



(12)发明专利

(10)授权公告号 CN 106603191 B

(45)授权公告日 2019.07.12

(21)申请号 201510664934.X

(22)申请日 2015.10.15

(65)同一申请的已公布的文献号
申请公布号 CN 106603191 A

(43)申请公布日 2017.04.26

(73)专利权人 普天信息技术有限公司
地址 100080 北京市海淀区海淀北二街6号

(72)发明人 池连刚 鲁智

(74)专利代理机构 北京德琦知识产权代理有限公司 11018
代理人 王民盛 王丽琴

(51)Int.Cl.
H04L 1/00(2006.01)

(56)对比文件

CN 104604152 A,2015.05.06,
CN 104124987 A,2014.10.29,
CN 102356554 A,2012.02.15,
WO 2008102978 A1,2008.08.28,
US 2008109618 A1,2008.05.08,

审查员 袁欣

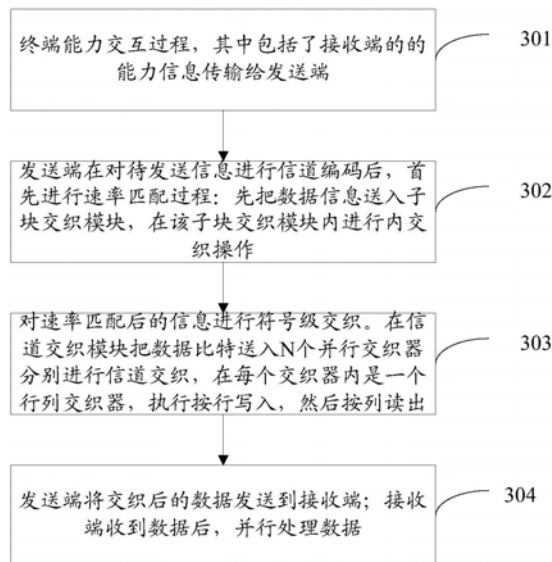
权利要求书3页 说明书7页 附图2页

(54)发明名称

一种基于并行处理的分块交织方法及装置

(57)摘要

本申请公开了一种基于并行处理的分块交织方法,包括:将接收端的并行接收的处理器个数N发送到发送端;发送端在进行码块级联后,进入N个信道交织模块进行分块交织操作;发送端将交织后的数据发送到接收端;接收端收到数据后,并行处理数据。本申请还公开了一种基于并行处理的分块交织装置。通过应用本申请方案,接收端可以采用多个信号处理器并行解调数据,达到降低时延的需要。



1. 一种基于并行处理的分块交织方法,其特征在于,包括:

A、将接收端的并行接收的处理器个数 N 发送到发送端;

B、发送端在进行码块级联后,进入 N 个信道交织模块进行分块交织操作;

C、发送端将交织后的数据发送到接收端;接收端收到数据后,并行处理数据;

其中,所述步骤B包括:

B1、发送端在对待发送信息进行信道编码后,首先进行速率匹配过程:先把数据信息送入子块交织模块,在该子块交织模块内进行内交织操作;

B2、对速率匹配后的信息进行符号级交织:在信道交织模块把数据比特送入 N 个并行交织器分别进行信道交织,在每个交织器内是一个行列交织器,执行按行写入,然后按列读出;

步骤B1包括:

B1-1:确定内交织的交织深度,即交织矩阵的列数为 C_{subblock} ,行数为 R_{subblock} ,数据比特数 D 需满足 $D \leq (R_{\text{subblock}} \times C_{\text{subblock}})$;如果 $(R_{\text{subblock}} \times C_{\text{subblock}}) > D$,那么将在矩阵起始部分填充 $N_D = (R_{\text{subblock}} \times C_{\text{subblock}} - D)$ 个伪比特,填充的伪比特为 $y_k = \langle \text{NULL} \rangle$, $k = 0, 1, \dots, N_D - 1$;

B1-2:按每行从左到右的顺序,每行及每列从0开始把数据写入矩阵,如下所示:

$$\begin{bmatrix} y_0 & y_1 & y_2 & \cdots & y_{C_{\text{subblock}}-1} \\ y_{C_{\text{subblock}}} & y_{C_{\text{subblock}}+1} & y_{C_{\text{subblock}}+2} & \cdots & y_{2C_{\text{subblock}}-1} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ y_{(R_{\text{subblock}}-1) \times C_{\text{subblock}}} & y_{(R_{\text{subblock}}-1) \times C_{\text{subblock}}+1} & y_{(R_{\text{subblock}}-1) \times C_{\text{subblock}}+2} & \cdots & y_{(R_{\text{subblock}} \times C_{\text{subblock}}-1)} \end{bmatrix};$$

B1-3:按照选定的内部列交织模式 $\langle P(j) \rangle_{j \in \{0, 1, \dots, C_{\text{subblock}}-1\}}$ 在该矩阵内部进行列交织,交织后的数据表示为:交织后如下所示

$$\begin{bmatrix} y_{P(0)} & y_{P(1)} & y_{P(2)} & \cdots & y_{P(C_{\text{subblock}}-1)} \\ y_{P(0)+C_{\text{subblock}}} & y_{P(1)+C_{\text{subblock}}} & y_{P(2)+C_{\text{subblock}}} & \cdots & y_{P(C_{\text{subblock}}-1)+C_{\text{subblock}}} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ y_{P(0)+(R_{\text{subblock}}-1) \times C_{\text{subblock}}} & y_{P(1)+(R_{\text{subblock}}-1) \times C_{\text{subblock}}} & y_{P(2)+(R_{\text{subblock}}-1) \times C_{\text{subblock}}} & \cdots & y_{P(C_{\text{subblock}}-1)+(R_{\text{subblock}}-1) \times C_{\text{subblock}}} \end{bmatrix};$$

B1-4:按列从0行,0列开始进行比特收集和选择,跳过填充的伪比特,完成速率匹配过程。

2. 根据权利要求1所述的方法,其特征在于,步骤A是在终端能力交互过程中进行。

3. 根据权利要求1所述的方法,其特征在于,所述内部列交织模式为 $\langle P(0), P(1), \dots, P(C_{\text{subblock}}-1) \rangle = \langle 0, 16, 8, 24, 4, 20, 12, 28, 2, 18, 10, 26, 6, 22, 14, 30, 1, 17, 9, 25, 5, 21, 13, 29, 3, 19, 11, 27, 7, 23, 15, 31 \rangle$ 。

4. 根据权利要求1所述的方法,其特征在于,步骤B2进一步包括:

B2-1:确定信道交织的交织深度,即交织矩阵的列数为 $C_{\text{mux}} = N_{\text{symb}}^{\text{PUSCH}}$,那么矩阵的列从左到右为 $0, 1, \dots, C_{\text{mux}}-1$,其中 $N_{\text{symb}}^{\text{PUSCH}}$ 为数据所占的符号数;确定矩阵的行数为 $R_{\text{mux}} = (H_{\text{total}} \cdot Q_m \cdot N_L) / C_{\text{mux}}$,其中 H_{total} 为每层调制符号数, Q_m 为调制阶数, N_L 为传输的层数;

B2-2:定义 $R'_{\text{mux}} = R_{\text{mux}} / (Q_m \cdot N_L)$,矩阵的行从上到下计数为 $0, 1, 2, \dots, R_{\text{mux}}-1$;按照 $(Q_m \cdot$

N_L) 个集合把数据输入向量写进矩阵 ($R_{mux} \times C_{mux}$), 按照从0列开始, 从0行到 ($Q_m \cdot N_L - 1$) 行增加的顺序, 如下所示

$$\begin{bmatrix} \underline{y}_0 & \underline{y}_1 & \underline{y}_2 & \cdots & \underline{y}_{C_{mux}-1} \\ \underline{y}_{C_{mux}} & \underline{y}_{C_{mux}+1} & \underline{y}_{C_{mux}+2} & \cdots & \underline{y}_{2C_{mux}-1} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \underline{y}_{(R'_{mux}-1) \times C_{mux}} & \underline{y}_{(R'_{mux}-1) \times C_{mux}+1} & \underline{y}_{(R'_{mux}-1) \times C_{mux}+2} & \cdots & \underline{y}_{(R'_{mux}-1) \times C_{mux}-1} \end{bmatrix};$$

B2-3: 从矩阵 ($R_{mux} \times C_{mux}$) 按列读出交织输出比特, 信道交织后的比特为 $h_0, h_1, h_2, \dots, h_{H-1}$, H 为包含多个层传输的数据的矩阵的列数。

5. 一种基于并行处理的分块交织装置, 其特征在于, 该装置位于发送端, 包括:

接收模块, 用于接收接收端能力信息, 其中包括接收端的并行接收的处理器个数 N ;

子道交织模块, 用于对信道编码后的待发送信息进行内交织操作, 并输出内交织操作后的信息;

信道交织模块, 用于根据所述处理器个数 N , 使能 N 个并行交织器; 将子道交织模块输出的信息送入所述使能的并行交织器分别进行信道交织, 在每个交织器内是一个行列交织器, 执行按行写入, 然后按列读出;

其中, 子道交织模块包括:

内交织深度确定单元, 用于确定内交织的交织深度, 即交织矩阵的列数为 $C_{subblock}$, 行数为 $R_{subblock}$, 数据比特数 D 需满足 $D \leq (R_{subblock} \times C_{subblock})$; 如果 $(R_{subblock} \times C_{subblock}) > D$, 那么将在矩阵起始部分填充 $N_D = (R_{subblock} \times C_{subblock} - D)$ 个伪比特, 填充的伪比特为 $y_k = \langle \text{NULL} \rangle$, $k = 0, 1, \dots, N_D - 1$;

矩阵写入单元, 用于按每行从左到右的顺序, 每行及每列从0开始把数据写入矩阵, 如下所示:

$$\begin{bmatrix} y_0 & y_1 & y_2 & \cdots & y_{C_{subblock}-1} \\ y_{C_{subblock}} & y_{C_{subblock}+1} & y_{C_{subblock}+2} & \cdots & y_{2C_{subblock}-1} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ y_{(R_{subblock}-1) \times C_{subblock}} & y_{(R_{subblock}-1) \times C_{subblock}+1} & y_{(R_{subblock}-1) \times C_{subblock}+2} & \cdots & y_{(R_{subblock}-1) \times C_{subblock}-1} \end{bmatrix};$$

列交织单元, 用于按照选定的内部列交织模式 $\langle P(j) \rangle_{j \in \{0, 1, \dots, C_{subblock}-1\}}$ 在该矩阵内部进行列交织, 交织后的数据表示为: 交织后如下所示

$$\begin{bmatrix} y_{P(0)} & y_{P(1)} & y_{P(2)} & \cdots & y_{P(C_{subblock}-1)} \\ y_{P(0)+C_{subblock}} & y_{P(1)+C_{subblock}} & y_{P(2)+C_{subblock}} & \cdots & y_{P(C_{subblock}-1)+C_{subblock}} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ y_{P(0)+(R_{subblock}-1) \times C_{subblock}} & y_{P(1)+(R_{subblock}-1) \times C_{subblock}} & y_{P(2)+(R_{subblock}-1) \times C_{subblock}} & \cdots & y_{P(C_{subblock}-1)+(R_{subblock}-1) \times C_{subblock}} \end{bmatrix};$$

输出单元, 用于按列从0行, 0列开始进行比特收集和选择, 跳过填充的伪比特, 输出所收集和选择的比特。

6. 根据权利要求5所述的装置, 其特征在于, 所述内部列交织模式为 $\langle P(0), P(1), \dots, P(C_{subblock}-1) \rangle = \langle 0, 16, 8, 24, 4, 20, 12, 28, 2, 18, 10, 26, 6, 22, 14, 30, 1, 17, 9, 25, 5, 21, 13, 29, 3, 19, 11, 27, 7, 23, 15, 31 \rangle$ 。

7. 根据权利要求5所述的装置,其特征在于,所述信道交织模块中的每一个并行交织器进一步包括:

信道交织深度确定单元,用于确定信道交织的交织深度,即交织矩阵的列数为 $C_{mux} = N_{\text{symbol}}^{\text{PUSCH}}$,那么矩阵的列从左到右为 $0, 1, \dots, C_{mux}-1$,其中 $N_{\text{symbol}}^{\text{PUSCH}}$ 为数据所占的符号数;确定矩阵的行数为 $R_{mux} = (H_{\text{total}} \cdot Q_m \cdot N_L) / C_{mux}$,其中 H_{total} 为每层调制符号数, Q_m 为调制阶数, N_L 为传输的层数;

矩阵写入单元,用于定义 $R'_{mux} = R_{mux} / (Q_m \cdot N_L)$,矩阵的行从上到下计数为 $0, 1, 2, \dots, R_{mux}-1$;按照 $(Q_m \cdot N_L)$ 个集合把数据输入向量写进矩阵 $(R_{mux} \times C_{mux})$,按照从0列开始,从0行到 $(Q_m \cdot N_L-1)$ 行增加的顺序,如下所示

$$\begin{bmatrix} \underline{y}_0 & \underline{y}_1 & \underline{y}_2 & \cdots & \underline{y}_{C_{mux}-1} \\ \underline{y}_{C_{mux}} & \underline{y}_{C_{mux}+1} & \underline{y}_{C_{mux}+2} & \cdots & \underline{y}_{2C_{mux}-1} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \underline{y}_{(R'_{mux}-1) \times C_{mux}} & \underline{y}_{(R'_{mux}-1) \times C_{mux}+1} & \underline{y}_{(R'_{mux}-1) \times C_{mux}+2} & \cdots & \underline{y}_{(R'_{mux}-1) \times C_{mux}-1} \end{bmatrix};$$

输出单元,用于从矩阵 $(R_{mux} \times C_{mux})$ 按列读出交织输出比特,信道交织后的比特为 $h_0, h_1, h_2, \dots, h_{H-1}$, H 为包含多个层传输的数据的矩阵的列数。

一种基于并行处理的分块交织方法及装置

技术领域

[0001] 本申请涉及移动通信技术领域,尤其涉及一种基于并行处理的分块交织方法及装置。

背景技术

[0002] 低时延和高可靠通信是未来5G通信的一个研究热点。一个典型的应用场景是设备之间能实时、快速地相互交换动态信息,对这些信息交互“严格实时”的要求主要体现在:时延须控制在毫秒级并且数据交换的频率要达到每秒数十次。特别地,对于低时延场景,对传输时延有了更严格的要求,例如单数据包用户面空口单次传输时延不大于1ms。这个要求对现有技术带来了挑战。

[0003] 现有技术中,LTE系统在进行速率匹配的过程中,采用对分配的资源进行整体交织的方案,如图1所示,在进行信道编码后,把编码比特整体进行交织处理。相应地,接收端使用一个信号处理器对分配的所有资源进行串行处理。

[0004] 对于低时延高可靠场景,对接收端的处理时延有很高要求,为满足低时延要求,需要接收端采用并行处理方式,尽快完成数据解码。目前的系统没有这种机制。

发明内容

[0005] 本申请提供了一种基于并行处理的分块交织方法,接收端可以采用多个信号处理器并行解调数据,达到降低时延的需要。

[0006] 本申请实施例提供了一种基于并行处理的分块交织方法,包括:

[0007] A、将接收端的并行接收的处理器个数 N 发送到发送端;

[0008] B、发送端在进行码块级联后,进入 N 个信道交织模块进行分块交织操作;

[0009] C、发送端将交织后的数据发送到接收端;接收端收到数据后,并行处理数据。

[0010] 可选地,步骤A是在终端能力交互过程中进行。

[0011] 可选地,步骤B包括:

[0012] B1、发送端在对待发送信息进行信道编码后,首先进行速率匹配过程:先把数据信息送入子块交织模块,在该子块交织模块内进行内交织操作;

[0013] B2、对速率匹配后的信息进行符号级交织:在信道交织模块把数据比特送入 N 个并行交织器分别进行信道交织,在每个交织器内是一个行列交织器,执行按行写入,然后按列读出。

[0014] 可选地,步骤B1进一步包括:

[0015] B1-1:确定内交织的交织深度,即交织矩阵的列数为 C_{subblock} ,行数为 R_{subblock} ,数据比特数 D 需满足 $D \leq (R_{\text{subblock}} \times C_{\text{subblock}})$;如果 $(R_{\text{subblock}} \times C_{\text{subblock}}) > D$,那么将在矩阵起始部分填充 $N_D = (R_{\text{subblock}} \times C_{\text{subblock}} - D)$ 个伪比特,填充的伪比特为 $y_k = \langle \text{NULL} \rangle, k = 0, 1, \dots, N_D - 1$;

[0016] B1-2:按每行从左到右的顺序,每行及每列从0开始把数据写入矩阵,如下所示:

$$[0017] \begin{bmatrix} y_0 & y_1 & y_2 & \cdots & y_{C_{\text{subblock}}-1} \\ y_{C_{\text{subblock}}} & y_{C_{\text{subblock}}+1} & y_{C_{\text{subblock}}+2} & \cdots & y_{2C_{\text{subblock}}-1} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ y_{(R_{\text{subblock}}-1) \times C_{\text{subblock}}} & y_{(R_{\text{subblock}}-1) \times C_{\text{subblock}}+1} & y_{(R_{\text{subblock}}-1) \times C_{\text{subblock}}+2} & \cdots & y_{(R_{\text{subblock}} \times C_{\text{subblock}}-1)} \end{bmatrix}$$

[0018] B1-3:按照选定的内部列交织模式 $\langle P(j) \rangle_{j \in \{0,1,\dots,C_{\text{subblock}}-1\}}$ 在该矩阵内部进行列交织,交织后的数据表示为:交织后如下所示

[0019]

$$\begin{bmatrix} y_{P(0)} & y_{P(1)} & y_{P(2)} & \cdots & y_{P(C_{\text{subblock}}-1)} \\ y_{P(0)+C_{\text{subblock}}} & y_{P(1)+C_{\text{subblock}}} & y_{P(2)+C_{\text{subblock}}} & \cdots & y_{P(C_{\text{subblock}}-1)+C_{\text{subblock}}} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ y_{P(0)+(R_{\text{subblock}}-1) \times C_{\text{subblock}}} & y_{P(1)+(R_{\text{subblock}}-1) \times C_{\text{subblock}}} & y_{P(2)+(R_{\text{subblock}}-1) \times C_{\text{subblock}}} & \cdots & y_{P(C_{\text{subblock}}-1)+(R_{\text{subblock}}-1) \times C_{\text{subblock}}} \end{bmatrix};$$

[0020] B1-4:按列从0行,0列开始进行比特收集和选择,跳过填充的伪比特,完成速率匹配过程。

[0021] 可选地,所述内部列交织模式为 $\langle P(0), P(1), \dots, P(C_{\text{subblock}}-1) \rangle = \langle 0, 16, 8, 24, 4, 20, 12, 28, 2, 18, 10, 26, 6, 22, 14, 30, 1, 17, 9, 25, 5, 21, 13, 29, 3, 19, 11, 27, 7, 23, 15, 31 \rangle$ 。

[0022] 可选地,步骤B2进一步包括:

[0023] B2-1:确定信道交织的交织深度,即交织矩阵的列数为 $C_{\text{mux}} = N_{\text{sym}}^{\text{PUSCH}}$,那么矩阵的列从左到右为 $0, 1, \dots, C_{\text{mux}}-1$,其中 $N_{\text{sym}}^{\text{PUSCH}}$ 为数据所占的符号数。确定矩阵的行数为 $R_{\text{mux}} = (H_{\text{total}} \cdot Q_m \cdot N_L) / C_{\text{mux}}$,其中 H_{total} 为每层调制符号数, Q_m 为调制阶数, N_L 为传输的层数;

[0024] B2-2:定义 $R'_{\text{mux}} = R_{\text{mux}} / (Q_m \cdot N_L)$,矩阵的行从上到下计数为 $0, 1, 2, \dots, R_{\text{mux}}-1$ 。按照 $(Q_m \cdot N_L)$ 个集合把数据输入向量写进矩阵 $(R_{\text{mux}} \times C_{\text{mux}})$,按照从0列开始,从0行到 $(Q_m \cdot N_L-1)$ 行增加的顺序,如下所示

$$[0025] \begin{bmatrix} \underline{y}_0 & \underline{y}_1 & \underline{y}_2 & \cdots & \underline{y}_{C_{\text{mux}}-1} \\ \underline{y}_{C_{\text{mux}}} & \underline{y}_{C_{\text{mux}}+1} & \underline{y}_{C_{\text{mux}}+2} & \cdots & \underline{y}_{2C_{\text{mux}}-1} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \underline{y}_{(R'_{\text{mux}}-1) \times C_{\text{mux}}} & \underline{y}_{(R'_{\text{mux}}-1) \times C_{\text{mux}}+1} & \underline{y}_{(R'_{\text{mux}}-1) \times C_{\text{mux}}+2} & \cdots & \underline{y}_{(R'_{\text{mux}} \times C_{\text{mux}}-1)} \end{bmatrix};$$

[0026] B2-3:从矩阵 $(R_{\text{mux}} \times C_{\text{mux}})$ 按列读出交织输出比特,信道交织后的比特为 $h_0, h_1, h_2, \dots, h_{H-1}$ 。

[0027] 本申请还提供了一种基于并行处理的分块交织装置,该装置位于发送端,包括:

[0028] 接收模块,用于接收接收端能力信息,其中包括接收端的并行接收的处理器个数 N

[0029] 子道交织模块,用于对信道编码后的待发送信息进行内交织操作,并输出内交织操作后的信息;

[0030] 信道交织模块,用于根据所述处理器个数 N ,使能 N 个并行交织器;将子道交织模块输出的信息送入所述使能的并行交织器分别进行信道交织,在每个交织器内是一个行列交织器,执行按行写入,然后按列读出。

[0031] 可选地,子道交织模块包括:

[0032] 内交织深度确定单元,用于确定内交织的交织深度,即交织矩阵的列数为 C_{subblock} ,行数为 R_{subblock} ,数据比特数 D 需满足 $D \leq (R_{\text{subblock}} \times C_{\text{subblock}})$;如果 $(R_{\text{subblock}} \times C_{\text{subblock}}) > D$,那么将在矩阵起始部分填充 $N_D = (R_{\text{subblock}} \times C_{\text{subblock}} - D)$ 个伪比特,填充的伪比特为 $y_k = \langle \text{NULL} \rangle, k=0, 1, \dots, N_D-1$;

[0033] 矩阵写入单元,用于按每行从左到右的顺序,每行及每列从0开始把数据写入矩阵,如下所示:

$$[0034] \begin{bmatrix} y_0 & y_1 & y_2 & \cdots & y_{C_{\text{subblock}}-1} \\ y_{C_{\text{subblock}}} & y_{C_{\text{subblock}}+1} & y_{C_{\text{subblock}}+2} & \cdots & y_{2C_{\text{subblock}}-1} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ y_{(R_{\text{subblock}}-1) \times C_{\text{subblock}}} & y_{(R_{\text{subblock}}-1) \times C_{\text{subblock}}+1} & y_{(R_{\text{subblock}}-1) \times C_{\text{subblock}}+2} & \cdots & y_{(R_{\text{subblock}} \times C_{\text{subblock}}-1)} \end{bmatrix}$$

[0035] 列交织单元,用于按照选定的内部列交织模式 $\{P(j)\}_{j \in \{0,1,\dots,C_{\text{subblock}}-1\}}$ 在该矩阵内部进行列交织,交织后的数据表示为:交织后如下所示

[0036]

$$\begin{bmatrix} y_{P(0)} & y_{P(1)} & y_{P(2)} & \cdots & y_{P(C_{\text{subblock}}-1)} \\ y_{P(0)+C_{\text{subblock}}} & y_{P(1)+C_{\text{subblock}}} & y_{P(2)+C_{\text{subblock}}} & \cdots & y_{P(C_{\text{subblock}}-1)+C_{\text{subblock}}} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ y_{P(0)+(R_{\text{subblock}}-1) \times C_{\text{subblock}}} & y_{P(1)+(R_{\text{subblock}}-1) \times C_{\text{subblock}}} & y_{P(2)+(R_{\text{subblock}}-1) \times C_{\text{subblock}}} & \cdots & y_{P(C_{\text{subblock}}-1)+(R_{\text{subblock}}-1) \times C_{\text{subblock}}} \end{bmatrix};$$

[0037] 输出单元,用于按列从0行,0列开始进行比特收集和选择,跳过填充的伪比特,输出所收集和选择的比特。

[0038] 可选地,所述内部列交织模式为 $\langle P(0), P(1), \dots, P(C_{\text{subblock}}-1) \rangle = \langle 0, 16, 8, 24, 4, 20, 12, 28, 2, 18, 10, 26, 6, 22, 14, 30, 1, 17, 9, 25, 5, 21, 13, 29, 3, 19, 11, 27, 7, 23, 15, 31 \rangle$ 。

[0039] 可选地,所述信道交织模块中的每一个并行交织器进一步包括:

[0040] 信道交织深度确定单元,用于确定信道交织的交织深度,即交织矩阵的列数为 $C_{\text{mux}} = N_{\text{sym}}^{\text{PUSCH}}$,那么矩阵的列从左到右为 $0, 1, \dots, C_{\text{mux}}-1$,其中 $N_{\text{sym}}^{\text{PUSCH}}$ 为数据所占的符号数。

确定矩阵的行数为 $R_{\text{mux}} = (H_{\text{total}} \cdot Q_m \cdot N_L) / C_{\text{mux}}$,其中 H_{total} 为每层调制符号数, Q_m 为调制阶数, N_L 为传输的层数;

[0041] 矩阵写入单元,用于定义 $R'_{\text{mux}} = R_{\text{mux}} / (Q_m \cdot N_L)$,矩阵的行从上到下计数为 $0, 1, 2, \dots, R_{\text{mux}}-1$ 。按照 $(Q_m \cdot N_L)$ 个集合把数据输入向量写进矩阵 $(R_{\text{mux}} \times C_{\text{mux}})$,按照从0列开始,从0行到 $(Q_m \cdot N_L-1)$ 行增加的顺序,如下所示

$$[0042] \begin{bmatrix} y_0 & y_1 & y_2 & \cdots & y_{C_{\text{mux}}-1} \\ y_{C_{\text{mux}}} & y_{C_{\text{mux}}+1} & y_{C_{\text{mux}}+2} & \cdots & y_{2C_{\text{mux}}-1} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ y_{(R'_{\text{mux}}-1) \times C_{\text{mux}}} & y_{(R'_{\text{mux}}-1) \times C_{\text{mux}}+1} & y_{(R'_{\text{mux}}-1) \times C_{\text{mux}}+2} & \cdots & y_{(R'_{\text{mux}} \times C_{\text{mux}}-1)} \end{bmatrix};$$

[0043] 输出单元,用于从矩阵 $(R_{\text{mux}} \times C_{\text{mux}})$ 按列读出交织输出比特,信道交织后的比特为 $h_0, h_1, h_2, \dots, h_{H-1}$ 。

[0044] 从以上技术方案可以看出,在终端能力交互过程中增加接收端能力信息(并行接收的信号处理器的个数 N),对于传输使用的资源块,根据接收端能力信息,发送端在进行码

块级联后,进入N个信道交织模块进行分块交织操作;接收端收到数据后可以并行处理数据,降低数据的处理时延。

附图说明

[0045] 图1为现有技术中LTE系统的编码交织处理示意图;

[0046] 图2为根据本申请实施例方案,且接收端并行接收的信号处理器个数为2时的编码交织处理示意图;

[0047] 图3为本申请实施例提供的基于并行处理的分块交织方法处理流程示意图。

具体实施方式

[0048] 本申请提供了一种基于并行处理的分块交织方法,在终端能力交互过程中增加接收端能力信息(并行接收的信号处理器的个数N),对于传输使用的资源块,根据接收端能力信息,发送端在进行码块级联后,进入N个信道交织模块进行分块交织操作;接收端收到数据后可以并行处理数据,降低数据的处理时延。例如,若接收端并行接收的信号处理器个数为2,编码交织处理如图2所示。

[0049] 为使本申请技术方案的技术原理、特点以及技术效果更加清楚,以下结合具体实施例对本申请技术方案进行详细阐述。

[0050] 本申请实施例提供的基于并行处理的分块交织方法处理流程如图3所示,包括如下步骤:

[0051] 步骤301:终端能力交互过程,其中包括了接收端的能力信息传输给发送端。

[0052] 根据本申请的一个实施例,所述接收端能力信息为并行信号处理器个数。

[0053] 步骤302:发送端在对待发送信息进行信道编码后,首先进行速率匹配过程:先把数据信息送入子块交织模块,在该子块交织模块内进行内交织操作。

[0054] 本申请另一实施例给出了内交织操作的具体实现过程,包括如下子步骤:

[0055] 子步骤302-1:确定交织的交织深度,即交织矩阵的列数为 C_{subblock} ,行数为 R_{subblock} ,数据比特数D需满足 $D \leq (R_{\text{subblock}} \times C_{\text{subblock}})$,如果 $(R_{\text{subblock}} \times C_{\text{subblock}}) > D$,那么将在矩阵起始部分填充 $N_D = (R_{\text{subblock}} \times C_{\text{subblock}} - D)$ 个伪比特,填充的伪比特为 $y_k = \langle \text{NULL} \rangle$, $k = 0, 1, \dots, N_D - 1$ 。

[0056] 子步骤302-2:按每行从左到右的顺序,每行及每列从0开始把数据写入矩阵,如下所示:

$$[0057] \begin{bmatrix} y_0 & y_1 & y_2 & \cdots & y_{C_{\text{subblock}}-1} \\ y_{C_{\text{subblock}}} & y_{C_{\text{subblock}}+1} & y_{C_{\text{subblock}}+2} & \cdots & y_{2C_{\text{subblock}}-1} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ y_{(R_{\text{subblock}}-1) \times C_{\text{subblock}}} & y_{(R_{\text{subblock}}-1) \times C_{\text{subblock}}+1} & y_{(R_{\text{subblock}}-1) \times C_{\text{subblock}}+2} & \cdots & y_{(R_{\text{subblock}} \times C_{\text{subblock}}-1)} \end{bmatrix}$$

[0058] 子步骤302-3:按照一个内部列交织模式在该矩阵内部进行列交织。

[0059] 该内部列交织模式的具体形式可以有多种,本申请对此不作限定。例如可用于本申请的一个32位的内部列交织模式为 $\langle P(j) \rangle_{j \in \{0, 1, \dots, C_{\text{subblock}}-1\}}$ 。内部列交织模式

$\langle P(j) \rangle_{j \in \{0,1,\dots,C_{subblock}-1\}}$ 具体如表1所示:

列数 $C_{subblock}$	内部列交织模式 $\langle P(0), P(1), \dots, P(C_{subblock}-1) \rangle$
[0060] 32	$\langle 0, 16, 8, 24, 4, 20, 12, 28, 2, 18, 10, 26, 6, 22, 14, 30, 1, 17, 9, 25, 5, 21, 13, 29, 3, 19, 11, 27, 7, 23, 15, 31 \rangle$

[0061] 表1

[0062] 交织后如下所示

$$[0063] \begin{bmatrix} y_{P(0)} & y_{P(1)} & y_{P(2)} & \dots & y_{P(C_{subblock}-1)} \\ y_{P(0)+C_{subblock}} & y_{P(1)+C_{subblock}} & y_{P(2)+C_{subblock}} & \dots & y_{P(C_{subblock}-1)+C_{subblock}} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ y_{P(0)+(R_{subblock}-1) \times C_{subblock}} & y_{P(1)+(R_{subblock}-1) \times C_{subblock}} & y_{P(2)+(R_{subblock}-1) \times C_{subblock}} & \dots & y_{P(C_{subblock}-1)+(R_{subblock}-1) \times C_{subblock}} \end{bmatrix}$$

[0064] 子步骤302-4:按列从0行,0列开始进行比特收集和选择,跳过填充的伪比特,完成速率匹配过程。

[0065] 步骤303:对速率匹配后的信息进行符号级交织。在信道交织模块把数据比特送入N个并行交织器分别进行信道交织,在每个交织器内是一个行列交织器,执行按行写入,然后按列读出。

[0066] 本申请的一个实施例中给出的一个交织器内部的交织过程如下:

[0067] 对于每一个块中待交织的比特流输入表示为 $\underline{g}_0, \underline{g}_1, \underline{g}_2, \underline{g}_3, \dots, \underline{g}_{H-1}$, H为包含多个层传输的数据(可能包含潜在控制信息)的矩阵的列数,输出交织后的比特序列产生过程包括如下子步骤:

[0068] 子步骤303-1:确定交织的交织深度,即交织矩阵的列数为 $C_{mux} = N_{\text{symb}}^{\text{PUSCH}}$, 那么矩阵的列从左到右为 $0, 1, \dots, C_{mux}-1$, 其中 $N_{\text{symb}}^{\text{PUSCH}}$ 为数据所占的符号数。确定矩阵的行数为 $R_{mux} = (H_{\text{total}} \cdot Q_m \cdot N_L) / C_{mux}$, 其中 H_{total} 为每层调制符号数, Q_m 为调制阶数, N_L 为传输的层数。

[0069] 子步骤303-2:定义 $R'_{mux} = R_{mux} / (Q_m \cdot N_L)$, 矩阵的行从上到下计数为 $0, 1, 2, \dots, R_{mux}-1$ 。按照 $(Q_m \cdot N_L)$ 个集合把数据输入向量写进矩阵 $(R_{mux} \times C_{mux})$, 按照从0列开始,从0行到 $(Q_m \cdot N_L-1)$ 行增加的顺序,如下所示

$$[0070] \begin{bmatrix} \underline{y}_0 & \underline{y}_1 & \underline{y}_2 & \dots & \underline{y}_{C_{mux}-1} \\ \underline{y}_{C_{mux}} & \underline{y}_{C_{mux}+1} & \underline{y}_{C_{mux}+2} & \dots & \underline{y}_{2C_{mux}-1} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \underline{y}_{(R'_{mux}-1) \times C_{mux}} & \underline{y}_{(R'_{mux}-1) \times C_{mux}+1} & \underline{y}_{(R'_{mux}-1) \times C_{mux}+2} & \dots & \underline{y}_{(R'_{mux} \times C_{mux}-1)} \end{bmatrix}$$

[0071] 子步骤303-3:交织输出比特从矩阵 $(R_{mux} \times C_{mux})$ 按列读出,信道交织后的比特为 $h_0, h_1, h_2, \dots, h_{H-1}$ 。

[0072] 通过这种并行交织形成多个独立的信道交织部分,接收端可以同时多个并行的数据单元进行独立解码。

[0073] 步骤304:发送端将交织后的数据发送到接收端;接收端收到数据后,并行处理数据。

[0074] 本申请实施例还提供了一种基于并行处理的分块交织装置,该装置位于发送端,

包括:

[0075] 接收模块,用于接收接收端能力信息,其中包括接收端的并行接收的处理器个数N

[0076] 子道交织模块,用于对信道编码后的待发送信息进行内交织操作,并输出内交织操作后的信息;

[0077] 信道交织模块,用于根据所述处理器个数N,使能N个并行交织器;将子道交织模块输出的信息送入所述使能的并行交织器分别进行信道交织,在每个交织器内是一个行列交织器,执行按行写入,然后按列读出。

[0078] 可选地,子道交织模块包括:

[0079] 内交织深度确定单元,用于确定内交织的交织深度,即交织矩阵的列数为 C_{subblock} ,行数为 R_{subblock} ,数据比特数D需满足 $D \leq (R_{\text{subblock}} \times C_{\text{subblock}})$;如果 $(R_{\text{subblock}} \times C_{\text{subblock}}) > D$,那么将在矩阵起始部分填充 $N_D = (R_{\text{subblock}} \times C_{\text{subblock}} - D)$ 个伪比特,填充的伪比特为 $y_k = \langle \text{NULL} \rangle, k = 0, 1, \dots, N_D - 1$;

[0080] 矩阵写入单元,用于按每行从左到右的顺序,每行及每列从0开始把数据写入矩阵,如下所示:

$$[0081] \begin{bmatrix} y_0 & y_1 & y_2 & \cdots & y_{C_{\text{subblock}}-1} \\ y_{C_{\text{subblock}}} & y_{C_{\text{subblock}}+1} & y_{C_{\text{subblock}}+2} & \cdots & y_{2C_{\text{subblock}}-1} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ y_{(R_{\text{subblock}}-1) \times C_{\text{subblock}}} & y_{(R_{\text{subblock}}-1) \times C_{\text{subblock}}+1} & y_{(R_{\text{subblock}}-1) \times C_{\text{subblock}}+2} & \cdots & y_{(R_{\text{subblock}} \times C_{\text{subblock}}-1)} \end{bmatrix}$$

[0082] 列交织单元,用于按照选定的内部列交织模式 $\langle P(j) \rangle_{j \in \{0, 1, \dots, C_{\text{subblock}}-1\}}$ 在该矩阵内部进行列交织,交织后的数据表示为:交织后如下所示

[0083]

$$\begin{bmatrix} y_{P(0)} & y_{P(1)} & y_{P(2)} & \cdots & y_{P(C_{\text{subblock}}-1)} \\ y_{P(0)+C_{\text{subblock}}} & y_{P(1)+C_{\text{subblock}}} & y_{P(2)+C_{\text{subblock}}} & \cdots & y_{P(C_{\text{subblock}}-1)+C_{\text{subblock}}} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ y_{P(0)+(R_{\text{subblock}}-1) \times C_{\text{subblock}}} & y_{P(1)+(R_{\text{subblock}}-1) \times C_{\text{subblock}}} & y_{P(2)+(R_{\text{subblock}}-1) \times C_{\text{subblock}}} & \cdots & y_{P(C_{\text{subblock}}-1)+(R_{\text{subblock}}-1) \times C_{\text{subblock}}} \end{bmatrix};$$

[0084] 输出单元,用于按列从0行,0列开始进行比特收集和选择,跳过填充的伪比特,输出所收集和选择的比特。

[0085] 可选地,所述内部列交织模式为 $\langle P(0), P(1), \dots, P(C_{\text{subblock}}-1) \rangle = \langle 0, 16, 8, 24, 4, 20, 12, 28, 2, 18, 10, 26, 6, 22, 14, 30, 1, 17, 9, 25, 5, 21, 13, 29, 3, 19, 11, 27, 7, 23, 15, 31 \rangle$ 。

[0086] 可选地,所述信道交织模块中的每一个并行交织器进一步包括:

[0087] 信道交织深度确定单元,用于确定信道交织的交织深度,即交织矩阵的列数为 $C_{\text{mux}} = N_{\text{symb}}^{\text{PUSCH}}$,那么矩阵的列从左到右为 $0, 1, \dots, C_{\text{mux}}-1$,其中 $N_{\text{symb}}^{\text{PUSCH}}$ 为数据所占的符号数。确定矩阵的行数为 $R_{\text{mux}} = (H_{\text{total}} \cdot Q_m \cdot N_L) / C_{\text{mux}}$,其中 H_{total} 为每层调制符号数, Q_m 为调制阶数, N_L 为传输的层数;

[0088] 矩阵写入单元,用于定义 $R'_{\text{mux}} = R_{\text{mux}} / (Q_m \cdot N_L)$,矩阵的行从上到下计数为 $0, 1, 2, \dots, R_{\text{mux}}-1$ 。按照 $(Q_m \cdot N_L)$ 个集合把数据输入向量写进矩阵 $(R_{\text{mux}} \times C_{\text{mux}})$,按照从0列开始,从0行到 $(Q_m \cdot N_L-1)$ 行增加的顺序,如下所示

$$[0089] \quad \begin{bmatrix} \underline{y}_0 & \underline{y}_1 & \underline{y}_2 & \cdots & \underline{y}_{C_{mux}-1} \\ \underline{y}_{C_{mux}} & \underline{y}_{C_{mux}+1} & \underline{y}_{C_{mux}+2} & \cdots & \underline{y}_{2C_{mux}-1} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \underline{y}_{(R'_{mux}-1) \times C_{mux}} & \underline{y}_{(R'_{mux}-1) \times C_{mux}+1} & \underline{y}_{(R'_{mux}-1) \times C_{mux}+2} & \cdots & \underline{y}_{(R'_{mux} \times C_{mux}-1)} \end{bmatrix};$$

[0090] 输出单元,用于从矩阵 $(R_{mux} \times C_{mux})$ 按列读出交织输出比特,信道交织后的比特为 $h_0, h_1, h_2, \dots, h_{H-1}$ 。

[0091] 应当理解,虽然本说明书是按照各个实施方式描述的,但并非每个实施方式仅包含一个独立的技术方案,说明书的这种叙述方式仅仅是为清楚起见,本领域技术人员应当将说明书作为一个整体,各实施方式中的技术方案也可以经适当组合,形成本领域技术人员可以理解的其他实施方式。

[0092] 以上所述仅为本申请的较佳实施例而已,并不用以限制本申请的保护范围,凡在本申请技术方案的精神和原则之内,所做的任何修改、等同替换、改进等,均应包含在本申请保护的范围之内。

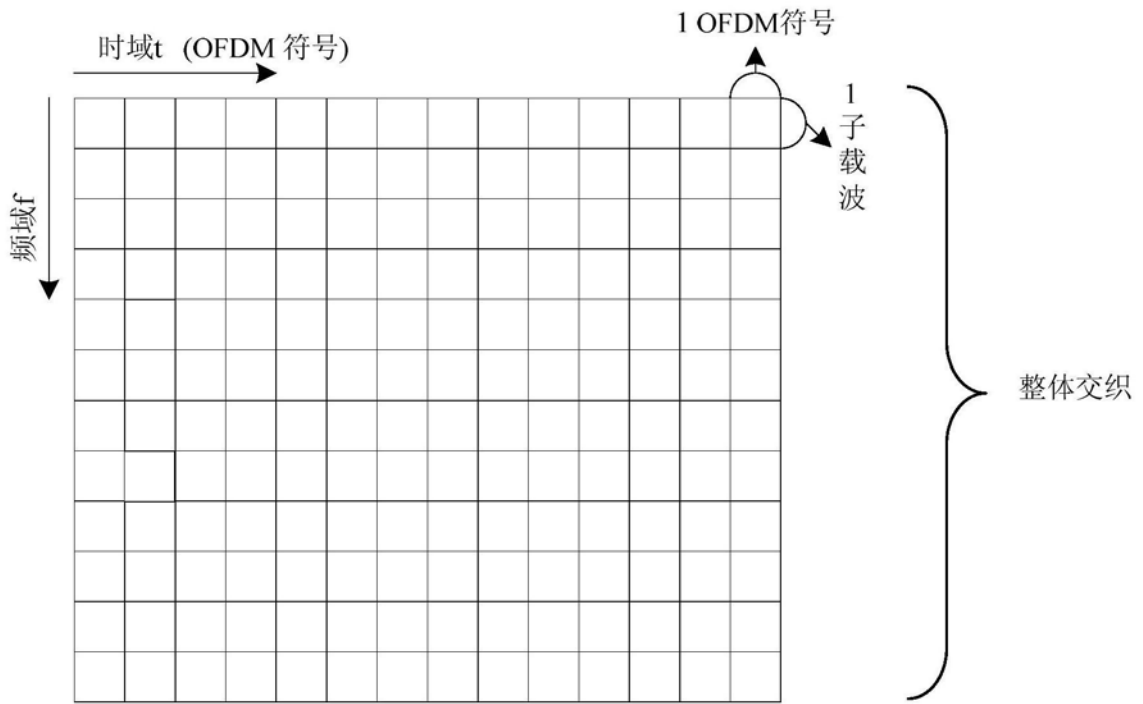


图1

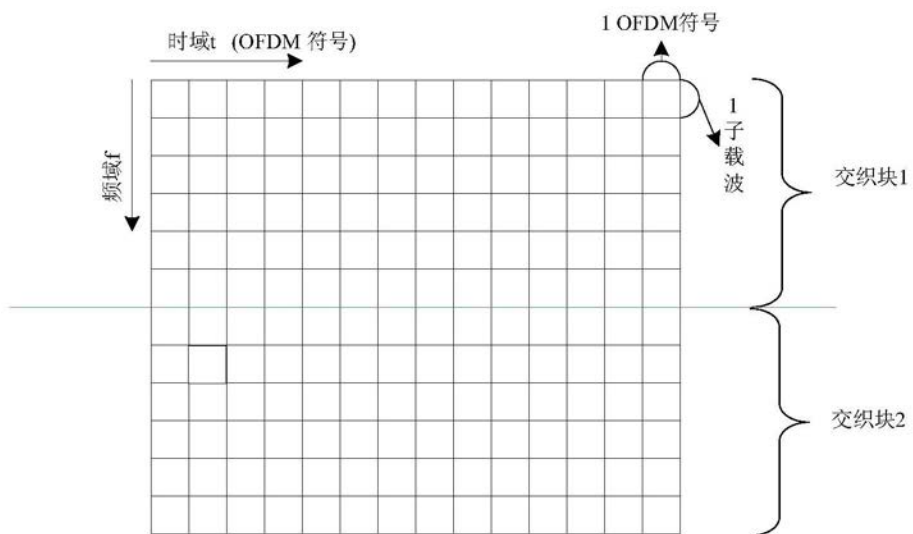


图2

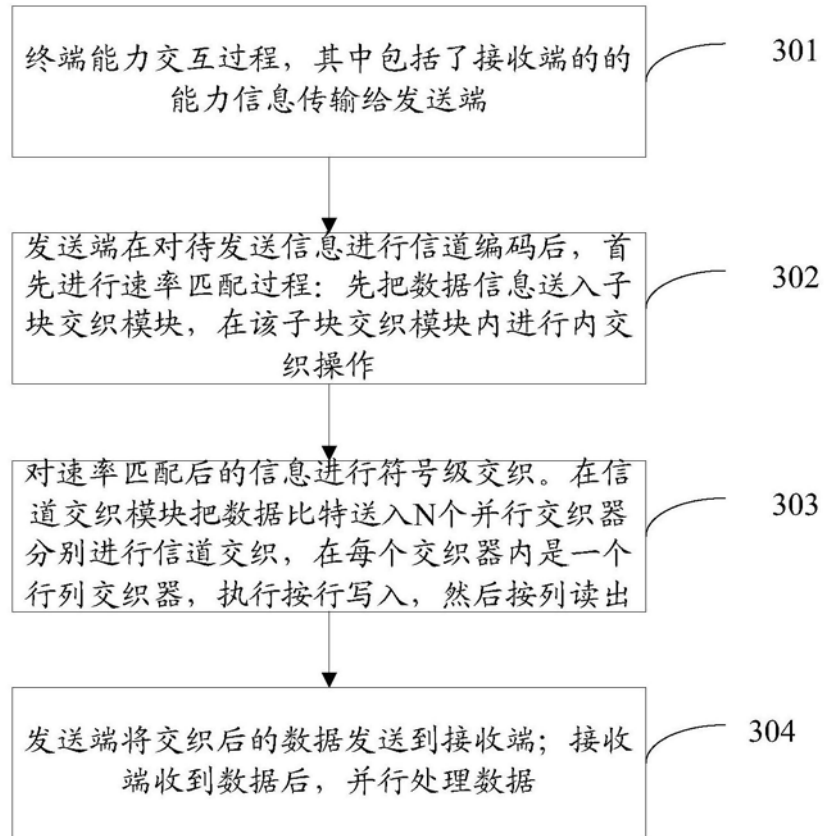


图3