 00110},
{10000, 11101, 00011, 01110},
35 {10000, 11101, 01011, 00110},
{11000, 00101, 01011, 10110},
{11000, 00101, 10011, 01110},

{11000, 01101, 00011, 10110},

{11000, 01101, 10011, 00110},

{11000, 10101, 00011, 01110},

{11000, 10101, 01011, 00110}

5 Será entendido que o código de Hamming [5,2] possui uma distância mínima de 3. Uma vez que a distância mínima de 3 é igual a um valor obtido adicionando uma distância mínima 2 do código simplex [3,2] a uma distância mínima 1 de um código de comprimento r , uma condição de que uma distância mínima entre os códigos seja maximizada dentro do limite de Plotkin é satisfeita.

10 Nessa modalidade exemplificativa, o código de Hamming [5,2] pode ser usado ao modificar o código básico descrito acima enquanto satisfaz uma característica do código de Hamming. Como modificações possíveis, há um método para alterar uma ordem de código do código de Hamming [5,2], um método para intercambiar 0 e 1 no código de Hamming [5,2], um método para intercambiar 0 e 1 no código de Hamming [5,2] e então alterar a or-

15 dem do código, um método para intercambiar as posições de bits no código de Hamming [5,2], um método para intercambiar as posições de bits no código de Hamming [5,2] e então alterar a ordem do código, um método para intercambiar 0 e 1 no código de Hamming [5,2] e então intercambiar as posições de bits, e um método para intercambiar 0 e 1, intercambiar as posições de bits, e alterar a ordem do código.

20 Ao repetir o código simplex [3,2], é possível utilizar seletivamente códigos simplex 10 vezes entre vários códigos simplex [3,2].

Exemplos do código [32,2] gerado pelos métodos descritos acima podem ser expressos conforme exposto a seguir.

Tabela 1

<0, 1,1, 0,1, 1,0, 1,1, 0,1, 1,0, 1,1, 0,1, 1,0, 1,1, 0,1,1,0,1>
<1, 0,1, 1,0, 1,1, 0,1, 1,0, 1,1, 0,1, 1,0, 1,1, 0,1, 1,0,1,1,0>
<1, 1,0, 1,1, 0,1, 1,0, 1,1, 0,1, 1,0, 1,1, 0,1, 1,0, 1,1,0,1,1>
<0, 0,0, 0,0, 0,0, 0,0, 0,0, 0,0, 0,0, 0,0, 0,0, 0,0, 0,0, 0,0,0,0,0>

25 Os quatro códigos acima podem ser gerados por um dos seguintes métodos.

Primeiro, os quatro códigos [32,2] mostrados na Tabela 1 podem ser gerados repetindo cada (0, 1, 1), (1, 0, 1), (1, 1, 0), (0, 0, 0) 11 vezes e então suprimindo o último bit do último código simplex. Como descrito na Modificação 1 acima, se $r=2$ e $k=2$, os casos onde o primeiro bit do código simplex [3,2] é suprimido e o último bit do código simplex [3,2] é su-

30 primido satisfazem uma condição de que a distância mínima seja maximizada dentro do limite de Plotkin. Portanto, um código ótimo pode ser gerado.

Segundo, quando os códigos simplex [3,2] (0, 1, 1), (1, 0, 1), (1, 1, 0), e (0, 0, 0) fo-

rem mapeados até os bits de informação (0, 1), (1, 0), (1, 1), e (0, 0), respectivamente, como ilustrado na Figura 4C, os códigos simplex [3,2] (0, 1, 1), (1, 0, 1), (1, 1, 0), e (0, 0, 0) são repetidos 10 vezes e então os bits de informação correspondentes são inseridos.

Terceiro, os quatro códigos [32,2] mostrados na Tabela 1 podem ser gerados repetindo os códigos simplex [3,2] (0, 1, 1), (1, 0, 1), (1, 1, 0), e (0, 0, 0) 9 vezes e então adicionando (0, 1, 1, 0, 1), (1, 0, 1, 1, 0), (1, 1, 0, 1, 1), e (0, 0, 0, 0, 0) como o código de Hamming [5,2].

Mais adiante nesse documento, será descrito um método para transmitir informações de controle utilizando o código gerado [32,2].

As Figuras 5A e 5B são diagramas para explicar um processo de transmissão de informações de controle utilizando um código [32,2] de acordo com uma modalidade exemplificativa da presente invenção.

As informações sobre o número de símbolos OFDM ocupados pelas informações de controle entre os símbolos OFDM dentro de um quadro OFDM em um sistema 3GPP LTE são transmitidas através de um PCFICH e o código [32,2] descrito acima é usado. Na Figura 5A, as informações de controle são transmitidas através de 2 símbolos OFDM dentro de um quadro OFDM. As informações sobre o número de símbolos OFDM ocupados pelas informações de controle podem ser expressas pelo código [32,2] como mostrado na Tabela 1.

Se o número de símbolos OFDM ocupados pelas informações de controle estiver presente como apenas 3 casos, então apenas 3 códigos podem ser usados entre os 4 códigos mostrados na Tabela 1.

A Figura 5B ilustra um exemplo que utiliza apenas as primeira, segunda e terceira linhas entre o código [32,2] mostrado na Tabela 1 quando as informações de controle ocuparem os símbolos OFDM 1, 2, e 3.

A descrição detalhada das modalidades exemplificativas da presente invenção foi dada para permitir que os versados na técnica implementem e pratiquem a invenção. Embora a invenção seja descrita com referência às modalidades exemplificativas, os versados na técnica irão avaliar que várias modificações e variações podem ser feitas na presente invenção sem que se abandone o espírito ou escopo da invenção descrito nas reivindicações em anexo.

Portanto, será entendido que essa patente não deve se limitar às modalidades específicas descritas aqui, porém deve estar de acordo com o escopo mais amplo compatível com os princípios e novas características descritas aqui.

Aplicabilidade industrial

A presente invenção proporciona um método para a geração de um código de comprimento longo que possui uma baixa taxa de código e um método para a transmissão de informações de controle utilizando o mesmo. Esses métodos podem ser diretamente aplica-

dos a um código [32,2] usado para um PCFICH em um sistema 3GPP LTE. Os métodos para gerar um código de comprimento longo que possuem uma baixa taxa de código repetindo e suprimindo um código básico, inserir um bit de informação, e utilizar códigos arbitrários exceto o código básico podem ser aplicados de forma variada como um método para gerar

5 um código utilizado para transmitir informações específicas cujo erro pode ocorrer seriamente e utilizar o código até em sistemas de comunicação exceto no sistema 3GPP LTE.

REIVINDICAÇÕES

1. Método para transmitir informações de controle por uma estação base (BS) em um sistema de comunicação sem fio, o método **CARACTERIZADO** pelo fato de que compreende:

5 transmitir, pela BS, um indicador de formato de controle para vários símbolos de multiplexação por divisão de frequências ortogonais (OFDM) utilizados para transmissão de informação de controle em um subquadro,

em que o indicador de formato de controle é baseado em um código prescrito, e

em que o código prescrito é um dentre:

10 (0,1,1,0,1,1,0,1,1,0,1,1,0,1,1,0,1,1,0,1,1,0,1,1,0,1),
 (1,0,1,1,0,1,1,0,1,1,0,1,1,0,1,1,0,1,1,0,1,1,0,1,1,0), ou
 (1,1,0,1,1,0,1,1,0,1,1,0,1,1,0,1,1,0,1,1,0,1,1,0,1,1).

2. Método, de acordo com a reivindicação 1, **CARACTERIZADO** pelo fato de que compreende adicionalmente:

15 modular, pela BS, o código prescrito com base em um esquema de chaveamento por deslocamento de fase em quadratura (QPSK).

3. Método, de acordo com a reivindicação 1, **CARACTERIZADO** pelo fato de que o indicador de formato de controle é transmitido através de um canal indicador de formato de controle físico (PCFICH).

20 4. Método, de acordo com a reivindicação 1, **CARACTERIZADO** pelo fato de que o código prescrito é:

(0,1,1,0,1,1,0,1,1,0,1,1,0,1,1,0,1,1,0,1,1,0,1,1,0,1) com base no fato do número de símbolos OFDM utilizado para informações de controle ser 1,

(1,0,1,1,0,1,1,0,1,1,0,1,1,0,1,1,0,1,1,0,1,1,0,1,1,0) com base no fato do
 25 número de símbolos OFDM utilizado para informações de controle ser 2, e

(1,1,0,1,1,0,1,1,0,1,1,0,1,1,0,1,1,0,1,1,0,1,1,0,1,1) com base no fato do número de símbolos OFDM utilizado para informações de controle ser 3.

5. Método para receber informações de controle por um equipamento de usuário (UE) em um sistema de comunicação sem fio, o método **CARACTERIZADO** pelo fato de
 30 que compreende:

receber, pelo UE, um indicador de formato de controle para vários símbolos de OFDM utilizados para recepção de informação de controle em um subquadro,

em que o indicador de formato de controle é baseado em um código prescrito, e

em que o código prescrito é um dentre:

35 (0,1,1,0,1,1,0,1,1,0,1,1,0,1,1,0,1,1,0,1,1,0,1,1,0,1),
 (1,0,1,1,0,1,1,0,1,1,0,1,1,0,1,1,0,1,1,0,1,1,0,1,1,0), ou
 (1,1,0,1,1,0,1,1,0,1,1,0,1,1,0,1,1,0,1,1,0,1,1,0,1,1).

6. Método, de acordo com a reivindicação 5, **CHARACTERIZADO** pelo fato de que o código prescrito é recebido com base em um esquema de chaveamento por deslocamento de fase em quadratura (QPSK).

5 7. Método, de acordo com a reivindicação 5, **CHARACTERIZADO** pelo fato de que a informação de controle é recebida através de um canal indicador de formato de controle físico (PCFICH).

8. Método, de acordo com a reivindicação 5, **CHARACTERIZADO** pelo fato de que o código prescrito é:

10 (0,1,1,0,1,1,0,1,1,0,1,1,0,1,1,0,1,1,0,1,1,0,1,1,0,1) com base no fato do número de símbolos OFDM utilizado para informações de controle ser 1,
 (1,0,1,1,0,1,1,0,1,1,0,1,1,0,1,1,0,1,1,0,1,1,0,1,1,0) com base no fato do número de símbolos OFDM utilizado para informações de controle ser 2, e
 (1,1,0,1,1,0,1,1,0,1,1,0,1,1,0,1,1,0,1,1,0,1,1,0,1,1) com base no fato do número de símbolos OFDM utilizado para informações de controle ser 3.

FIGURA 1

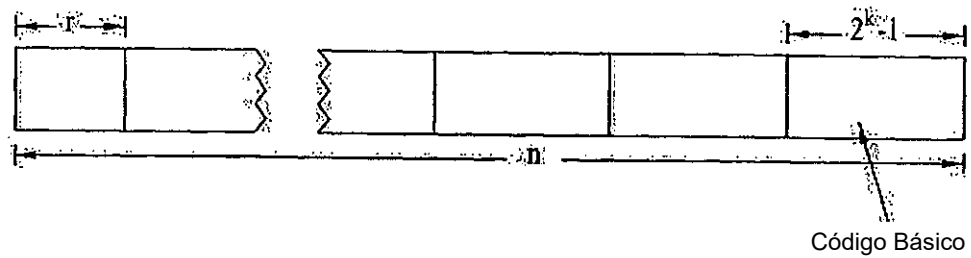


FIGURA 2

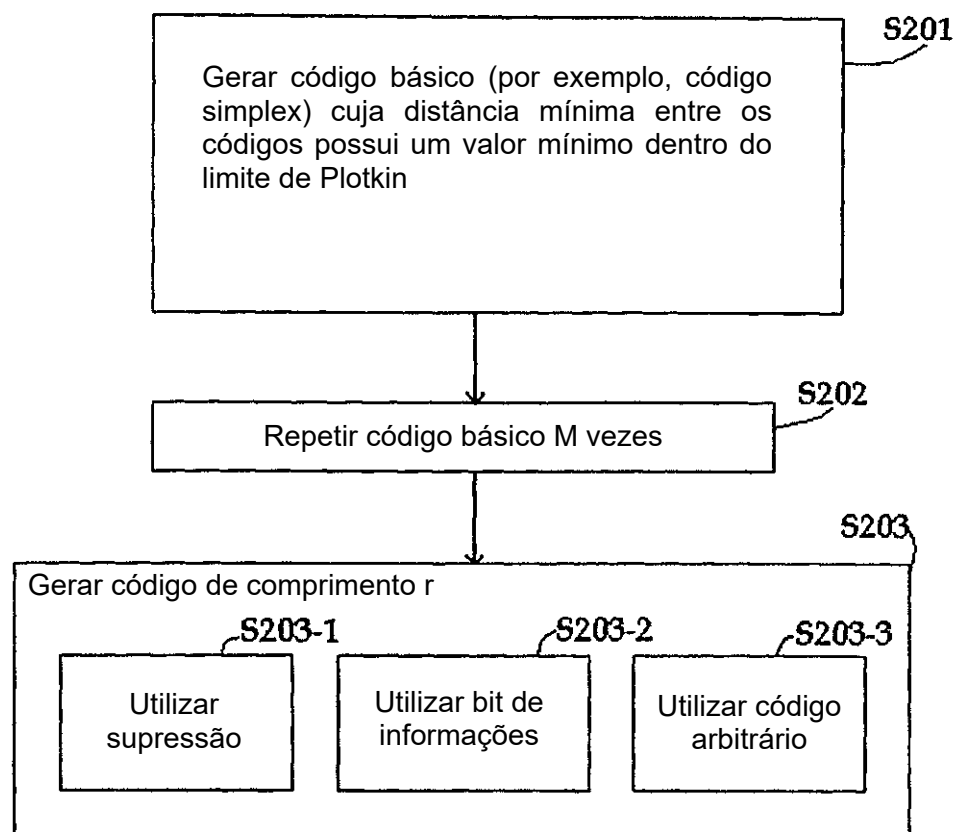


FIGURA 3

(a) Matriz de geração = $\begin{bmatrix} 1 & 1 \\ 1 & 0 \\ 0 & 1 \end{bmatrix}$

(b) $\begin{bmatrix} 1 & 1 \\ 1 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} 0 \\ 0 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}$ $\begin{bmatrix} 1 & 1 \\ 1 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} 0 \\ 1 \end{bmatrix} = \begin{bmatrix} 1 \\ 0 \\ 1 \end{bmatrix}$

$\begin{bmatrix} 1 & 1 \\ 1 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} 1 \\ 0 \end{bmatrix} = \begin{bmatrix} 1 \\ 1 \\ 0 \end{bmatrix}$ $\begin{bmatrix} 1 & 1 \\ 1 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} 1 \\ 1 \end{bmatrix} = \begin{bmatrix} 0 \\ 1 \\ 1 \end{bmatrix}$

(c)

Bit sistemático	Código simplex
0 0	$\begin{bmatrix} 0 & 0 & 0 \end{bmatrix}$
0 1	$\begin{bmatrix} 1 & 0 & 1 \end{bmatrix}$
1 0	$\begin{bmatrix} 1 & 1 & 0 \end{bmatrix}$
1 1	$\begin{bmatrix} 0 & 1 & 1 \end{bmatrix}$

Supressão(a)

Supressão(b)

FIGURA 4

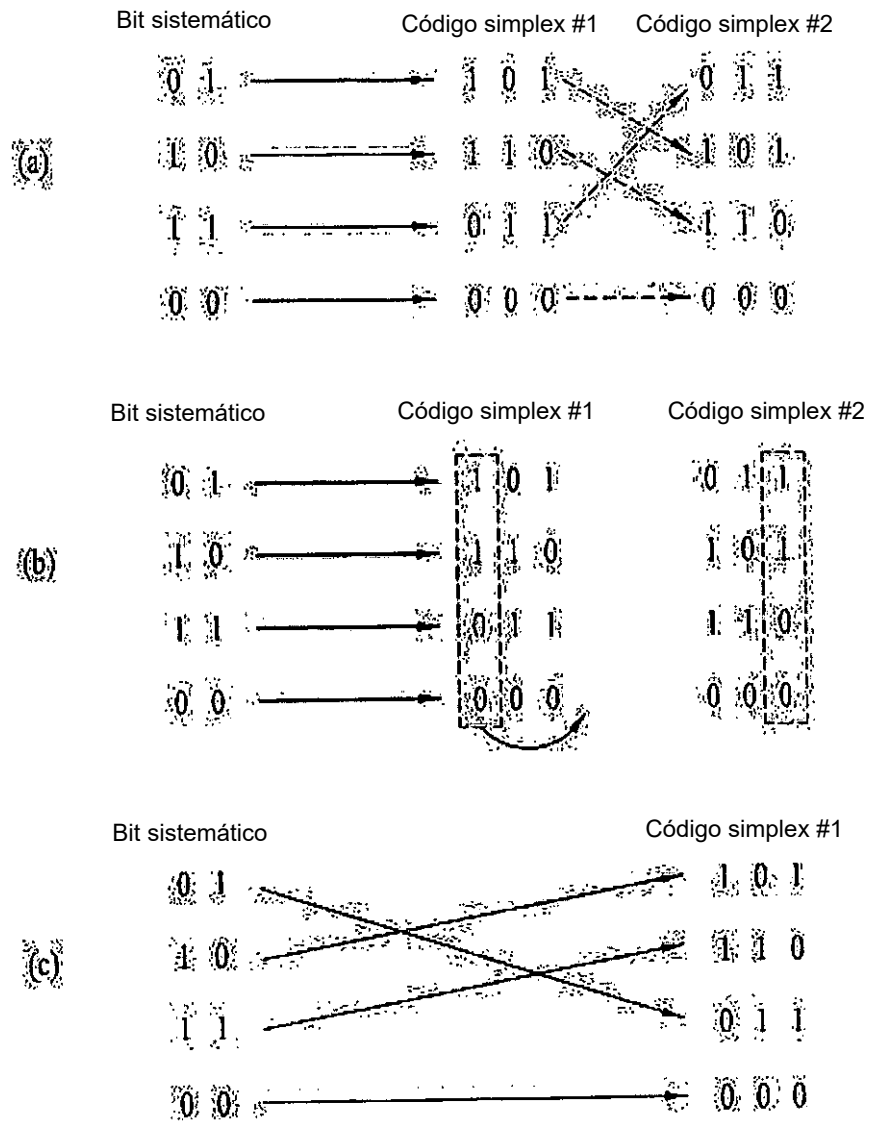
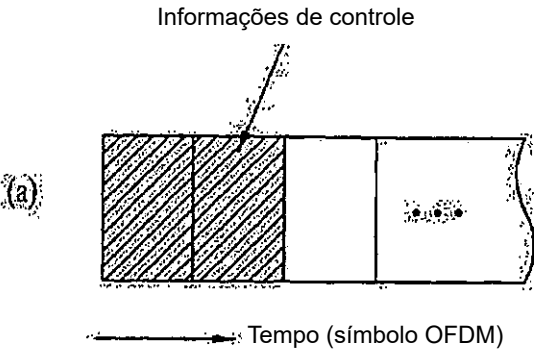


FIGURA 5



Símbolo OFDM	Código (32,2)
1	0 1 1 0 1 1 0 1 1 0 1 1 0 1 1 0 1 1 0 1 1 0 1 1 0 1 1 0 1 1 0 1
2	1 0 1 1 0 1 1 0 1 1 0 1 1 0 1 1 0 1 1 0 1 1 0 1 1 0 1 1 0 1 1 0
3	1 1 0 1 1 0 1 1 0 1 1 0 1 1 0 1 1 0 1 1 0 1 1 0 1 1 0 1 1 0 1 1