



(12)发明专利

(10)授权公告号 CN 107395545 B

(45)授权公告日 2019.12.06

(21)申请号 201710676926.6

(22)申请日 2017.08.09

(65)同一申请的已公布的文献号
申请公布号 CN 107395545 A

(43)申请公布日 2017.11.24

(73)专利权人 南京邮电大学
地址 210003 江苏省南京市栖霞区文苑路9号

(72)发明人 曾桂根 方婷

(74)专利代理机构 南京知识律师事务所 32207
代理人 李吉宽

(51)Int.Cl.
H04L 27/26(2006.01)
H04L 1/00(2006.01)

(56)对比文件

CN 104917587 A,2015.09.16,
CN 105991499 A,2016.10.05,
CN 102113231 A,2011.06.29,
EP 1463255 A1,2004.09.29,

审查员 白红昌

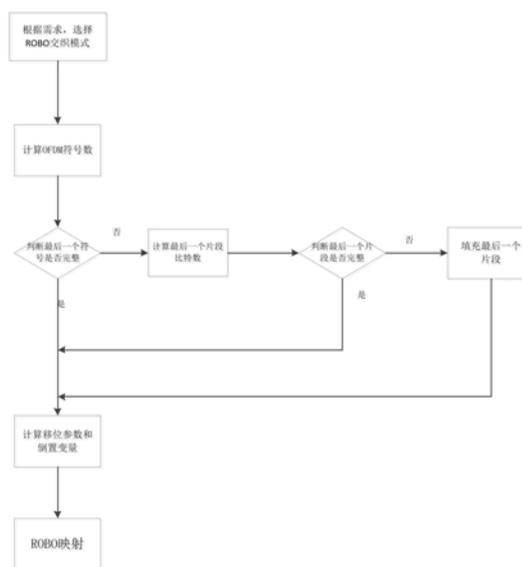
权利要求书1页 说明书8页 附图3页

(54)发明名称

一种ROBO交织技术的实现方法

(57)摘要

本发明公开了一种ROBO技术的实现方法,首先根据实际要求选择交织模式,根据参数来确定ROBO交织实际可用的载波数,计算OFDM符号数。将一个OFDM符号分成几个Segment,每个Segment所含比特数BitsInSegment与调制方式有关。对一部分数据进行拷贝填充N_{pad}使得数据可以正好用整数个OFDM符号传输,算出需要补充的比特数,得出完整的OFDM符号,有效利用子载波数。计算循环移位参数Cyclicshift,ROBO映射时依据N_{copies}及对应的移位参数Cyclicshift计算交织映射输出的数据。最后将相应的模式对应的参数用比特数表示,加入帧控制符号的控制帧中。本发明通过定义循环移位参数Cyclicshift和倒置变量NumberBitsShift,其取值与所想达到的拷贝次数与数据调制方式有关,避免了同一数据信息在同一个子信道上发送的情况,增强了载荷数据的抗信道衰落能力。



CN 107395545 B

1. 一种ROBO交织技术的实现方法,其特征在于,所述方法包括以下步骤:

步骤1:根据实际要求选择交织模式;

步骤2:计算OFDM符号数:根据实际可用载波数一个符号所含比特数, $BitsPerSymbol = BPC * N_{carrier_robo}$, BPC 为每子载波承载的比特数, $N_{carrier_robo}$ 为ROBO模式交织使用的子载波数目,计算所需OFDM符号个数 $N_{ofdm} = N_{raw} / BitsPerSymbol$, N_{raw} 为前面信道交织输出的数据,即进入ROBO交织的数据;

步骤3:将一个OFDM符号分成几个Segment,每个Segment所含比特数, $BitsInSegment$ 与调制方式有关, $BitsInSegment = BPC * CarriersInsegment$, $CarriersInsegment$ 为每个Segment所含的子载波数,最后一个OFDM符号如果不完整,ROBO交织通过补充数据使载波充分利用;

步骤4:为了避免出现最后一个OFDM符号的有用子载波只用一部分的情况,对一部分数据进行拷贝填充 N_{pad} , N_{pad} 为拷贝填充比特数,使得数据可以正好用整数个OFDM符号传输,计算最后一个符号所含比特数,如果最后一个符号比特数不为0,则有不完整符号,计算最后一段不完整的Segment所含比特数,由此算出需要补充的比特数,得出完整的OFDM符号,有效利用子载波数;

步骤5:计算循环移位参数Cyclicshift,它决定了每一次拷贝数据时,需要复制移位的比特序列长度,它与 N_{copies} 有关, N_{copies} 为拷贝次数,经过移位后,避免了同一数据信息在同一个子信道上发送的情况,增强了载荷数据的抗信道衰落能力,即每次拷贝的数据内容并不相同,利用了分集也提高了抗干扰性;

步骤6:进行ROBO映射,依据 N_{copies} 及对应的移位参数Cyclicshift计算交织映射输出的数据,每次拷贝所填充的数据不同,利用了分集,提高了数据可靠性;

步骤7:将相应的模式对应的参数用比特数表示,加入帧控制符号的控制帧中,接收端根据最大比合并处理接收数据,获取分集增益,增强数据的等效信噪比,再由相应的参数就可以解交织获得原数据,增强了数据传输中的稳定性。

2. 根据权利要求1所述的一种ROBO交织的实现方法,其特征在于步骤1中所述交织模式可以是PB136或PB520。

3. 根据权利要求1所述的一种ROBO交织的实现方法,其特征在于步骤1中选择交织模式时还支持一种扩展模式以适用于一些特殊场合,根据参数来确定ROBO交织实际可用的载波数 $N_{carrier_robo}$ 。

4. 根据权利要求1所述的一种ROBO交织的实现方法,其特征在于步骤2中实际传输所用子载波数随着ROBO交织的模式不同而不同,生成的OFDM符号个数也不同。

5. 根据权利要求4所述的一种ROBO交织的实现方法,其特征在于步骤4中所述 N_{pad} 为原输入数据的一部分。

6. 根据权利要求1所述的一种ROBO交织的实现方法,其特征在于步骤5和步骤6中定义的循环移位参数Cyclicshift,其取值与 N_{copies} 与 BPC 有关,避免了同一数据信息在同一个子信道上发送的情况,增强了载荷数据的抗信道衰落能力,即每次拷贝的数据内容并不相同,映射出的OFDM符号也就不同,利用了分集特性也提高了系统稳定性。

一种ROBO交织技术的实现方法

技术领域

[0001] 本发明属于通信技术领域,涉及ROBO交织技术的实现方法及其在宽带电力线载波通信上的应用。

背景技术

[0002] 低压电力线载波通信是利用低压配电线380/220V用户线作为信息传输媒介,通过载波方式将模拟或数字信号进行语音或数据传输的一种特殊通信方式。电力网作为电能输送的专用网络,是国家基础网络之一。其网络的建设质量、机械强度、安全经济技术指标等都是经过反复论证合理优化的。它的覆盖面之广、容量之大是任何网络都不能相比的。因此电力网是一种优质的不可多得资源。电力线载波技术在国内已经发展和应用多年,技术标准和管理规程已经相当的完备和成熟。随着电力线载波技术的不断发展,低压电力线载波技术也得到了飞速的发展。近几年,国内涌现出许多从事电力线载波技术研发的企业,如青岛东软、福星晓程、深圳瑞斯康、上海弥亚微等公司。其载波芯片在自动抄表系统、报警和安全监控系统、家居自动化系统等发面有广泛的应用。

[0003] 电力线载波通信信道的基本特征是:第一:时变衰减较大。对于一般的居民用户,我国采用的是220V交流两线供电。由于电网上负载的不断接入和切出、电器开和关等各种随机事件,使信道特性具有很强的时变性。另外,电力线本身的阻抗也会产生时变衰减。第二:信号变化复杂。实际测量表明在电力线上不同位置并联的诸多不同性质的负载对信号的传输影响很大。随着负载在电力线上的连接或断开,在不同的时刻信号衰减也会表现出不同的特点,即负载的变化是随机的,所以信号衰减也会随机地发生变化。第三:干扰噪声多样。电力线载波通信的最大干扰是噪声,其主要来源是电力网上的所有负载、无线电广播和天电等。

[0004] 总之,针对电力线载波通信信道的以上特点,已调信号应具有高的频谱利用率、抗噪声和抗干扰能力强、适宜于在衰落信道中传输等特点。高的抗干扰和抗多径衰落性能要求在恶劣的信道环境下能很好地工作,经过调制解调后的输出信噪比S/N较大或误码率较低。随着DFT的快速算法FFT的出现及数字信号处理技术的发展,OFDM再次成为研究的热点。OFDM技术有着诸多的优点:1)能有效克服码间干扰,抗干扰能力强;2)频带利用率高;3)系统均衡简单等等。电力线信道的恶劣条件正好给了OFDM技术施展的空间,它能有效地抵抗电力线网络环境中的各种干扰,实现数据的高速可靠的传输。

[0005] 宽带载波通信网络是以低压电力线为通信媒介,实现低压电力用户用电信息汇聚、传输、交互的通信网络,其主要采用正交频分复用技术,频段使用2MHz~12MHz。信道编码技术可以使得信号传输的可靠性大大增强,但是为了保证数据在宽带电力线信道中进行传输时的鲁棒特性,目前研究的热门技术未ROBO交织技术,目的是实现电力线通信具有较好的可靠性和鲁棒性。但是目前的ROBO交织技术仅仅利用了分集,每次分集包含的数据相同,不能充分利用分集的优点。

发明内容

[0006] 本发明的目的在于针对宽带载波通信过程中存在不稳定性,提出一种ROBO交织技术的实现方法,并得到ROBO交织在宽带载波通信上应用,该方法得到的交织技术,具有易于实现,解交织简单,可靠性高等特点,可以用于工宽带载波通信等通信领域。

[0007] 为解决上述问题,本发明采用的技术方案为一种ROBO交织技术的实现方法,包括以下步骤:

[0008] 步骤1:根据实际要求选择交织模式;

[0009] 步骤2:计算OFDM符号数:根据实际可用载波数一个符号所含比特数, $BitsPerSymbol = BPC * N_{carrier_robo}$,计算所需OFDM符号个数 $N_{ofdm} = N_{raw} / BitsPerSymbol$, N_{raw} 为前面信道交织输出的数据比特数,即进入ROBO交织的数据比特数;BPC是每个子载波承载的比特数; $N_{carrier_robo}$ 是ROBO交织使用的子载波数。

[0010] 步骤3:将一个OFDM符号分成几个Segment,每个Segment所含比特数, $BitsInSegment$ 与调制方式有关, $BitsInSegment = BPC * CarriersInSegment$, $CarriersInSegment$ 为每个Segment所含的子载波数,最后一个OFDM符号如果不完整,ROBO交织通过补充数据使载波充分利用;

[0011] 步骤4:为了避免出现最后一个OFDM符号的有用子载波只用一部分的情况,对一部分数据进行拷贝填充,拷贝填充比特数为 N_{pad} ,使得数据可以正好用整数个OFDM符号传输,计算最后一个符号所含比特数,如果最后一个符号比特数不为0,则有不完整符号,计算最后一段不完整的Segment所含比特数,由此算出需要补充的比特数,得出完整的OFDM符号,有效利用子载波数;

[0012] 步骤5:计算循环移位参数Cyclicshift,它决定了每一次拷贝数据时,需要复制移位的比特序列长度,它与拷贝次数 N_{copies} 有关。经过移位后,避免了同一数据信息在同一个子信道上发送的情况,增强了载荷数据的抗信道衰落能力,即每次拷贝的数据内容并不相同,利用了分集也提高了抗干扰性;

[0013] 步骤6:进行ROBO映射,依据拷贝次数 N_{copies} 及对应的移位参数Cyclicshift计算交织映射输出的数据,每次拷贝所填充的数据不同,利用了分集,提高了数据可靠性;

[0014] 步骤7:将相应的模式对应的参数用比特数表示,加入帧控制符号的控制帧中,接收端根据最大比合并处理接收数据,获取分集增益,增强数据的等效信噪比,再由相应的参数就可以解交织获得原数据,增强了数据传输中的稳定性。

[0015] 作为优选,步骤1中所述交织模式可以是PB136或PB520。

[0016] 步骤1中选择交织模式时还支持一种扩展模式以适用于一些特殊场合,根据参数来确定ROBO交织实际可用的载波数 $N_{carrier_robo}$ 。

[0017] 步骤2中实际传输所用子载波数随着ROBO交织的模式不同而不同,生成的OFDM符号个数也不同。

[0018] 步骤4中所述 N_{pad} 为原输入数据的一部分。

[0019] 步骤5和步骤6中定义的循环移位参数Cyclicshift,其取值与拷贝次数 N_{copies} 、每个子载波包含的比特数BPC有关,避免了同一数据信息在同一个子信道上发送的情况,增强了载荷数据的抗信道衰落能力,即每次拷贝的数据内容并不相同,映射出的OFDM符号也就不同,利用了分集特性也提高了系统稳定性。

[0020] 本发明具有如下有益效果：

[0021] 1、本发明利用基于分集拷贝准则所建立的多次拷贝数据模型，针对实际不同的通信条件与通信距离适合的ROBO交织模式进行了研究。

[0022] 2、本发明通过定义循环移位参数Cyclicshift和倒置变量NumberBitsShift，其取值与所想达到的拷贝次数与数据调制方式有关，避免了同一数据信息在同一个子信道上发送的情况，增强了载荷数据的抗信道衰落能力。

[0023] 3、本发明得到ROBO交织后的数据，运用最大比合并的方法来获取分集增益，增强数据的等效信噪比接收数据。

[0024] 4、本发明提出集分集复制与交织于一体的ROBO交织技术，该技术对有效信号进行一定次数的复制，复制次数越多，其ROBO特性越强。通过对信号进行ROBO传输，使得宽带电力线通信具有良好的鲁棒特性。

附图说明

[0025] 图1是ROBO交织后的符号图；

[0026] 图2是两种不同的交织参数对应的OFDM符号示意图；其中2a是HR_ROBO_0模式（ROBO基本模式）的第一次拷贝格式，2b是ROBO扩展模式的第一次拷贝格式，2c是ROBO扩展模式的第二次拷贝格式；

[0027] 图3是ROBO交织映射的流程图。

具体实施方式

[0028] 下面结合说明书附图对本发明创造作进一步的详细说明。本发明提出的一种ROBO交织技术，除了信道编码和信道交织之外，对载荷数据进行分集复制和一次ROBO交织。通过该技术对原数据进行分集复制与补充以保证在信道较差时仍有良好的通信性能。其中数据分集复制的方式及可使用子载波的数目与要求的分集次数和调制方式有关，并且每次拷贝包含的数据可能不同，保证不同的调制方式、不同的通信要求有不同的交织方式，与普通的交织方法不同，ROBO交织要求首先对有效信号进行一定次数的复制，复制次数越多，其ROBO特性越强。除此之外ROBO交织明确要求信号传输使用的子载波数必须是复制次数的整数倍，才能保证可行数目的载波调制。通过对信号进行ROBO传输，使得宽带电力线通信具有良好的鲁棒特性。

[0029] 如图1所示，为ROBO交织的几种模式图，本发明所涉及的ROBO交织技术要满足多种通信要求，因此确定结构的时候要首先计算好ROBO交织的参数。要由已知的数据长度，信道状况确定拷贝次数，拷贝次数越高，ROBO交织次数越多，稳定性越强，可以保证通信条件差时仍然具有较准确的接收数据。

[0030] 如图2所示，两种不同的交织参数对应的OFDM符号示意图。分别对应数据大小为PB520（520个字节）、调制方式为QPSK、拷贝次数为2与数据大小为PB520、16QAM调制、拷贝2次的情形，得到不同的移位参数Cyclicshift与倒置变量NumberBitsShift，从而形成包含不同子载波数的OFDM符号。

[0031] 第一种HR_ROBO_0模式具体参数（假设可用的载波数为411）：

[0032] 拷贝次数为2，则每个OFDM符号分成两段实际使用的子载波数

$N_{carrier_robo} = N_{copies} * \lceil \frac{N_{carrier}}{N_{copies}} \rceil = 2 * \lceil \frac{411}{2} \rceil = 410$ 每一段使用的子载波数为 $\lceil 411/2 \rceil = 205$, QPSK

调制则每个子载波包含2个比特,则每段比特数为 $205 * 2 = 410$,每个符号的比特数为 $410 * 2 = 820$,输入的总比特数为 $520 * 8 = 4160$ 需要的完整的OFDM符号数为 $\lceil 4160 / 820 \rceil = 5$,即最后一个符号没有使用全部的子载波,为了提高载波利用率并利用分集特性,采取补充数据的手段填充成完整的OFDM符号,计算最后一个符号的比特数为 $4160 - 820 * 5 = 60$,即最后一个符号只有不完整的一段,最后一段比特数为最后一个符号比特数-每段比特数*段数 = 60 ,补充数据使最后一段完整,则 $N_{pad} =$ 每段比特数-最后一段比特数 = $410 - 60 = 350$,其移位参数Cyclicshift为 $(0, 0)$,倒置变量NumberBitsShift = 0 ,每次拷贝所得到的OFDM符号即输入数据加上补充数据,具体见图2a。

[0033] 第二种具体参数:

[0034] 拷贝次数为2,则每个OFDM符号分成两段实际使用的子载波数

$N_{carrier_robo} = N_{copies} * \lceil \frac{N_{carrier}}{N_{copies}} \rceil = 2 * \lceil \frac{411}{2} \rceil = 410$,每一段使用的子载波数为 $\lceil 411/2 \rceil = 205$,采用

16QAM调制则每个子载波承载4个比特,则每段比特数为 $205 * 4 = 820$,每个符号的比特数为 $410 * 4 = 1640$,输入的总比特数为 $520 * 8 = 4160$,需要的完整的OFDM符号数为 $\lceil 4160 / 1640 \rceil = 2$,即最后一个符号没有使用全部的子载波,为了提高载波利用率并利用分集特性,采取补充数据的手段填充成完整的OFDM符号,计算最后一个符号的比特数为 $4160 - 1640 * 2 = 880$,即最后一个符号包含两端段,最后一段比特数为最后一个符号比特数-每段比特数*段数 = $880 - 820 = 60$,可见最后一段不完整,补充数据使最后一段完整,则 $N_{pad} =$ 每段比特数-最后一段比特数 = $820 - 60 = 760$,其移位参数Cyclicshift为 $(0, 1)$,倒置变量NumberBitsShift = 最后一段比特数 = 60 ,每次拷贝所得到的OFDM符号即输入数据加上补充数据,具体见图2b和2c。

[0035] 如图3所示,本发明方法的实现过程具体包括如下几个步骤:

[0036] 步骤1:根据实际要求选择交织模式(如PB136或PB520),具体模式见表1,本方法还支持一种扩展模式,适用于一些特殊场合,根据参数(如拷贝次数)来确定ROBO交织实际可

用的载波数 $N_{carrier_robo} = N_{copies} * \lceil \frac{N_{carrier}}{N_{copies}} \rceil$;

[0037] ROBO基本模式

TMI	ROBO 模式名称	PB 类型	Copy 次数	调制方式	码率	PB 数
0	SR_ROBO_0	PB520	4	QPSK	1/2	1-4
1	HR_ROBO_0	PB520	2	QPSK	1/2	1-4
2	MR_ROBO_0	PB136	5	QPSK	1/2	1-4
3	MR_ROBO_1	PB136	11	BPSK	1/2	1-4
4	MR_ROBO_2	PB136	7	BPSK	1/2	1-4
5	MR_ROBO_3	PB136	11	QPSK	1/2	1-4
6	MR_ROBO_4	PB136	7	QPSK	1/2	1-4
7	SR_ROBO_1	PB520	7	BPSK	1/2	1-4
8	SR_ROBO_2	PB520	4	BPSK	1/2	1-4
9	SR_ROBO_3	PB520	7	QPSK	1/2	1-4
10	HR_ROBO_1	PB520	2	BPSK	1/2	1-4

[0039] ROBO扩展模式

TMEx t	ROBO 模式名称	PB 类型	Copy 次数	调制方式	码率	PB 数
1	Sounding_520_1	PB520	1	16QAM	16/18	1-4
2	Sounding_520_2	PB520	2	16QAM	16/18	1-4
3	Sounding_520_3	PB520	1	16QAM	1/2	1-4
4	Sounding_520_4	PB520	2	16QAM	1/2	1-4
5	Sounding_520_5	PB520	4	16QAM	1/2	1-4
6	Sounding_520_6	PB520	1	QPSK	1/2	1-4
10	Sounding_520_1	PB136	5	16QAM	1/2	1-4

11	Sounding_520_2	PB136	2	QPSK	1/2	1-4
12	Sounding_520_3	PB136	2	16QAM	1/2	1-4
13	Sounding_520_4	PB136	1	QPSK	1/2	1-4
14	Sounding_520_5	PB136	1	16QAM	1/2	1-4

[0042] 表1

[0043] 步骤2: 计算OFDM符号数: 根据实际可用载波数一个符号所含比特数 $BitsPerSymbol = BPC * N_{carrier_robo}$ (BPC为每子载波调制的比特数) 计算所需OFDM符号个数 $N_{ofdm} = N_{raw} / BitsPerSymbol$ (N_{raw} 为前面信道交织输出的数据, 即进入ROBO交织的数据)

[0044] 步骤3: 将一个OFDM符号分成几个Segment, 每个Segment所含比特数

(BitsInSegment) 与调制方式有关, $BitsInSegment = BPC * CarriersInSegment$ (CarriersInSegment为每个Segment所含的子载波数), 最后一个OFDM符号可能不完整, ROBO交织通过补充数据使载波充分利用。

[0045] 步骤4: 为了避免出现最后一个OFDM符号的有用子载波只用一部分的情况, 对一部分数据进行拷贝填充 N_{pad} (原输入数据的一部分), 使得数据可以正好用整数个OFDM符号

[0046] 传输。计算最后一个符号所含比特数 $BitsInLastSymbol$, 看是否需要补充 N_{pad}

[0047] $BitsInLastSymbol = N_{raw} - BitsPerSymbol \lceil \frac{N_{raw}}{BitsPerSymbol} \rceil$ 如果最后一个符号比特数不为0, 则有不完整符号, 计算最后一段不完整的Segment所含比特数

[0048] $BitsInLastSegment = BitsInLastSymbol - BitsInSegment \lceil \frac{BitsInLastSymbol - 1}{BitsInSegment} \rceil$, 由此

算出需要补充的比特数 $N_{pad} = BitsInSegment - BitsInLastSegment$, 由此得出完整的OFDM符号, 有效利用子载波数。

[0049] 步骤5: 计算循环移位参数Cyclicshift, 它决定了每一次拷贝数据时, 需要复制移位的比特序列长度, 它与 N_{copies} 有关。经过移位后, 避免了同一数据信息在同一个子信道上发送的情况, 增强了载荷数据的抗信道衰落能力。即每次拷贝的数据内容并不相同, 利用了分集也提高了抗干扰性。具体参数计算如下:

[0050] 1) 如果拷贝次数 $N_{copies} = 1$, 则第一次拷贝的移位参数 $Cyclicshift(0) = 0$;

[0051] 2) 若 $N_{copies} = 2$, 则第一次拷贝的移位参数为 $Cyclicshift(0) = 0$, 第二次拷贝分两种情况:

[0052] a. 最后一个符号的比特数 $BitsInLastSymbol$ 不够一段的长度 $BitsInSegment$, 则移位参数 $Cyclicshift(1) = 0$

[0053] b. 最后一个符号的比特数 $BitsInLastSymbol$ 大于一段的长度 $BitsInSegment$, 则移位参数 $Cyclicshift(1) = 1$

[0054] 3) 若 $N_{copies} = 4$, 则情况比较复杂, 具体如下:

[0055] a. 若最后一个符号的比特数 $BitsInLastSymbol$ 小于 $BitsInSegment$, 即最后一个符号只有一个Segment, 则4次拷贝的移位参数 $Cyclicshift(0, 1, 2, 3) = (0, 0, 0, 0)$

[0056] b. 若最后一个符号包含2个Segment, 即 $BitsInLastSymbol \leq 2 * BitsInSegment$, 则移位参数 $Cyclicshift(0, 1, 2, 3) = (0, 0, 1, 1)$

[0057] c. 若最后一个符号包含3个Segment, 即 $BitsInLastSymbol \leq 3 * BitsInSegment$, 则移位参数 $Cyclicshift(0, 1, 2, 3) = (0, 0, 0, 0)$

[0058] d. 若最后一个符号包含4个Segment, 即 $BitsInLastSymbol \leq 4 * BitsInSegment$, 则移位参数 $Cyclicshift(0, 1, 2, 3) = (0, 1, 2, 3)$

[0059] 4) 若 $N_{copies} = 5$, 则有两种情况, 具体如下

[0060] a. 若最后一个符号不足5个Segment, 即 $BitsInLastSymbol \leq 4 * BitsInSegment$, 则5次拷贝的移位参数 $Cyclicshift(0, 1, 2, 3, 4) = (0, 0, 0, 0, 0)$

[0061] b. 若最后一个符号包含5个Segment, 即 $BitsInLastSymbol \leq 5 * BitsInSegment$, 则5次拷贝的移位参数 $Cyclicshift(0, 1, 2, 3, 4) = (0, 1, 2, 3, 4)$

[0062] 5) 若 $N_{copies} = 7$, 则有两种情况, 具体如下

[0063] a. 若最后一个符号不足7个Segment, 即 $BitsInLastSymbol \leq 6 * BitsInSegment$, 则7次拷贝的移位参数 $Cyclicshift(0,1,2,3,4,5,6) = (0,0,0,0,0,0,0)$

[0064] b. 若最后一个符号包含7个Segment, 即 $BitsInLastSymbol \leq 7 * BitsInSegment$, 则7次拷贝的移位参数 $Cyclicshift(0,1,2,3,4,5,6) = (0,1,2,3,4,5,6)$

[0065] 6) 若 $N_{copies} = 11$, 则有两种情况, 具体如下

[0066] a. 若最后一个符号不足11个Segment, 即 $BitsInLastSymbol \leq 10 * BitsInSegment$, 则11次拷贝的移位参数 $Cyclicshift(0,1,2,3,4,5,6,7,8,9,10) = (0,0,0,0,0,0,0,0,0,0,0)$

[0067] b. 若最后一个符号包含11个Segment, 即 $BitsInLastSymbol \leq 11 * BitsInSegment$, 则11次拷贝的移位参数 $Cyclicshift(0,1,2,3,4,5,6,7,8,9,10) = (0,1,2,3,4,5,6,7,8,9,10)$

[0068] 步骤6: ROBO映射, 依据 N_{copies} 及对应的移位参数 $Cyclicshift$ 计算交织映射输出的数据, 每次拷贝所填充的数据不同, 利用了分集, 提高了数据可靠性。

[0069] 设 k 为1到 N_{copies} , 若对应移位参数 $Cyclicshift(k) = 0$, 则每次拷贝的ROBO交织输出的数据就是输入数据加上补充数据 N_{pad} , 符号如图1;

[0070] 若对应移位参数 $Cyclicshift(k) = 1$, 则定义一个倒置变量 $NumberBitsShift$ (取输入数据的后 $NumberBitsShift$ 个数据), 再跟上补充数据 N_{pad} , 剩下的Segment由输入数据顺序组成, 符号如图1;

[0071] 步骤7: 将相应的模式对应的参数用比特数表示, 加入帧控制符号的控制帧中, 接收端根据最大比合并处理接收数据, 获取分集增益, 增强数据的等效信噪比, 再由相应的参数就可以解交织获得原数据, 增强了数据传输中的稳定性。

[0072] 此外, 本发明步骤一中的模式选择需要由表一中的模式选择, 从而达到更好的效果

[0073] 上述步骤2中, 当模式选定时, 由于拷贝次数、数据长度、调制方式不同, 实际传输使用的子载波数、生成的OFDM符号数、每个符号包含的数据不同, 使得步骤3中移位参数、倒置变量不同, 根据具体的计算公式推算每次交织的符号, 避免了同一数据信息在同一个子信道上发送的情况, 增强了载荷数据的抗信道衰落能力。

[0074] 具体地, 实际使用子载波数 $N_{carrier_robo} = N_{copies} * \lceil \frac{N_{carriers}}{N_{copies}} \rceil$, $BitsPerSymbol = BPC * N_{carrier_robo}$

$N_{carrier_robo}$

[0075] 为了避免出现最后一个OFDM符号的有用子载波只用一部分的情况, 对一部分数据进行拷贝填充 N_{pad} , 使得数据可以正好用整数个OFDM符号传输。计算最后一段Segment比特数

[0076] $BitsInLastSegment = BitsInLastSymbol - BitsInSegment \lceil \frac{BitsInLastSymbol - 1}{BitsInSegment} \rceil$, 由此

算出需要补充的比特数 $N_{pad} = BitsInSegment - BitsInLastSegment$, 由此得出完整的OFDM符号, 有效利用子载波数。

[0077] 其中, 填充完整以后需要对数据进行ROBO映射, 具体的映射方式与移位参数 N_{copies} 和倒置变量 $NumberBitsShift$ 有关, 相关详细参数见表2。

参数表示	解释	参数表示	解释
N_{raw}	进入 ROBO 交织的数据	$BitsInSegment$	每个片段所含比特数
BPC	每个子载波承载的比特数	$CarriersInSegment$	每个片段使用子载波数
$N_{carrier_robo}$	实际交织使用子载波数	$BitsInLastSymbol$	最后一个符号所含比特数
N_{copies}	复制（拷贝）次数	$BitsInLastSegment$	最后一个片段所含比特数
$BitsPerSymbol$	每个符号包含比特数	N_{pad}	填充的比特数
$Cyclicshift$	移位参数	$NumberBitsShift$	倒置变量

[0078] 表2

[0079] 表2

[0080] 本发明得到的ROBO交织技术,充分利用了子载波,并通过定义一个循环移位参数,得到不同的交织数据,避免了同一数据信息在同一个子信道上发送的情况,实现一种具有良好的鲁棒特性的数据符号,增强了载荷数据的抗信道衰落能力。本发明设计的ROBO交织技术根据电力线信道的衰落情况与收发机之间距离的不同而不同,适合各种通信需求。接收端采用最大比合并的方法来获取分集增益,增强数据的等效信噪比,这可以运用到宽带电力线载波通信之中,使宽带载波通信具有良好的可靠性。

[0081] 本发明利用基于分集拷贝准则所建立的多次拷贝数据模型,针对实际不同的通信条件与通信距离适合的ROBO交织模式进行了研究。根据定义循环移位参数Cyclicshift和倒置变量NumberBitsShift,其取值与所想达到的拷贝次数与数据调制方式有关,避免了同一数据信息在同一个子信道上发送的情况,增强了载荷数据的抗信道衰落能力。得到ROBO交织后的数据,运用最大比合并的方法来获取分集增益,增强数据的等效信噪比。该方法对多路信号进行同相加权合并,各支路的权重是由各支路信号所对应的信号功率与噪声功率的比值所决定的,合并输出的SNR等于各支路的SNR之和,所以即使各路信号都很差以至于没有一路信号可以被单独正确解调时,经过最大比合并后,仍有可能合成输出一个达到SNR要求的可被正确解调的信号。仿真结果可以显示,用普通交织方式对接收信号进行处理后,由于电力线信道衰落严重,每个子信道的数据信息都受到了严重的影响,导致每个子信道的数据信息都不能正确解调;用本发明得到的ROBO交织方式对接收信号处理,由于对数据进行ROBO交织时将原始数据拷贝了m次,最大比合并时充分利用了每个子信道的数据信息,其输出的SNR相当于各子信道SNR的m倍,使SNR水平达到载荷数据可以被正确解调的要求,所以系统的误比特性能有明显提升。进而得到可靠性强、稳定性高的OFDM符号,在宽带载波通信上得到很好的应用。

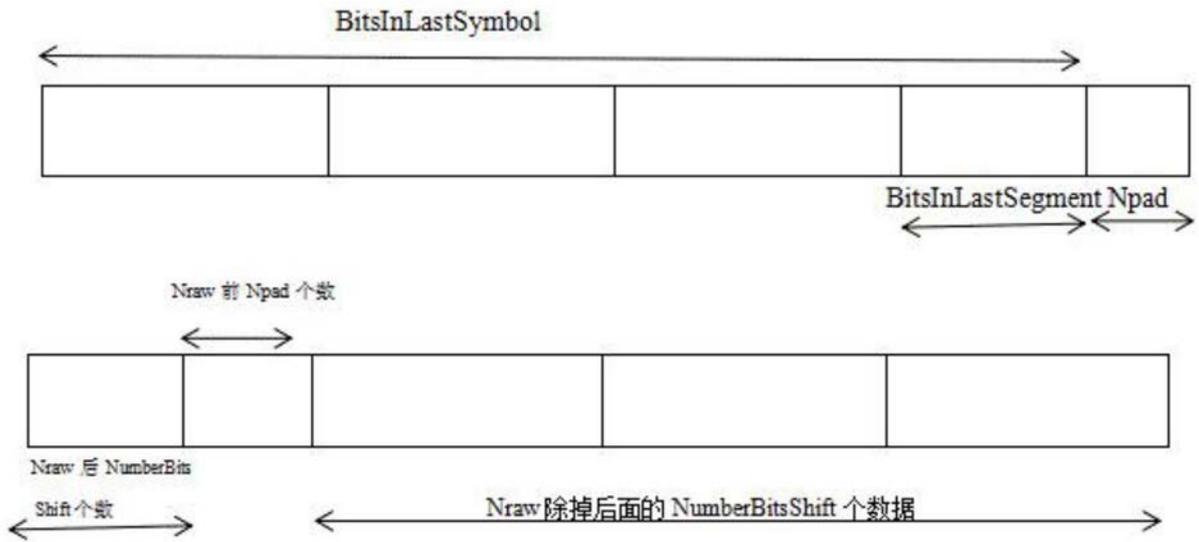


图1

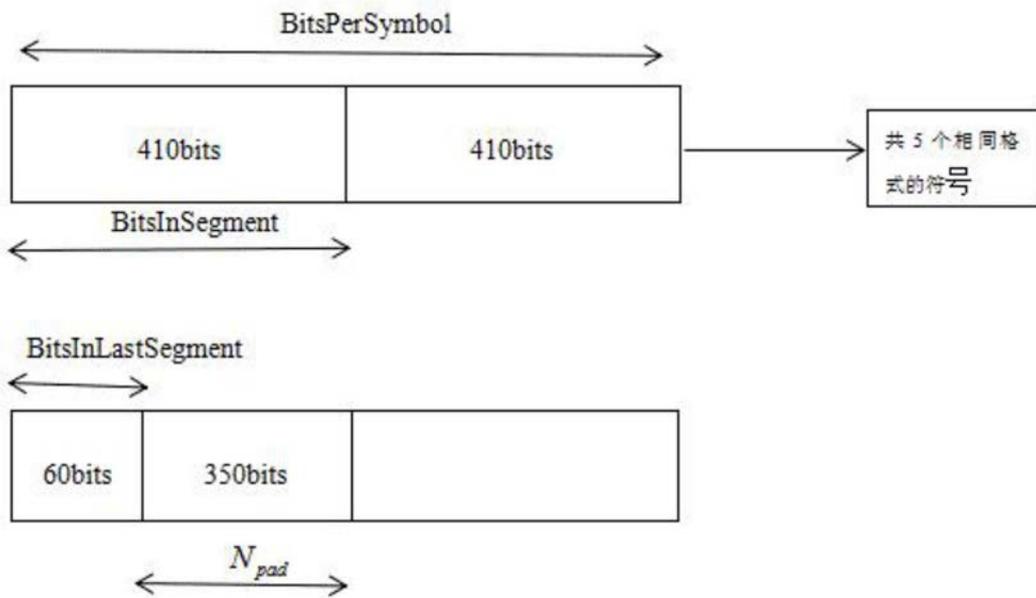


图2a

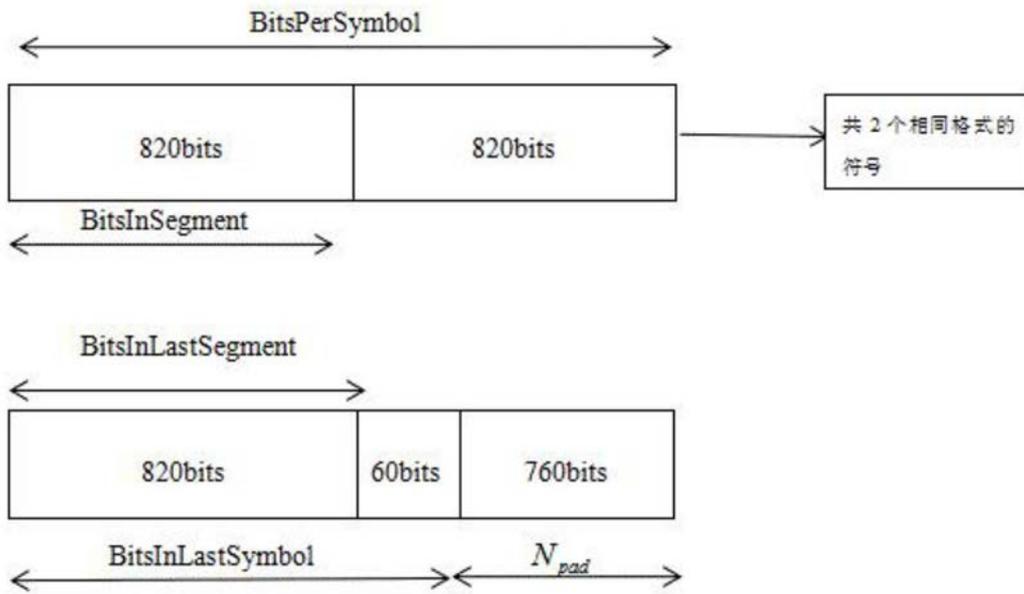


图2b

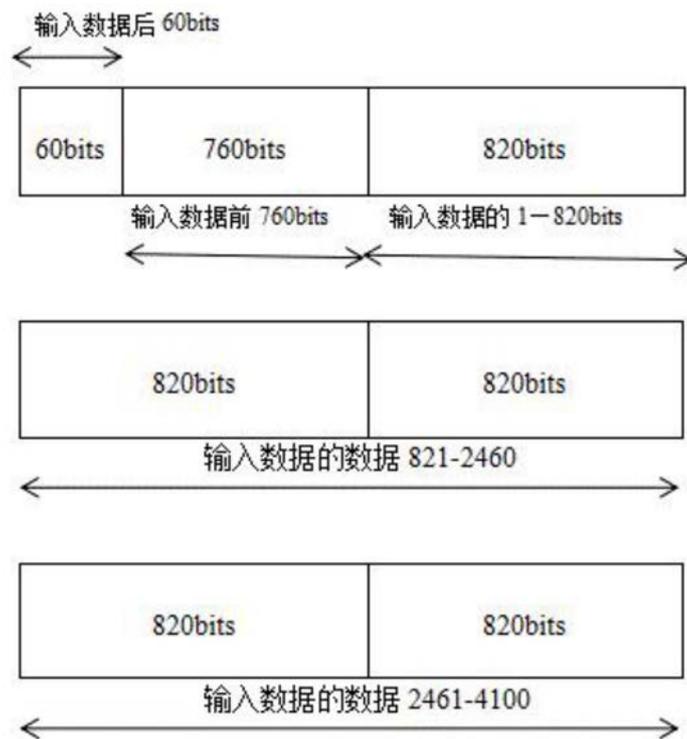


图2c

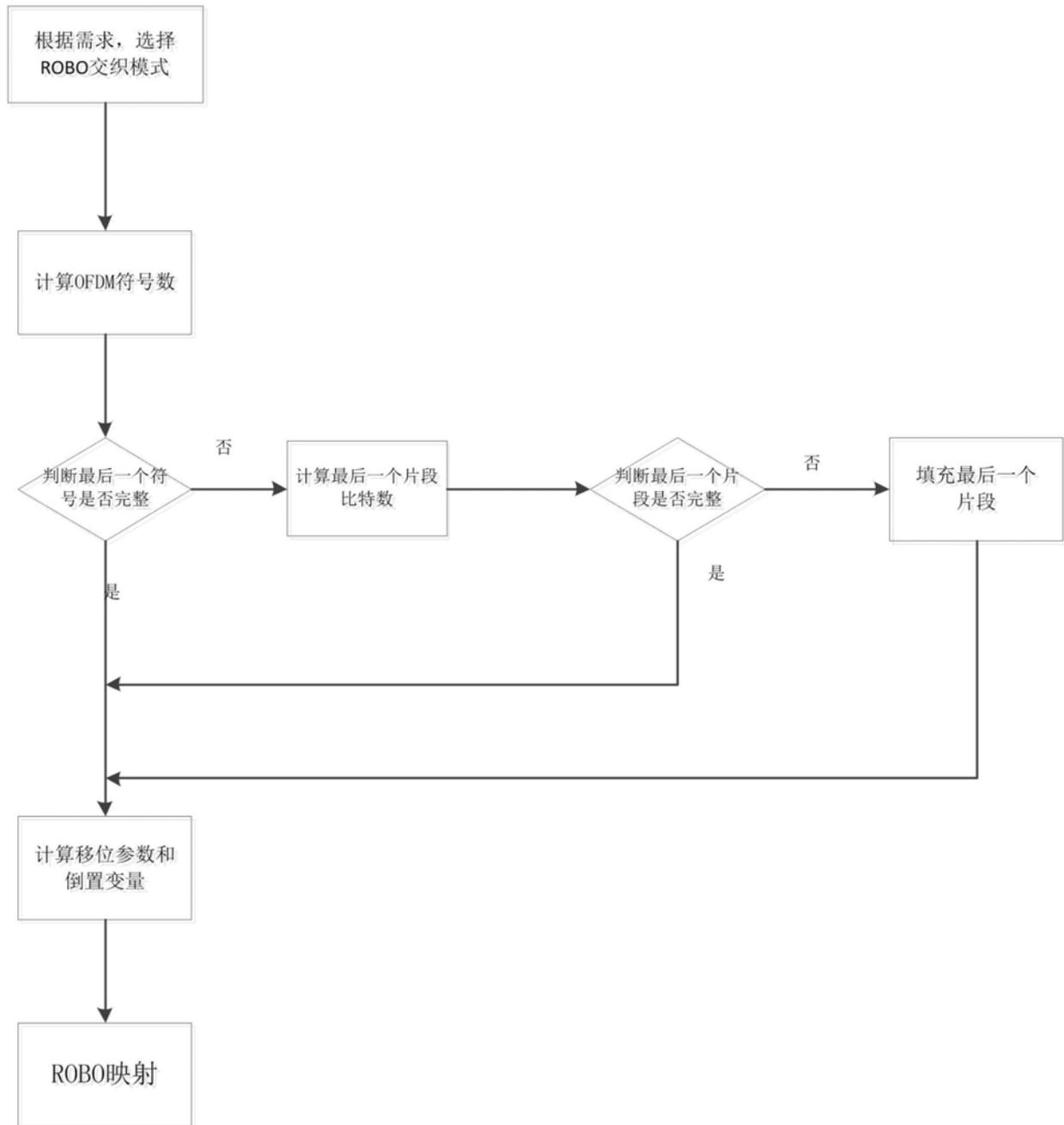


图3