



(12)发明专利申请

(10)申请公布号 CN 110999088 A

(43)申请公布日 2020.04.10

(21)申请号 201880048686.8

(74)专利代理机构 北京市柳沈律师事务所  
11105

(22)申请日 2018.04.20

代理人 郑海涛

(30)优先权数据

2017-143403 2017.07.25 JP

(51)Int.Cl.

H03M 7/40(2006.01)

(85)PCT国际申请进入国家阶段日

2020.01.20

G10L 19/00(2013.01)

(86)PCT国际申请的申请数据

PCT/JP2018/016227 2018.04.20

(87)PCT国际申请的公布数据

W02019/021552 JA 2019.01.31

(71)申请人 日本电信电话株式会社

地址 日本东京都

(72)发明人 杉浦亮介 镰本优 守谷健弘

原田登

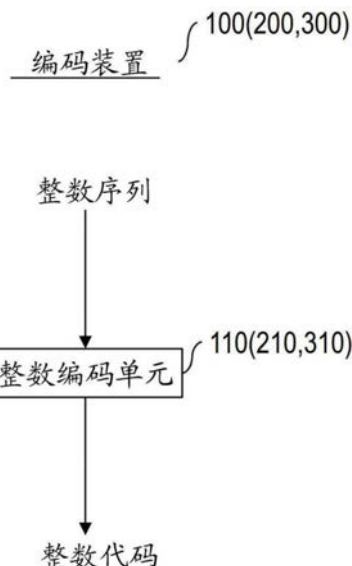
权利要求书4页 说明书27页 附图11页

(54)发明名称

编码装置、解码装置、代码串的数据结构、编码方法、解码方法、编码程序、解码程序

(57)摘要

本发明提供即使是还包含不是零值的小的值、其分布大大偏向于小的值的整数值的序列，也能够以较少的平均比特数进行编码的编码解码技术。包括整数编码单元110，对于基于被输入的非负的整数 $x_n, n \in \{1, 2, \dots, N\}$ 的序列(以下称为“整数序列”)，将K设为2以上的整数，作为与整数序列中包含的整数值0的2个至 $2K-1$ 个中的预先确定的任意个数即L个连续对应的代码，得到1比特的代码，作为与基于整数序列中包含的、0个至 $L-1$ 个连续的整数值0、和1个0以外的整数值 $x_n$ 的组对应的代码，得到 $K \times x_n$ 比特或者 $K \times x_n + 1$ 比特的代码。



1. 一种编码装置,包括:整数编码单元,其对于基于被输入的非负的整数值 $x_n, n \in \{1, 2, \dots, N\}$ 的序列、即整数序列,

将 $K$ 设为2以上的整数,

作为与在所述整数序列中包含的整数值0的2个至 $2^{K-1}$ 个中的预先确定的任意个数即 $L$ 个连续对应的代码,得到1比特的代码、即代码A,

作为与基于在所述整数序列中包含的、0个至 $L-1$ 个连续的整数值0、和1个0以外的整数值 $x_n$ 的组对应的代码,得到 $K \times x_n$ 比特或者 $K \times x_n + 1$ 比特的代码、即代码B。

2. 如权利要求1所述的编码装置,

假设比特值“x”为比特值“1”、比特值“y”为比特值“0”、或者比特值“x”为比特值“0”、比特值“y”为比特值“1”,

所述代码A是比特值为“x”的1比特的代码,

所述代码B在第1比特至第 $K$ 比特中包含至少一个比特值“x”和至少一个比特值“y”,并且从最后起的 $K \times (x_n - 1)$ 比特是比特值为“y”。

3. 如权利要求2所述的编码装置,

$L$ 为 $2^{K-1}$ ,

整数序列中的整数值0的0个至 $2^{K-1} - 1$ 个的 $2^{K-1}$ 种类的种类的连续个数的每一个与代码B为 $K \times x_n + 1$ 比特时的、

由该代码B的第1比特至第 $K+1$ 比特为止的 $K+1$ 比特构成、第1比特的比特值为“x”、第2比特至第 $K+1$ 比特为止的比特值为“y”的、1种类的部分代码、和

代码B为 $K \times x_n$ 比特时的、

由该代码B的第1比特至第 $K$ 比特为止的 $K$ 比特构成、第1比特的比特值为“y”、第2比特至第 $K$ 比特为止的 $K-1$ 比特中的至少1比特的比特值为“x”的、 $2^{K-1} - 1$ 种类的部分代码

的 $2^{K-1}$ 种类的部分代码中的其中一个一对一地对应。

4. 如权利要求1至3的任一项所述的编码装置,

所述整数序列是基于被输入的非负的整数值的序列中所包含的基于每规定样本的整数值的序列,

所述编码装置还包含参数决定单元,对于每个所述整数序列,进行

基于在所述整数序列中包含的整数值通过下式求出赖斯参数 $r$ 的处理

$$r = \log_2 \left( \frac{\ln 2}{N} \sum_{n=1}^N x_n \right);$$

基于在所述整数序列中包含的整数值通过下式求出编码参数 $K$ 的处理中的至少其中一个

$$K = \log_2 \left( 1 + \frac{1}{\frac{1}{N} \sum_{n=1}^N x_{-n}} \right); \text{ 以及}$$

对赖斯参数 $r$ 和编码参数 $K$ 中的至少其中一个进行编码,得到参数代码的处理,

所述整数编码单元对于每个所述整数序列,将所述参数决定单元得到的编码参数 $K$ 、或者从所述参数决定单元得到的所述赖斯参数 $r$ 通过 $K=2^r$ 得到的编码参数 $K$ ,用作上述 $K$ 。

5. 如权利要求4所述的编码装置,

所述整数编码单元在与所述整数序列对应的所述编码参数 $K$ 为1以下的情况下、或者与所述整数序列对应的所述赖斯参数 $r$ 为非负的情况下,将所述整数序列进行格伦布-赖斯编码,得到代码串。

6. 一种解码装置,包括:整数解码单元,其对于被输入的代码串,

将 $K$ 设为2以上的整数,

从被输入的所述代码串中包含的1比特的代码、即代码 $A$ ,得到整数值0的2个至 $2^{K-1}$ 个中的预先确定的任意个数即 $L$ 个的连续,

通过从被输入的所述代码串中包含的 $K \times x_n$ 比特或者 $K \times x_{n+1}$ 比特的代码、即代码 $B$ ,得到基于0个至 $L-1$ 个连续的整数值0、和1个0以外的整数值 $x_n$ 的组,从而得到基于整数值 $x_n, n \in \{1, 2, \dots, N\}$ 的序列。

7. 如权利要求6所述的解码装置,

假设比特值“ $x$ ”为比特值“1”、比特值“ $y$ ”为比特值“0”、或者比特值“ $x$ ”为比特值“0”、比特值“ $y$ ”为比特值“1”,

所述代码 $A$ 是比特值为“ $x$ ”的1比特的代码,

所述代码 $B$ 在第1比特至第 $K$ 比特中包含至少一个比特值“ $x$ ”和至少一个比特值“ $y$ ”,并且从最后起的 $K \times (x_{n-1})$ 比特是比特值为“ $y$ ”。

8. 如权利要求6或7所述的解码装置,

所述整数解码单元从被输入的所述代码串的最初开始按顺序进行

将被输入的所述代码串每次连续的 $K$ 比特、即 $K$ 比特串地读入、

在被读入的所述 $K$ 比特串是包含“ $y$ ”的比特串时,包含该“ $y$ ”的 $K$ 比特串的紧跟前的比特识别为是所述代码 $A$ 或者所述代码 $B$ 的末端的比特,

确定在被输入的所述代码串中包含的所述代码 $A$ 和所述代码 $B$ 。

9. 如权利要求7或8所述的解码装置,

$L$ 为 $2^{K-1}$ ,

代码 $B$ 为 $K \times x_{n+1}$ 比特时的、

由该代码 $B$ 的第1比特至第 $K+1$ 比特为止的 $K+1$ 比特构成、第1比特的比特值为“ $x$ ”、第2比特至第 $K+1$ 比特为止的比特值为“ $y$ ”的、1种类的部分代码、和

代码 $B$ 为 $K \times x_n$ 比特时的、

由该代码 $B$ 的第1比特至第 $K$ 比特为止的 $K$ 比特构成、第1比特的比特值为“ $y$ ”、第2比特至

第K比特为止的K-1比特中的至少1比特的比特值为“x”的、 $2^{K-1}-1$ 种类的部分代码的 $2^{K-1}$ 种类的部分代码的每一个

与基于通过所述整数解码单元得到的整数值的序列中的整数值0的0个至 $2^{K-1}-1$ 个的 $2^{K-1}$ 种类的连续个数中的其中一个一对一地对应。

10. 如权利要求6至9的任一项所述的解码装置，

还包括参数解码单元，被输入每规定单位的参数代码和所述代码串，对于每个规定单位，对所述参数代码进行解码，得到赖斯参数r和编码参数K中的至少其中一个，

所述整数解码单元对于每个规定单位，将所述参数解码单元得到的编码参数K、或者从所述参数解码单元得到的所述赖斯参数r通过 $K=2^r$ 得到的编码参数K，用作上述K。

11. 如权利要求10所述的解码装置，

所述整数解码单元在所述编码参数K为1以下的情况下、或者所述赖斯参数r为非负的情况下，对所述代码串进行格伦布-赖斯解码，得到基于整数值的序列。

12. 一种代码串的数据结构，是表示作为非负值的整数值 $x_n$ 、 $n \in \{1, 2, \dots, N\}$ 的序列、即整数序列的代码串的数据结构，

将K设为2以上的整数，

所述代码串，是在从所述整数序列得到所述代码串的编码装置、或者/以及从所述代码串得到所述整数序列的解码装置中使用的代码串，

包括：作为与在所述整数序列中包含的整数值0的2个至 $2^{K-1}$ 个中的预先确定的任意个数即L个连续对应的代码的1比特的代码、即代码A、

作为基于在所述整数序列包含的、与0个至L-1个连续的整数值0、和1个0以外的整数值 $x_n$ 的组对应的代码的 $K \times x_n$ 比特或者 $K \times x_n + 1$ 比特的代码、即代码B。

13. 如权利要求12所述的代码串的数据结构，

假设比特值“x”为比特值“1”、比特值“y”为比特值“0”、或者比特值“x”为比特值“0”、比特值“y”为比特值“1”，

所述代码A是比特值为“x”的1比特的代码，

所述代码B在第1比特至第K比特中包含至少一个比特值“x”和至少一个比特值“y”，并且从最后起的 $K \times (x_n - 1)$ 比特是比特值为“y”。

14. 如权利要求13所述的代码串的数据结构，

L为 $2^{K-1}$ ，

整数序列中的整数值0的0个至 $2^{K-1}-1$ 个的 $2^{K-1}$ 种类的连续个数的每一个与

代码B为 $K \times x_n + 1$ 比特时的、

由该代码B的第1比特至第K+1比特为止的K+1比特构成、第1比特的比特值为“x”、第2比特至第K+1比特为止的比特值为“y”的、1种类的部分代码、和

代码B为 $K \times x_n$ 比特时的、

由该代码B的第1比特至第K比特为止的K比特构成、第1比特的比特值为“y”、第2比特至第K比特为止的K-1比特中的至少1比特的比特值为“x”的、 $2^{K-1}-1$ 种类的部分代码

的 $2^{K-1}$ 种类的部分代码中的其中一个一对一地对应。

15. 一种编码方法，包括：整数编码步骤，其中，编码装置对于基于被输入的非负的整数值 $x_n$ 、 $n \in \{1, 2, \dots, N\}$ 的序列、即整数序列，

将K设为2以上的整数，

作为与所述整数序列中包含的整数值0的2个至 $2^{K-1}$ 个中的预先确定的任意个数即L个的连续对应的代码，得到1比特的代码、即代码A，

作为与基于所述整数序列中包含的、0个至L-1个连续的整数值0、和1个0以外的整数值 $x_n$ 的组对应的代码，得到 $K \times x_n$ 比特或者 $K \times x_n + 1$ 比特的代码、即代码B。

16. 一种解码方法，执行整数解码步骤，其中，解码装置对于被输入的代码串，

将K设为2以上的整数，

从被输入的所述代码串中包含的1比特的代码、即代码A，得到整数值0的2个至 $2^{K-1}$ 个中的预先确定的任意个数即L个的连续，

通过从在被输入的所述代码串中包含的 $K \times x_n$ 比特或者 $K \times x_n + 1$ 比特的代码、即代码B，得到基于0个至L-1个连续的整数值0、和1个0以外的整数值 $x_n$ 的组，得到基于整数值 $x_n, n \in \{1, 2, \dots, N\}$ 的序列。

17. 一种编码程序，用于使计算机具有权利要求1至5的任一项记载的编码装置的功能。

18. 一种解码程序，用于使计算机具有权利要求6至11的任一项记载的解码装置的功能。

## 编码装置、解码装置、代码串的数据结构、编码方法、解码方法、编码程序、解码程序

### 技术领域

[0001] 本发明涉及对由语音或音频时间序列数字信号的样本序列等的整数值构成的样本序列进行编码、解码的技术。

### 背景技术

[0002] 作为以压缩为目的对样本序列进行编码的技术,有通过将样本值进行量化而得到的有限精度的值(以下将其称为整数值)进行可逆编码,削减用于样本序列的记述的比特长的技术。在该技术中,对哪个整数值分配哪个长度的代码,直接关系到压缩的性能。该事实在将图像信号的样本串进行编码解码的图像编码或将音频信号的样本串进行编码解码的音频编码等的、样本序列的编码或解码的工学应用之处也并不例外。

[0003] 一般地,在可变长度的可逆编码中,由于解码可能性的制约,分配给整数值的代码的结构存在制约。具体地说,是若对某一整数值分配短的代码,则为了设为可解码的代码,必须对其它的整数值分配长的代码这样的制约。因此,为了提高压缩性能,代码的结构(对各整数值的代码的分配)需要设为适合整数值的序列中的值的分布的结构。具体地说,通过对出现概率高的整数值分配短的代码,对出现概率低的整数值分配长的代码,能够减小整数值的序列的压缩后的比特长的期望值。

[0004] 作为上述那样的可逆编码中最单纯的可变长度代码之一而使用的是格伦布-赖斯代码(Golomb-Rice代码)。已知格伦布-赖斯代码在整数值的序列属于拉普拉斯分布的情况下、即整数值的出现概率相对于值的大小指数性变低的情况下,实现最小的期望比特长(最小的比特长)。

[0005] 该格伦布-赖斯代码由于是非常单纯的结构,因此被广泛采用。格伦布-赖斯代码在将例如在对音频信号进行AD变换成为了PCM的整数值的序列或对该序列进行傅立叶变换而变换为频谱的序列等中包含的各样本值除以量化值,将得到的整数值的绝对值的序列进行编码时使用。或者,格伦布-赖斯代码在对将上述整数值的序列的各样本值 $x$ 分别置换为由以下的式(1)求出的 $x'$ 后的非负整数值构成的序列进行编码时使用。

$$[0006] \quad x' = \begin{cases} 2 \lfloor x \rfloor - 1 & (\text{如果 } x > 0) \\ 2 \lfloor x \rfloor & (\text{否则}) \end{cases} \quad \dots (1)$$

[0007] 因此,以下说明对非负整数值进行编码/解码的情况,但是从能够通过式(1)等将整数值整体一对一地变换为非负整数值这样的上述的事实,通过在编码处理的前段通过式(1)等将整数值整体变换为非负整数值,或者在解码处理的后段通过式(1)的逆变换等将非负整数值变换为整数值整体,还能够适应于整数值整体的编码/解码。

[0008] 格伦布-赖斯编码通过适当地赖斯参数(Rice参数) $r$ 的值,如图1的表那样,将相对于非负整数值的输入 $x$ 的代码的比特长 $B(x)$ 设为

$$[0009] \quad B(x) = \frac{x}{2^r} + r + 1 \quad \dots(2)$$

[0010] ,能够对于遵照以下的单侧拉普拉斯分布的非负整数值的序列提供近似且最佳的期望比特长地进行编码。

$$[0011] \quad p(x, 2^r) = \frac{1}{Z_{2^r}} \cdot \frac{1}{2^{2^r}} \left( Z_{2^r} = \sum_{k=0}^{\infty} \frac{1}{2^{2^k}} \right) \quad \dots(3)$$

[0012] 其中,赖斯参数 $r$ 是非负的整数值。即,如果将赖斯参数 $r$ 设为小的非负的整数值,则能够将值的偏向性大的序列以较少的平均比特数进行编码,如果将 $r$ 设为大的非负的整数值,则能够将值的偏向性小的序列以较少的平均比特数进行编码。

[0013] 但是格伦布-赖斯编码由于不能将赖斯参数 $r$ 设为负值,因此例如随着遵从分布 $p(x, 0.1)$ 的值的序列等序列的值的偏向性比分布 $p(x, 1)$ 变大,无论怎样设定赖斯参数 $r$ ,压缩的效率都会降低。

[0014] 例如,在将音频信号进行AD变换设为PCM时,由于设定连大音量也能记录这样可表现的数值的范围,因此在多数的信号的序列中包含的值处于可表现的数值的范围的绝对值偏向于小的值的位置的倾向。特别地,语音信号由于存在没有发话时间区间,因此偏向于零值或仅有背景噪声的非常小的值。而且,将在这样的信号的序列中包含的各值除以量化值得到的整数值的序列,通过量化,与原来的序列相比偏向于还包含零值的小的值。

[0015] 作为对偏向于零值的整数值的序列进行编码的现有技术,有非专利文献1的技术。非专利文献1的技术中,在输入的整数值的序列中包含的零值,是将零值连续的个数以预先决定的固定比特数进行编码,在输入的整数值的序列中包含的零值以外的值,是对从该值减去1得到的值(值-1)进行格伦布-赖斯编码的值。

[0016] 而且,作为对偏向于包含零值的小的值的整数值的序列进行编码的现有技术,存在非专利文献2的技术。非专利文献2的技术,是将基于两个整数值的组和一个整数值如全单射那样预先相关联,将基于在整数值的序列中包含的两个整数值的组变换为一个整数值,将变换后的整数值的序列进行格伦布-赖斯编码。

[0017] 【现有技术文献】

[0018] 【非专利文献】

[0019] 【非专利文献1】H.S.Malvar, "Adaptive Run-Length/Golomb-Rice Encoding of Quantized Generalized Gaussian Sources with Unknown Statistics", in Proc.Data Compression Conference (DCC) 06, IEEE Computer Society, pp.23-32, Mar.2006.

[0020] 【非专利文献2】高村誠之,八島由幸,“分布写像に基づくガウス性情報源の効率的符号化”,映像情報メディア学会誌,Vol.61, No.9, pp.1357-1362, 2007.

## 【发明内容】

[0021] 【发明要解决的课题】

[0022] 根据非专利文献1的技术,能够对于零值以较少的平均比特数进行编码。但是,在非专利文献1的技术中,对于零值以外由于进行与格伦布-赖斯编码大致相同的编码,因此在非专利文献1的技术中,在不是零值的小的值比分布 $p(x,1)$ 大大偏离的情况下,存在压缩的性能降低这样的课题。

[0023] 而且,非专利文献1的技术并不是与数学上所表现的分布对应的编码方法。因此,在非专利文献1的技术中,虽然知道整数值的序列中的值的分布,但是也存在不知道作为用于表现零值的连续的个数的比特数,要选择哪个进行编码,才能够以较少的平均比特数进行编码的课题。

[0024] 另一方面,根据非专利文献2的技术,能够对与分布 $p(x,1)$ 相比值偏离的整数值的序列以较少的平均比特数进行编码。但是,非专利文献2的技术,存在不能使对一个整数值的平均比特数比0.5比特小,对于理论上最佳的平均比特数成为不足0.5比特这样的偏向性大的整数值的序列,压缩的性能降低这样的课题。

[0025] 因此,本发明其目的在于,提供即使是还包含不是零值的小的值,其分布大大偏向于小的值的整数值的序列,也能够以较少的平均比特数进行编码的编码解码技术。

[0026] 【用于解决课题的手段】

[0027] 本发明的一个方式,包括整数编码单元,其对于基于被输入的非负的整数值 $x_n, n \in \{1, 2, \dots, N\}$ 的序列(以下称为“整数序列”),将 $K$ 设为2以上的整数,作为与所述整数序列中包含的整数值0的2个至 $2^{K-1}$ 个中的预先确定的任意个数即 $L$ 个的连续对应的代码,得到1比特的代码(以下,称为“代码A”),作为与基于所述整数序列中包含的、0个至 $L-1$ 个连续的整数值0、和1个0以外的整数值 $x_n$ 的组对应的代码,得到 $K \times x_n$ 比特或者 $K \times x_n + 1$ 比特的代码(以下,称为“代码B”)。

[0028] 本发明的一个方式包含整数解码单元,其对于被输入的代码串,将 $K$ 设为2以上的整数,从被输入的所述代码串中包含的1比特的代码(以下称为“代码A”),得到整数值0的2个至 $2^{K-1}$ 个中的预先确定的任意个数即 $L$ 个的连续,通过从被输入的所述代码串中包含的 $K \times x_n$ 比特或者 $K \times x_n + 1$ 比特的代码(以下称为“代码B”),得到基于0个至 $L-1$ 个连续的整数值0、和1个0以外的整数值 $x_n$ 的组,得到基于整数值 $x_n, n \in \{1, 2, \dots, N\}$ 的序列。

[0029] 本发明的一个方式,是表示作为非负值的整数值 $x_n, n \in \{1, 2, \dots, N\}$ 的序列(以下,称为“整数序列”)的代码串的数据结构,将 $K$ 设为2以上的整数,所述代码串在从所述整数序列得到所述代码串的编码装置、或者/以及从所述代码串得到所述整数序列的解码装置中使用的代码串,包含:作为与所述整数序列中包含的整数值0的2个至 $2^{K-1}$ 个中的预先确定的任意个数即 $L$ 个的连续对应的码的1比特的代码(以下,称为“代码A”)、和作为与在所述整数序列中包含的、基于0个至 $L-1$ 个连续的整数值0、和1个0以外的整数值 $x_n$ 的组对应的代码的 $K \times x_n$ 比特或者 $K \times x_n + 1$ 比特的代码(以下,称为“代码B”)。

[0030] 【发明的效果】

[0031] 按照本发明,即使是还包含不是零值的小的值并且大大偏向于小的值的整数值的序列、或在非专利文献2的技术中无法应对的程度的值偏向的整数值的序列,也能够以较少的平均比特数进行编码。而且,能够对那样的代码进行解码。



**附图说明**

- [0032] 图1是表示对于非负整数值的输入x的格伦布-赖斯代码的图。
- [0033] 图2是表示编码装置100 (200,300) 的结构的一例的方框图。
- [0034] 图3是表示编码装置100 (200,300) 的动作的一例的流程图。
- [0035] 图4是表示K=2时的代码树的一例的图。
- [0036] 图5是表示K=3时的代码树的一例的图。
- [0037] 图6是表示将K一般化时的代码树的一例的图。
- [0038] 图7是表示解码装置150 (250,350) 的结构的一例的方框图。
- [0039] 图8是表示解码装置150 (250,350) 的动作的一例的流程图。
- [0040] 图9A是表示K=2时的对应表的一例的图。
- [0041] 图9B是表示K=2时的对应表的一例的图。
- [0042] 图10A是表示K=3时的对应表的一例的图。
- [0043] 图10B是表示K=3时的对应表的一例的图。
- [0044] 图10C是表示K=3时的对应表的一例的图。
- [0045] 图10D是表示K=3时的对应表的一例的图。
- [0046] 图11是表示编码装置400 (401,402,500,501,502,600) 的结构的一例的方框图。
- [0047] 图12是表示编码装置400 (401,402,500,501,502,600) 的动作的一例的流程图。
- [0048] 图13是表示解码装置450 (451,452,550,551,552,650) 的结构的一例的方框图。
- [0049] 图14是表示解码装置450 (451,452,550,551,552,650) 的动作的一例的流程图。

**【具体实施方式】**

[0050] 以下,详细地说明本发明的实施的方式。而且,对于具有相同的功能的结构部分附加相同的标号,省略重复说明。

[0051] <第一实施方式>

[0052] 《编码装置》

[0053] 参照图2以及图3,说明第一实施方式的编码装置100执行的编码方法的处理过程。第一实施方式的编码装置100如图2所示,包含整数编码单元110。第一实施方式的编码装置100通过执行图3所示的各步骤的处理,实现第一实施方式的编码方法。

[0054] 在第一实施方式的编码装置100中被输入了非负的整数值的序列。作为该非负的整数值的序列,例如,也可以输入基于将由麦克风拾音的语音或者音乐等转换为数字信号得到的信号、将由摄像机拍摄的图像或影像转换为数字信号得到的信号等的信号的一部分或者全部,通过现有技术进行量化设为有限精度的值得到的整数值的绝对值的序列、或者基于将整数值作为x通过式(1)得到的 $x'$ 的序列。

[0055] 第一实施方式的编码装置100通过按照预先确定的规则使用预先确定的多个代码树,将非负的整数值的序列进行可变长度编码,对于比拉普拉斯分布偏向性大的分布的非负的整数值的序列,实现成为比格伦布-赖斯编码短的比特的长的编码处理。这里,所谓代码树,指表示对输入的非负的整数值的分配哪个代码的预先确定的规则的图表。以下,将对编码装置100输入的非负的整数值的序列简称为“整数值的序列”进行说明。

[0056] [整数编码单元110]

[0057] 整数编码单元110中输入被输入到编码装置100中的整数值的序列中、基于每次N样本(N为自然数)的整数值的序列。将基于输入的整数值的序列设为整数序列 $x_1, x_2, \dots, x_N$ 。整数编码单元110基于预先确定的或者通过未图示的部件输入的2以上的自然数即编码参数K,将整数序列 $x_1, x_2, \dots, x_N$ 以使用了以下的代码树的编码处理进行编码得到代码,将得到的代码作为整数代码输出(S110)。以下,设整数序列 $x_1, x_2, \dots, x_N$ 的各整数为 $x_n$ (n为序列中的样本号, $n \in \{1, 2, \dots, N\}$ )进行说明。

[0058] [[用于编码处理的代码树]]

[0059] 首先,例示用于整数编码单元110中的编码处理的代码树。

[0060] 整数编码单元110中预先存储图4至图6中例示那样的 $2^{K-1}$ 个(2的K-1次方个)的代码树。图4是K=2时的代码树的例子,T(0)和T(1)分别为代码树。图5是K=3时的代码树的例子,T(0)至T(3)分别为代码树。图6是将K一般化时的代码树的例子,从T(0)至T( $2^{K-1}-1$ )分别为代码树。以下,将对T附加的括弧内的号码称为代码树号码。

[0061] [[代码树中的所输入的整数值和与该整数值对应的代码的关系]]

[0062] 接着,说明各代码树中的被输入的整数值和与该整数值对应的代码之间的关系。

[0063] 以位于各代码树的节点的右侧的三角包围的整数与输入的整数对应。而且,位于各代码树的边缘的左侧的比特值‘0’以及‘1’与分配给输入的整数的代码的规则对应。该规则是,将位于从代码树的上端的节点至与输入的整数对应的节点为止的边缘的左侧的比特值排列后的值作为分配给输入的整数的代码而得到的规则。

[0064]  $2^{K-1}$ 个代码树中的一个代码树,是与图4至图6的T(0)对应的代码树,是与如下规则对应的代码树:作为与输入的整数 $x_n$ 对应的代码,得到比特数为 $K \times x_n + 1$ 比特、最初的第1比特的比特值为‘1’、从第2比特至第 $K \times x_n + 1$ 比特为止的比特值为‘0’的代码。而且,根据该代码树,在输入的整数 $x_n$ 为0的情况下,由于代码的比特数 $K \times x_n + 1$ 为1比特,因此得到比特值为‘1’的1比特的代码。

[0065]  $2^{K-1}$ 个代码树中的上述以外的 $2^{K-1}-1$ 个代码树,与图4至图6的T(1)至T( $2^{K-1}-1$ )对应,是与如下规则对应的代码树:作为与输入的整数 $x_n$ 对应的代码,得到比特数为 $K \times x_n$ 比特、最初的第1比特的比特值为‘0’、从第2比特至第K比特为止的K-1比特中的至少1比特的比特值为‘1’、从第K+1比特至第 $K \times x_n$ 比特为止的 $K \times (x_n - 1)$ 比特的比特值为‘0’、即从最后起的 $K \times (x_n - 1)$ 比特的比特值为‘0’的代码。这里,从第2比特至第K比特的K-1比特是与 $2^{K-1}-1$ 个代码树各自一对一地对应的比特群。而且,根据这些代码树,在输入的整数 $x_n$ 为0的情况下,由于代码的比特数 $K \times x_n$ 为0比特,因此得不到代码。

[0066] 例如,在图4的K=2的情况下的输入的整数和与该整数对应的代码的规则是如下这样的规则:在将输入的整数0以代码树T(0)进行编码的情况下,作为与输入的整数0对应的代码得到‘1’,在将输入的整数1以代码树T(0)进行编码的情况下,作为与输入的整数1对应的代码得到‘100’,在将输入的整数2以代码树T(0)进行编码的情况下,作为与输入的整数2对应的代码得到‘10000’,在将输入的整数0以代码树T(1)进行编码的情况下,得不到与输入的整数0对应的代码,在将输入的整数1以代码树T(1)进行编码的情况下,作为与输入的整数1对应的代码得到‘01’,在将输入的整数2以代码树T(1)进行编码的情况下,作为与输入的整数2对应的代码得到‘0100’。

[0067] [[使用多个代码树的规则]]

[0068] 接着,说明使用多个代码树的规则。整数编码单元110中,还预先存储使用多个代码树的规则。

[0069] 位于图4至图6的各代码树的节点的右侧的包围整数值的三角方向,与使用多个代码树的规则对应。

[0070] 朝右的三角,表示在对输入的整数值的下一整数值进行编码时,使用右邻的代码树(代码树号码大1的代码树)的规则。而且,此时,设为位于最右的代码树(代码树号码最大的代码树)的右邻为位于最左的代码树(代码树号码最小的代码树)。而且,该规则最终是一例,只要是循环使用多个代码树的规则,则什么样的规则都可以。即,对于各代码树,是在朝右的三角时,在下一整数值的编码中使用不同的代码树的规则,对于全部的代码树,只要是在朝右的三角时,用于下一整数值的编码的代码树不重复的规则,则什么样的规则都可以。而且,在循环使用多个代码树的规则中,使用的代码树的最大个数为 $2^{K-1}$ 个。

[0071] 朝下三角,表示在对输入的整数值的下一整数值进行编码时,使用代码树T(1)的规则。而且,该规则最终也是一例,在为朝下三角时,只要是对于各朝下三角将预先决定的任意一个代码树作为用于下一整数值的编码的代码树的规则,则什么样的规则都可以。例如,在为朝下三角时,也可以对于全部朝下三角将预先决定的任意一个相同的代码树作为用于下一整数值的编码的代码树的规则。而且,为了缩短平均比特数,在为朝下三角时,优选将与图4至图6的T(0)对应的代码树以外的代码树作为用于下一整数值的编码的代码树的规则。

[0072] 而且,为了使平均比特数最短,在为朝右三角时,可以设为如下的规则:循环使用 $2^{K-1}$ 个全部的代码树,并且对于全部的代码树,在为朝下三角时,将在与图4至图6的T(0)对应的代码树中为朝右三角时用于下一整数值的编码的代码树,作为用于下一整数值的编码的代码树。

[0073] 例如,使用图4的 $K=2$ 的情况下的代码树的编码的规则是如下规则:在将输入的整数0以代码树T(1)进行编码的情况下,得不到与输入的整数0对应的代码,而将在对下一整数进行编码时使用的代码树设为T(0),在对输入的整数2以代码树T(0)进行编码的情况下,作为与输入的整数2对应的代码得到‘10000’,将在对下一整数进行编码时使用的代码树设为T(1)。

[0074] 虽然在编码处理的最初使用多个代码树中的哪个代码树都可以,但是将哪个代码树用于编码处理的最初的规则,也作为使用多个代码树的规则而预先确定并存储。例如,在图4的 $K=2$ 的情况中,在将整数序列 $x_1, x_2, \dots, x_N$ 从整数 $x_1$ 起按顺序进行编码的情况下,既可以设为将整数 $x_1$ 以T(0)进行编码的规则,也可以设为将整数 $x_1$ 以T(1)进行编码的规则,但是预先确定并存储将T(0)和T(1)的哪一个用于整数 $x_1$ 的编码。

[0075] [[使用代码树的编码处理]]

[0076] 整数编码单元110使用上述的多个代码树,按照使用多个代码树的上述的规则,将输入的整数序列 $x_1, x_2, \dots, x_N$ 中包含的各整数按照预先确定的顺序进行编码得到代码,并将得到的代码作为整数代码输出。例如,预先确定的顺序是序列中的样本号码顺序,在最初使用的代码树为T(0)的情况下,整数编码单元110进行的编码步骤如下述这样。

[0077] (编码步骤)

[0078] 1.将样本号码 $n$ 、代码树号码的初始值分别设为1、0。

[0079] 2. 直至 $n=N$ 为止,反复以下的(i)~(iv)。

[0080] (i) 如果代码树号码为0,则得到代码‘1’。

[0081] (ii) 如果是整数值 $x_n=0$ ,则将对代码树号码加上1后得到的值作为新的代码树号码。这时,在加上1后代码树号码变为 $2^{K-1}$ 的情况下,将0作为新的代码树号码。

[0082] (iii) 如果是整数值 $x_n \neq 0$ ,则得到基于1比特的‘0’和与代码树一对一地对应的对应的 $K-1$ 比特的 $K$ 比特的代码,进而得到 $K \times (x_n-1)$ 比特的代码‘0’,将1作为新的代码树号码。

[0083] (iv) 将对 $n$ 加上1得到的值作为新的 $n$ 。

[0084] 3. 去除在通过1~2得到的代码中的最初得到的代码‘1’。

[0085] (或者不进行开始的1次的2.(i)操作。)

[0086] 4. 对通过1~3得到的代码的最后加上代码‘1’(末端代码)。

[0087] 5. 将通过1~4得到的代码作为整数代码输出。

[0088] 而且,进行编码步骤中的3并不是必须的。在最初使用的代码树为 $T(0)$ 的情况下,无论 $x_1$ 的值为什么样的值,通过1~2得到的代码中的最初得到的代码都为‘1’。因此,即使编码装置100将去除了最初得到的代码‘1’的结果作为整数代码输出,如果对解码装置150所输入的整数代码的最初加上代码‘1’,则解码装置150也能够进行与编码装置100的整数编码单元110进行的编码处理对应的解码处理。由此,在编码步骤的3中就去除了最初得到的代码‘1’。但是,也可以未必需要进行编码步骤的3,而是不去除通过1~2得到的代码中的最初得到的代码‘1’。在该情况下,对应的解码装置150只要不进行在输入的整数代码的最初加上代码‘1’的处理即可,也可以不进行编码步骤中的3。

[0089] 而且,通过从编码装置100向解码装置150的整数代码的传递方法,也可以不进行编码步骤中的4。解码装置150中的解码的步骤如后所述,但是为了解码装置150进行与整数编码单元110进行的编码处理对应的解码处理,需要解码装置150能够确定整数代码的末端。作为从编码装置100向解码装置150的整数代码的传递的结构,例如可以采用在编码装置100内或者编码装置100的后段中、将关于整数序列 $x_1, x_2, \dots, x_N$ 得到的整数代码使用公知的分组化的方法存储在分组中传递给解码装置150的结构。在该情况下,由于在解码侧,采用在解码装置150内或者解码装置150的前段中从分组取出整数代码的结构,因此解码装置150能够确定整数代码的末端。因此,在这样的情况下,也可以不进行编码步骤中的4。作为从编码装置100向解码装置150传递整数代码的结构,有时也采用将在对于整数序列 $x_1, x_2, \dots, x_N$ 得到的整数代码的后面连结了其它的代码后的代码存储在上述的分组中传递给解码装置150的结构、或者不进行上述的分组化而使在对于整数序列 $x_1, x_2, \dots, x_N$ 得到的整数代码的后面连结其它的代码并传递给解码装置150的结构等。在这些结构中,若假设没有进行编码步骤中的4,在使后面连结的代码为‘0000’时等,则解码装置150会无法判别与第 $N$ 个的整数值对应的代码到哪里为止,使在后面连结的代码从哪里开始。因此,在这些结构中,需要通过将预先确定的相同的 $N$ 的值存储在编码装置100和解码装置150中等,解码装置150能够掌握整数代码是与 $N$ 个整数值对应的代码的情况,并且还进行编码步骤中的4,解码装置150能够确定整数代码的末端。

[0090] 而且,上述的说明的代码‘0’是2值的比特值中的其中一个的比特值、上述的说明的代码‘1’是2值的比特值中的另一个比特值的例子。即,当然,也可以取代上述的说明的代

码‘0’使用代码‘1’，取代代码‘1’而使用代码‘0’。

[0091] [[编码处理的例子]]

[0092] 作为编码处理的例子，说明 $K=2$ 中最初使用的代码树为 $T(0)$ 的情况下的、输入的整数序列 $x_1, x_2, x_3, x_4, x_5$ 为 $0, 0, 2, 0, 0$ 的情况下的编码处理。首先，由于开始的整数 $x_1=0$ 以 $T(0)$ 进行编码，因此得到代码‘1’，由于将对代码树号码 $0$ 加上 $1$ 得到的值设为新的代码树号码，因此将用于下一整数 $x_2=0$ 的编码的代码树设为 $T(1)$ 。由于下一整数 $x_2=0$ 以 $T(1)$ 进行编码，因此得不到代码，将对代码树号码 $1$ 加上了 $1$ 得到的值设为代码树号码，但是由于加上了 $1$ 后的代码树号码成为 $2^{K-1}$ ，因此，将 $0$ 设为新的代码树号码，将用于下一个整数 $x_3=2$ 的编码的代码树设为 $T(0)$ 。由于第3个的整数 $x_3=2$ 以 $T(0)$ 进行编码，因此得到代码‘10000’，并且由于是将新的代码树号码设为 $1$ 的规则，因此将用于下一整数 $x_4=0$ 的编码的代码树设为 $T(1)$ 。由于第4个的整数 $x_4=0$ 以 $T(1)$ 进行编码，因此什么代码也不追加，将代码树从 $T(1)$ 向 $T(0)$ 迁移。由于最后的整数 $0$ 以 $T(0)$ 进行编码，因此得到代码‘1’。至此为止所得到的代码为‘1100001’，但是在进行上述的编码步骤的 $3$ ，不进行上述的编码步骤的 $4$ 的情况下，省略最初得到的代码‘1’，得到最终的整数代码‘100001’。

[0093] 《解码装置》

[0094] 参照图7以及图8，说明第一实施方式的解码装置150执行的解码方法的处理过程。如图7所示，第一实施方式的解码装置150包含整数解码单元160。第一实施方式的解码装置150通过执行图8所示的各步骤的处理，实现第一实施方式的解码方法。

[0095] 在第一实施方式的解码装置150中，被输入第一实施方式的编码装置100输出的整数代码。第一实施方式的解码装置150，是将输入的整数代码，以按照与第一实施方式的编码装置100对应的预先确定的规则使用了与第一实施方式的编码装置100对应的预先确定的多个代码树的解码处理进行解码，得到非负的整数值的序列，将输入到第一实施方式的编码装置100的非负的整数值的序列本身复原的装置。以下，与上述的编码装置100的说明同样，将非负的整数值仅称为“整数值”进行说明。

[0096] [整数解码单元160]

[0097] 整数解码单元160中，被输入在解码装置150中被输入的整数代码。整数解码单元160，基于与编码装置100的整数编码单元110使用的编码参数 $K$ 相同的值的、且为预先确定的或者通过未图示的部件输入的 $2$ 以上的自然数的编码参数 $K$ ，将被输入的整数代码以使用了以下的代码树的解码处理进行解码，得到整数值的序列，输出得到的整数序列 $x_1, x_2, \dots, x_N$  (S160)。以下，设整数序列 $x_1, x_2, \dots, x_N$ 的各整数为 $x_n$  ( $n$ 为序列中的样本号码， $n \in \{1, 2, \dots, N\}$ ) 进行说明。

[0098] 用于解码处理的代码树，与第一实施方式的编码装置100的整数编码单元110使用的代码树相同，在整数解码单元160中预先存储图4至图6例示那样的 $2^{K-1}$ 个 ( $2$ 的 $K-1$ 次方个)的代码树。解码处理中的代码树中的输入的代码和与该该代码对应的整数值之间的关系以及使用多个代码树的规则是如下规则：是与第一实施方式的编码装置100的整数编码单元110的编码处理中的规则对应，将第一实施方式的编码装置100的整数编码单元110的编码处理中的规则的“分配给输入的整数值的代码”换读为“输入的代码”，将“输入的整数值”换读为“分配给输入的代码的整数值”的规则，被预先存储在整数解码单元160中。而且，用于解码处理的参数 $K$ 的值，是与第一实施方式的编码装置100的整数编码单元110使用的参数 $K$

相同的值,预先存储在整数解码单元160中。而且,在解码处理中最初使用哪个代码树的规则,也是与第一实施方式的编码装置100的整数编码单元110使用的规则相同的规则,预先存储在整数解码单元160中。

[0099] 整数解码单元160使用上述的多个代码树,按照使用多个代码树的上述的规则,得到基于与输入的整数代码对应的整数值的列,将基于得到的整数值的列的各整数设为预先确定的顺序后的结果,得到作为整数序列 $x_1, x_2, \dots, x_N$ 并输出。例如,在预先确定的顺序为序列中的样本号码顺序,最初使用的代码树为 $T(0)$ 的情况下,整数解码单元160进行的解码步骤,例如如下述的解码步骤1或解码步骤2这样。

[0100] (解码步骤1-1)在不进行基于编码步骤4的末端代码的追加的情况下,即,解码装置150可确定整数代码的末端的情况下

[0101] 1.在编码步骤3中去除了代码‘1’的情况下,在整数代码的最初追加代码‘1’。

[0102] 2.将代码树号码的初始值设为0。

[0103] 3.反复以下(i)~(ii),直到没有未读入的整数代码为止,如果没有了未读入的整数代码,则进至4。

[0104] (i)在与代码树号码对应的代码树中存在与读入的代码对应的代码的期间(即,直至与代码树号码对应的代码树中没有与读入的代码对应的代码的前一个比特为止),按顺序读入被输入的整数代码。

[0105] (ii)从与代码树号码对应的代码树,得到与读入的代码对应的代码所对应的整数值,按照对于该代码树的整数值的代码树的迁移的规则,得到新的代码树号码。

[0106] 4.将基于由3的(ii)得到的整数值的序列作为整数序列 $x_1, x_2, \dots, x_N$ 输出。

[0107] (解码步骤1-2)在进行编码步骤4的末端代码的追加的情况下,即,解码装置150无法确定整数代码的末端的情况下

[0108] 1.在编码步骤3中去除了代码‘1’的情况下,在输入的代码串(基于与整数代码后续的有关代码的代码串)的最初追加代码‘1’。

[0109] 2.将样本号码 $n$ 、代码树号码的初始值分别设为1、0。

[0110] 3.直至成为 $n=N$ 为止,反复以下(i)~(ii)。

[0111] (i)在与代码树号码对应的代码树中存在与读入的代码对应的代码的期间(即,直至与代码树号码对应的代码树中没有与读入的代码对应的代码的前一个比特为止),按顺序读入被输入的代码串。

[0112] (ii)从与代码树号码对应的代码树,得到与读入的代码对应的代码对应的整数值,按照关于该代码树的整数值的代码树的迁移的规则,得到新的代码树号码,将对 $n$ 加上1后的值设为新的 $n$ 。

[0113] 4.将基于由3的(ii)得到的整数值的序列作为整数序列 $x_1, x_2, \dots, x_N$ 输出。

[0114] (解码步骤2)

[0115] 1.在编码步骤4中未进行末端代码的追加的情况下,在输入的代码串的最后追加末端代码‘1’。而且,在编码步骤3中未去除代码‘1’的情况下,在输入的代码串的最初去除代码‘1’。

[0116] 2.将样本号码 $n$ 、代码树号码的初始值分别设为1、0。

[0117] 3.直至变为 $n \geq N$ 为止,反复以下(i)~(v)。

[0118] (i) 在K比特全部为代码‘0’的期间,将按顺序每次K比特地读取输入的代码串,将在其间读入的代码‘0’的个数除以K得到的数设为L。

[0119] (ii) 如果代码树号码为0,则得到整数L。

[0120] (iii) 如果代码树号码不为0,则得到整数(L+1)。

[0121] (iv) 如果输入的代码串的下一个的1比特为代码‘1’,则读取该1比特,得到 $(2^{K-1}-1)$ 个整数0,将0设为新的代码树号码,将对n加上了 $2^{K-1}$ 后的值设为新的n。

[0122] (v) 如果输入的代码串的下一个的1比特为代码‘0’,则读入包含该1比特的K比特,将与该K比特的代码对应的数设为新的代码树号码,得到(新的代码树号码-1)个整数0,将对n加上了新的代码树号码后的值设为新的n。

[0123] 4. 在所得到的整数多于N个的情况下,去除第N+1个的整数以后。

[0124] [[解码处理的例子]]

[0125] 作为解码处理的例子,说明K=2中在最初使用的代码树为T(0)的情况下的、将在上述的编码处理的例子中得到的整数代码‘100001’以上述的解码步骤1-1进行解码的情况下的解码处理。首先,在整数代码‘100001’的开头追加代码‘1’,将整数代码设为‘1100001’。接着,作为整数代码‘1100001’的最初的1比特的‘1’虽然有与代码树T(0)对应的代码,但是由于作为最初的2比特的‘11’没有与代码树T(0)对应的代码,因此读入作为整数代码的最初的1比特的‘1’,得到与‘1’对应的整数0,将接下来使用的代码树设为T(1)。接着,由于作为整数代码的剩余的‘100001’的最初的1比特的‘1’没有与代码树T(1)对应的代码,因此也不读入最初的1比特,得到与0比特对应的整数0,将接下来使用的代码树设为T(0)。接着,从作为整数代码的剩余的‘100001’的最初的1比特的‘1’到作为最初的5比特的‘10000’为止,虽然对应的代码存在于代码树T(0),但是,由于作为整数代码的剩余的‘100001’的最初的6比特的‘100001’对应的代码不存在于代码树T(0)中,因此读入作为整数代码的剩余的‘100001’的最初的5比特的‘10000’,得到与‘10000’对应的整数2,将接下来使用的代码树设为T(1)。接着,由于整数代码的剩余的‘1’对应的代码不存在于代码树T(1)中,因此也不读入最初的1比特,而得到与0比特对应的整数0,将接下来使用的代码树设为T(0)。接着,整数代码的剩余的‘1’对应的代码存在于代码树T(0),在该‘1’以后没有未读入的整数代码,因此得到与‘1’对应的整数0,将基于通过这些处理得到的整数0,0,2,0,0的序列作为整数序列 $x_1, x_2, x_3, x_4, x_5$ 输出。

[0126] <发明的原理的说明>

[0127] 这里说明本发明的原理。

[0128] 根据第一实施方式的编码装置100,能够对全部的非负整数分配代码,通过第一实施方式的解码装置150对该代码进行解码,能够将非负整数本身复原。如图4至图6所示那样,与非负整数 $x_n$ 对应的代码为 $K \times x_n$ 比特或者 $K \times x_n + 1$ 比特。由此,在第一实施方式中使用的本发明的代码,是对于以下的偏向性大的按照单侧拉普拉斯分布的非负整数的序列近似地提供最佳的期望比特长的代码,是能够使相对于大大偏向于小的值的非负整数的序列的平均比特数比格伦布-赖斯代码少的代码。

$$[0129] \quad p(x, 1/K) = \frac{1}{Z_{1/K}} \cdot \frac{1}{2^{Kx}} \quad \left( Z_{1/K} = \sum_{k=0}^{\infty} \frac{1}{2^{Kk}} \right) \quad \dots (4)$$

[0130] 根据以式(3)所示的格伦布-赖斯代码为最佳的分布 $p(x, 2^r)$ 、和以式(4)所示的本发明的代码为最佳的分布 $p(x, 1/K)$ 的对应关系,可知本发明的编码参数K的倒数的对数与赖斯参数r相对应。而且,由于本发明的编码参数K为正的整数值,因此编码参数K的倒数的对数成为负值,所以编码参数K可以解释作为与扩展为负值的赖斯参数对应的参数。即,通过进行对非负整数值的序列最佳的赖斯参数的估计,能够估计分配给非负整数值的序列的本发明的代码中的最佳的参数K。

[0131] 在第一实施方式的编码装置100和解码装置150中使用的本发明的代码树中的整数值和与该整数值对应的代码之间的关系、以及使用多个代码树的规则,不仅将与非负整数值 $x_n$ 对应的代码设为 $K \times x_n$ 比特或者 $K \times x_n + 1$ 比特,还考虑设为可解码的代码来进行设计。具体地说,设置如下这样的制约:对于整数值0的2个至 $2^{K-1}$ 个中的预先确定的任意个数的连续,分配作为1比特的代码的‘1’,对于整数值0的0个至预先确定的最大 $2^{K-1} - 1$ 个连续和其后续的一个非负整数值 $x_n \neq 0$ ,分配在最初的K比特中至少包含1比特的比特值‘1’、并且以后的比特值为‘0’的代码。通过设置该制约,在解码装置150中读入与由多个整数值构成的整数序列对应的整数代码时,会知道在K比特中包含比特值‘1’的部分中,包含整数代码中的某一代码和后一个的代码的边界。而且,为了满足上述的制约,对于整数值0的0个至预先确定的最大 $2^{K-1} - 1$ 个连续和其后续的一个非负整数值 $x_n$ 的代码的比特数不仅是 $K \times x_n$ 比特,还设为 $K \times x_n$ 比特或者 $K \times x_n + 1$ 比特。

[0132] <第二实施方式>

[0133] 在第一实施方式中,说明了使用预先确定的代码树实现本发明的编码和解码的方式,但是即使是取代预先确定的代码树,使用预先确定的对应表,也能够实现与第一实施方式等效的编码处理和解码处理。将使用该预先确定的对应表的方式作为第二实施方式进行说明。

[0134] 《编码装置》

[0135] 参照图2以及图3,说明第二实施方式的编码装置200执行的编码方法的处理过程。第二实施方式的编码装置200如图2所示,与第一实施方式的编码装置100同样,包含整数编码单元210。第二实施方式的编码装置200与第一实施方式的编码装置100同样,通过执行图3所示的各步骤的处理,实现第二实施方式的编码方法。

[0136] 在第二实施方式的编码装置200中,与第一实施方式同样,被输入非负的整数值的序列。第二实施方式的编码装置200通过遵照预先确定的对应表,将非负的整数值的序列进行可变长度编码,对于比拉普拉斯分布偏向性大的分布的非负的整数值的序列,实现成为比格伦布-赖斯编码短的比特长的编码处理。以下,将被输入到编码装置200的非负的整数值仅称为“整数值”进行说明。

[0137] [整数编码单元210]

[0138] 整数编码单元210中被输入在编码装置200中所输入的整数值的序列中的、每次N样本的整数序列 $x_1, x_2, \dots, x_N$ 。第二实施方式的整数编码单元210基于与作为2以上的自然数的编码参数K对应地预先创建的、在整数编码单元210内存储的对应表,对输入的整数序列 $x_1, x_2, \dots, x_N$ 进行编码得到代码,将得到的代码作为整数代码输出(S210)。

[0139] [[用于编码处理的对应表]]

[0140] 预先存储的对应表,存储了基于零值的连续(0个以上的连续)和一个非零值的组



合的整数值的部分序列、和与该部分序列对应的代码的对应关系的规则。在图9 (A,B) 和图10 (A,B,C,D) 中分别举出 $K=2$ 和 $K=3$ 中的对应表的例子。在对应表中存储的规则,是如下的规则:对于基于 $2^{K-1}$ 个连续的零值的部分序列分配比特值为‘1’的1比特的代码,对于除此以外的部分序列、即基于0个至 $2^{K-1}-1$ 个连续的零值和其后续的一个非零值 $x_n$ 的部分序列,对于各个零值的连续和非零值 $x$ 的组合分配 $K \times x_n$ 比特或者 $K \times x_n + 1$ 比特的代码。

[0141] 在例示的对应表中,是如下规则:对于基于0个至 $2^{K-1}-1$ 个连续的零值和其后续的一个非零值 $x_n$ 的部分序列的各个部分序列,分配在开始的 $K$ 比特内至少存在1比特的‘1’、剩余的 $K \times (x_n - 1)$ 比特或者 $K \times (x_n - 1) + 1$ 比特全部为‘0’的代码。更详细而言,在该规则中,对于基于0个至 $2^{K-1}-1$ 个连续的零值和一个非零值的部分序列,分配第1比特的比特值为‘1’、第2比特至第 $K$ 比特为止的比特值全部为‘0’、第 $K+1$ 比特至第 $K \times x_n + 1$ 比特为止的比特值全部为‘0’的 $K \times x_n + 1$ 比特的代码、或者分配第1比特的比特值为‘0’、第2比特至第 $K$ 比特为止的 $K-1$ 比特中的至少1比特的比特值为‘1’、第 $K+1$ 比特至第 $K \times x_n$ 比特为止的比特值全部为‘0’的 $K \times x_n$ 比特的代码。更详细而言,在该规则中,基于 $K \times x_n + 1$ 比特的代码中的第1比特的比特值为‘1’、第2比特至第 $K+1$ 比特目为止为‘0’、即从第1比特至第 $K+1$ 比特为止的部分代码1种类、和 $K \times x_n$ 比特的代码中的第1比特的比特值为‘0’、第2比特至第 $K$ 比特为止的 $K-1$ 比特中的至少1比特的比特值为‘1’、即第1比特至第 $K$ 比特为止的部分代码 $2^{K-1}-1$ 种类的 $2^{K-1}$ 种的部分代码、和从0个至 $2^{K-1}-1$ 个零值的连续数 $2^{K-1}$ 种类一一对地对应。

[0142] [[使用了对应表的编码处理]]

[0143] 整数编码单元210使用上述的对应表,将所输入的整数序列 $x_1, x_2, \dots, x_N$ 中包含的各整数值按照预先确定的顺序编码得到代码,将得到的代码作为整数代码输出。例如,在预先确定的顺序为序列中的样本号码顺序的情况下,整数编码单元210进行的编码步骤如下。

[0144] (编码步骤)

[0145] 1. 将样本号码 $n$ 设为1。

[0146] 2. 直至成为 $n=N$ 为止,重复以下的(i)~(ii)。

[0147] (i) 在 $x_n$ 以后的 $2^{K-1}$ 个整数值中存在非零值的情况下,从代码表得到与至 $x_n$ 以后的最初的非零值为止的部分序列对应的代码,将对 $n$ 加上至 $x_n$ 以后的最初的非零值为止的样本数得到的结果设为新的 $n$ 。

[0148] (ii) 在 $x_n$ 以后的 $2^{K-1}$ 个整数值全部为零值的情况下,从代码表得到与 $x_n$ 以后的 $2^{K-1}$ 个整数值的部分序列对应的代码,将对 $n$ 加上 $2^{K-1}$ 后的结果设为新的 $n$ 。

[0149] 3. 在通过2得到的代码的最后加上代码‘1’(末端代码)。

[0150] 4. 将通过3得到的代码作为整数代码输出。

[0151] 而且,与第一实施方式同样,通过从编码装置200至解码装置250的整数代码的传递方法,也可以不进行加上末端代码的处理(编码步骤中的3)。

[0152] 而且,通过第二实施方式的编码装置200的整数编码单元210中使用了图9 (A,B) 和图10 (A,B,C,D) 的编码处理得到的代码,分别与第一实施方式的编码装置100的整数编码单元110中使用图4和图5的代码树的编码处理中、在将最初使用的代码树设为了 $T(1)$ 的情况下得到的代码一致。而且,在第二实施方式的编码装置200的整数编码单元210中,在使用图9 (A,B) 和图10 (A,B,C,D) 各自进行编码时,只要在输入的整数序列 $x_1, x_2, \dots, x_N$ 的开头

追加  $(2^{k-1}-1)$  个零值之后进行编码,并去除位于得到的整数代码的最初的代码‘1’,就与在第一实施方式的编码装置100的整数编码单元110中使用图4和图5各自的代码树的编码处理中,将最初使用的代码树设为T(0)的情况下得到的代码一致。这样,第二实施方式的编码装置200的整数编码单元210中的使用了对应表的编码处理,与第一实施方式的编码装置100的整数编码单元110中使用了代码树的编码处理是等效的。

[0153] 《解码装置》

[0154] 参照图7以及图8,说明第二实施方式的解码装置250执行的解码方法的处理过程。第二实施方式的解码装置250如图7所示,与第一实施方式的解码装置150同样,包含整数解码单元260。第二实施方式的解码装置250与第一实施方式的解码装置150同样,通过执行图8所示的各步骤的处理,实现第二实施方式的解码方法。

[0155] 在第二实施方式的解码装置250中,与第一实施方式的解码装置150同样,被输入第二实施方式的编码装置200输出的整数代码。第二实施方式的解码装置250,对于输入的整数代码,通过在使用了第二实施方式的编码装置200的整数编码单元210中存储的对应表和对对应表的解码处理进行解码得到非负的整数值的序列,将输入到第二实施方式的编码装置200的非负的整数值的序列本身进行还原。以下,与上述的编码装置的说明同样,将非负的整数值仅称为“整数值”进行说明。

[0156] [整数解码单元260]

[0157] 在整数解码单元260中被输入在解码装置250中输入的整数代码。整数解码单元260基于预先创建并存储在整数解码单元260内的、与编码参数K对应的预先创建的对应表,即与在编码装置200的整数编码单元210内预先存储并在整数编码单元210中使用的对应表相同的对应表,对输入的整数代码进行解码,得到整数序列 $x_1, x_2, \dots, x_N$ 并输出(S260)。

[0158] 更具体地说,整数解码单元260,将从最初至存在与对应表对应的代码的部分为止,读入在输入的整数代码中的还未读入的代码部分,从对应表中得到与读入的代码对应的整数的部分序列的处理,从被输入的整数代码的最初到最后为止反复进行,将由此得到的整数的部分序列汇总后的序列作为整数序列 $x_1, x_2, \dots, x_N$ 输出。通过该解码处理,能够得到与第一实施方式的解码装置150的整数解码单元160中使用图4和图5各自的代码树的解码处理中、将最初使用的代码树设为T(1)的情况相同的整数序列 $x_1, x_2, \dots, x_N$ 。

[0159] 而且,在编码装置200的整数编码单元210中,在输入的整数序列 $x_1, x_2, \dots, x_N$ 的开头追加  $(2^{k-1}-1)$  个零值之后进行编码,进行将位于得到的整数代码的最初的代码‘1’去除的情况下,整数解码单元260在被输入的整数代码的开头追加了代码‘1’后,基于上述对应表在上述的解码处理中得到各部分序列,省略了开始得到的  $(2^{k-1}-1)$  个零值后,将汇总了部分序列后的整数序列 $x_1, x_2, \dots, x_N$ 输出即可。通过该解码处理,能够得到与第一实施方式的解码装置150的整数解码单元160中的使用图4和图5各自的代码树的解码处理中、将最初使用的代码树设为了T(0)的情况相同的整数序列 $x_1, x_2, \dots, x_N$ 。这样,第二实施方式的解码装置250的整数解码单元260中的使用了对应表的解码处理,是与第一实施方式的解码装置150的整数解码单元160中的使用了代码树的解码处理等效的。

[0160] 从以上可知,也可以将通过第一实施方式的编码装置100在使用了代码树的编码处理中得到的整数代码,在通过第二实施方式的解码装置250使用了与该编码处理对应的对应表的解码处理中进行解码,得到整数序列 $x_1, x_2, \dots, x_N$ ,或者将通过第二实施方式

的编码装置200在使用了对应表的编码处理中得到的整数代码,在通过第一实施方式的解码装置150使用了与该编码处理对应的代码树的解码处理中进行解码,得到整数序列 $x_1, x_2, \dots, x_N$ 。

[0161] <第三实施方式>

[0162] 在第一实施方式以及第二实施方式中,从使用预先确定的代码树或者预先确定的对应表进行编码解码的过程的侧面,说明了实现本发明的编码和解码的方式。在第三实施方式中,从通过这些过程得到的代码或整数值的序列这样的数据的侧面,说明实现本发明的编码和解码的方式。

[0163] 《编码装置》

[0164] 参照图2以及图3,说明第三实施方式的编码装置300执行的编码方法的处理过程。第三实施方式的编码装置300如图2所示,与编码装置100或编码装置200同样,包含整数编码单元310。第三实施方式的编码装置300与编码装置100或编码装置200同样,通过执行图3所示的各步骤的处理,实现第三实施方式的编码方法。

[0165] 在第三实施方式的编码装置300中,与编码装置100或编码装置200同样,被输入非负的整数值的序列(以下,称为“整数序列”)。以下,将被输入到编码装置300的非负的整数值仅称为“整数值”进行说明。

[0166] [整数编码单元310]

[0167] 整数编码单元310,作为与在整数序列中包含的整数值0的 $2^k-1$ 个中的预先确定的任意的个数的连续对应的代码,得到1比特的代码(以下,称为“代码A”),作为与基于整数序列中包含的、0个至 $2^{k-1}-1$ 个连续的整数值0、和1个0以外的整数值 $x_n$ 的组对应的代码,得到 $K \times x_n$ 比特或者 $K \times x_n + 1$ 比特的代码(以下,称为“代码B”) (S310)。其中, $K$ 表示为2以上的自然数的编码参数。具体地说,将每次 $N$ 样本的整数序列 $x_1, x_2, \dots, x_N$ ,使用在第一实施方式中说明的代码树或在第二实施方式中说明的对应表进行编码,得到代码A或代码B。这里,若将与代码A对应的整数值0的连续中所包含的0的个数设为 $L$  ( $2 \leq L \leq 2^{k-1}$ ),则在S310中得到的代码B,为与整数值0的0个至 $L-1$ 个、即预先确定的最大 $2^{k-1}-1$ 个连续和其后续的一个0以外的整数值 $x_n$ 对应的 $K \times x_n$ 比特或者 $K \times x_n + 1$ 比特的代码。

[0168] 而且,这里,假设比特值“ $x$ ”为比特值“1”、比特值“ $y$ ”为比特值“0”,代码A是比特值为“ $x$ ”的1比特的代码(参照图9B,图10D),代码B在从第1比特至第 $K$ 比特中包含至少一个比特值“ $x$ ”和至少一个比特值“ $y$ ”,并且从最后起的 $K \times (x_n - 1)$ 比特为比特值“ $y$ ”(参照图9A以及图9B,图10A~图10D)。而且,如前述那样,也可以设为比特值“ $x$ ”为比特值“0”,比特值“ $y$ ”为比特值“1”。

[0169] 进而,关于代码B,更详细而言,在整数序列中的整数值0的0个至 $2^{k-1}-1$ 个的 $2^{k-1}$ 种类连续个数、和以下的 $2^{k-1}$ 种类的部分代码之间存在一对一的对应关系

[0170] (代码B的部分代码)

[0171] (1) 代码B为 $K \times x_n + 1$ 比特时的、由该代码B的第1比特至第 $K+1$ 比特为止的 $K+1$ 比特构成、第1比特的比特值为“ $x$ ”、从第2比特至第 $K+1$ 比特为止的的比特值为“ $y$ ”的、1种类的部分代码

[0172] (2) 代码B为 $K \times x_n$ 比特时的、由该代码B的第1比特至第 $K$ 比特为止的 $K$ 比特构成、第1比特的比特值为“ $y$ ”、第2比特至第 $K$ 比特为止的 $K-1$ 比特中的至少1比特的比特值为“ $x$ ”

的、 $2^{k-1}-1$ 种类的部分代码

[0173] 《解码装置》

[0174] 参照图7以及图8,说明第三实施方式的解码装置350执行的解码方法的处理过程。第三实施方式的解码装置350如图7所示,与解码装置150或解码装置250同样,包含整数解码单元360。第三实施方式的解码装置350与解码装置150或解码装置250同样,通过执行图8所示的各步骤的处理,实现第三实施方式的解码方法。

[0175] 在第三实施方式的解码装置350中,与解码装置150或解码装置250同样,被输入第三实施方式的编码装置300输出的代码串。以下,与前述的编码装置的说明同样,将非负的整数值仅称为“整数值”进行说明。

[0176] [整数解码单元360]

[0177] 整数解码单元360从在所输入的代码串中包含的1比特的代码(以下称为“代码A”),得到整数值0的2个至 $2^{k-1}$ 个中的预先确定的任意个数的连续,从在所输入的代码串中包含的 $K \times x_n$ 比特或者 $K \times x_{n+1}$ 比特的代码(以下,称为“代码B”)中,得到基于0个至 $2^{k-1}-1$ 个连续的整数值0、和1个0以外的整数值 $x_n$ 的组(S360)。其中, $K$ 表示作为2以上的自然数的编码参数。具体地说,对于所输入的代码串中包含的代码,使用在第一实施方式中说明的代码树或在第二实施方式中说明的对应表进行解码,得到与代码A或代码B对应的上述整数值的序列。这里,如果将从代码A得到的整数值0的连续中所包含的0的个数设为 $L$  ( $2 \leq L \leq 2^{k-1}$ ),则在S360中从代码B可得到的整数值0的连续中所包含的0的个数成为从0个至 $L-1$ 个。

[0178] 而且,这里,作为比特值“x”为比特值“1”、比特值“y”为比特值“0”,代码A是比特值为“x”的1比特的代码(参照图9B,图10D),代码B在第1比特至第 $K$ 比特中包含至少一个比特值“x”和至少一个比特值“y”,并且从最后起的 $K \times (x_n - 1)$ 比特为比特值“y”(参照图9A以及图9B,图10A~图10D)。而且,也可以与编码装置300同样,比特值“x”为比特值“0”,比特值“y”为比特值“1”。

[0179] 整数解码单元360通过进行以下的两个处理,能够确定在所输入的代码串中包含的代码A和代码B。具体地说,进行整数解码单元360从所输入的代码串的最初开始按顺序将所输入的代码串每次连续的 $K$ 比特(以下,称为“ $K$ 比特串”)地读入;和所读入的 $K$ 比特串是包含“y”的比特串时,识别为包含该“y”的 $K$ 比特串的紧跟前的比特是所述代码A或者所述代码B的末端的比特。

[0180] 进而,对于代码B,更详细而言,以下的 $2^{k-1}$ 种类的部分代码、和基于通过整数解码单元360得到的整数值的序列中的整数值0的0个至 $2^{k-1}-1$ 个的 $2^{k-1}$ 种类连续个数之间,存在一对一的对应关系。

[0181] (代码B的部分代码)

[0182] (1) 代码B为 $K \times x_{n+1}$ 比特时的、由该代码B的第1比特至第 $K+1$ 比特为止的 $K+1$ 比特构成、第1比特的比特值为“x”、从第2比特至第 $K+1$ 比特为止的比特值为“y”的、1种类的部分代码

[0183] (2) 代码B为 $K \times x_n$ 比特时的、由从该代码B的第1比特至第 $K$ 比特为止的 $K$ 比特构成、第1比特的比特值为“y”、从第2比特至第 $K$ 比特为止的 $K-1$ 比特中的至少1比特的比特值为“x”的、 $2^{k-1}-1$ 种类的部分代码

[0184] 《代码串的数据结构》

[0185] 说明第三实施方式的编码装置以及解码装置处理的代码串的数据结构,即作为第一实施方式以及第二实施方式的编码装置从整数值的序列中得到的代码串、且为第一实施方式以及第二实施方式的解码装置为了得到整数值的序列而使用的代码串的数据结构。

[0186] [代码串的数据结构]

[0187] 表示作为非负值的整数值的序列(以下称为“整数序列”)的代码串,在从整数序列得到代码串的编码装置或从代码串得到整数序列的解码装置中被使用。

[0188] 代码串的数据结构,包含:作为与整数序列中包含的整数值0的 $2$ 个至 $2^{K-1}$ 个中的预先确定的任意的个数的连续对应的代码的1比特的代码(以下,称为“代码A”)、和作为与基于在整数序列包含的、0个至 $2^{K-1}-1$ 个连续的整数值0、和1个0以外的整数值 $x_n$ 的组对应的代码的 $K \times x_n$ 比特或者 $K \times x_n + 1$ 比特的代码(以下,称为“代码B”)。其中, $K$ 表示作为2以上的自然数的编码参数。这里,如果将与代码A对应的整数值0的连续中包含的0的个数设为 $L$ ( $2 \leq L \leq 2^{K-1}$ ),则代码B是,与整数值0的0个至 $L-1$ 个、即预先确定的最大 $2^{K-1}-1$ 个连续和其后续的一个0以外的整数值 $x_n$ 对应的 $K \times x_n$ 比特或者 $K \times x_n + 1$ 比特的代码。

[0189] 而且,这里,假设比特值“x”为比特值“1”、比特值“y”为比特值“0”,代码A是比特值为“x”的1比特的代码(参照图9B,图10D),代码B在第1比特至第K比特中包含至少一个比特值“x”和至少一个比特值“y”,并且从最后起的 $K \times (x_n - 1)$ 比特是比特值为“y”(参照图9A以及图9B,图10A~图10D)。而且,如前述那样,也可以设为比特值“x”为比特值“0”,比特值“y”为比特值“1”。

[0190] 进而,对于代码B,更详细而言,在整数序列中的整数值0的0个至 $2^{K-1}-1$ 个 $2^{K-1}$ 种类的连续个数、和以下的 $2^{K-1}$ 种类的部分代码之间存在一对一的对应关系。

[0191] (代码B的部分代码)

[0192] (1) 代码B为 $K \times x_n + 1$ 比特时的、由该代码B的第1比特至第 $K+1$ 比特为止的 $K+1$ 比特构成、第1比特的比特值为“x”、第2比特至第 $K+1$ 比特为止的比特值为“y”的、1种类的部分代码

[0193] (2) 代码B为 $K \times x_n$ 比特时的、由该代码B的第1比特至第K比特为止的K比特构成、第1比特的比特值为“y”,第2比特至第K比特的 $K-1$ 比特中的至少1比特的比特值为“x”的、 $2^{K-1}-1$ 种类的部分代码

[0194] 根据本实施方式的发明,即使是还包含不是零值的小的值且大大偏向于小的值的整数值的序列、在非专利文献2的技术中值偏向至无法应对的程度的整数值的序列,也能够以较少的平均比特数进行编码。而且,能够对那样的代码进行解码。

[0195] <第四实施方式>

[0196] 如上述那样,用于本发明的代码的编码参数 $K$ ,其倒数的对数与格伦布-赖斯代码中的赖斯参数对应,本发明的代码能够作为相对于负值的赖斯参数的格伦布-赖斯代码来解释。因此,基于以往的最佳的赖斯参数的估计法,对于基于所输入的整数值的序列的部分序列求出最佳的编码参数 $K$ ,将使用了求出的编码参数 $K$ 的第一实施方式至第三实施方式的整数编码单元的处理与格伦布-赖斯编码组合,对所输入的整数值的序列进行编码也可以,进行与该编码对应的解码也可以。将该方式作为第四实施方式进行说明。

[0197] 《编码装置》

[0198] 参照图11以及图12,说明第四实施方式的编码装置400执行的编码方法的处理过

程。第四实施方式的编码装置400如图11所示,包含参数决定单元420和整数编码单元410。第四实施方式的编码装置400通过执行图12所示的各步骤的处理,实现第四实施方式的编码方法。

[0199] 在第四实施方式的编码装置400中,与第一实施方式至第三实施方式同样,被输入非负的整数值的序列。第四实施方式的编码装置400中被输入的非负的整数值的序列,每次N样本地被输入到参数决定单元420和整数编码单元410。即,第四实施方式的编码装置400例如将由麦克风拾音的语音或音乐等转换为数字信号进行量化,将基于得到的整数值的绝对值的序列等的整数值的序列对基于N样本的每个帧进行编码。

[0200] 以下,将在本实施方式的编码装置中输入的非负的整数值仅称为“整数值”进行说明。

[0201] [参数决定单元420]

[0202] 参数决定单元420中,被输入在编码装置400中被输入的整数值的序列中的、每次N样本的整数序列 $x_1, x_2, \dots, x_N$ 。参数决定单元420基于所输入的整数序列 $x_1, x_2, \dots, x_N$ ,得到与该整数序列对应的赖斯参数 $r$ 和作为表示该参数的代码的参数代码并输出(S420)。参数代码对赖斯参数进行编码得到,使得解码装置450通过对该参数代码进行解码能够得到参数决定单元420决定的赖斯参数 $r$ 即可。

[0203] 参数决定单元420例如使用在被输入的整数序列 $x_1, x_2, \dots, x_N$ 中包含的各整数值,通过式(5)得到赖斯参数 $r$ 。

$$[0204] \quad r = \log_2 \left( \frac{\ln 2}{N} \sum_{n=1}^N x_n \right) \quad \dots(5)$$

[0205] 在式(5)中求的赖斯参数 $r$ ,是将对于整数序列 $x_1, x_2, \dots, x_N$ 通过式(2)估计的格伦布-赖斯编码时的总比特长的估计值进行最小化的值。

[0206] 然后,参数决定单元420例如将通过式(5)得到的赖斯参数 $r$ 进行标量量化得到代码,将得到的代码作为参数代码输出,并将与该参数代码对应的赖斯参数 $r$ 的量化值作为赖斯参数 $r$ 输出。

[0207] [整数编码单元410]

[0208] 在整数编码单元410中,被输入在编码装置400中所输入的整数值的序列中的、每次N样本的整数序列 $x_1, x_2, \dots, x_N$ 、和参数决定单元420输出的赖斯参数 $r$ 。整数编码单元410在赖斯参数 $r$ 为非负值的情况下,通过将赖斯参数 $r$ 的小数第一位进行四舍五入等,将赖斯参数 $r$ 舍入成整数值后,对于整数序列 $x_1, x_2, \dots, x_N$ 进行基于整数值的赖斯参数 $r$ 的格伦布-赖斯编码,将通过格伦布-赖斯编码得到的代码作为整数代码输出,在赖斯参数 $r$ 为负值的情况下,从赖斯参数 $r$ 通过 $K = 2^{-r}$ 得到 $K$ 的值,通过将得到的 $K$ 的值的的小数第一位进行四舍五入等,将 $K$ 的值舍入成为整数值,从而得到编码参数 $K$ ,使用得到的编码参数 $K$ ,通过与第一实施方式至第三实施方式的整数编码单元的任一处理相同的处理,得到整数代码并输出(S410)。而且,在编码参数 $K$ 为1和2之间的值的情况下,例如使舍入后的整数值为2即可。

[0209] 《解码装置》

[0210] 参照图13以及图14,说明第四实施方式的解码装置450执行的解码方法的处理过程。第四实施方式的解码装置450如图13所示,包含参数解码单元470和整数解码单元460。

第四实施方式的解码装置450通过执行图14所示的各步骤的处理,实现第四实施方式的解码方法。

[0211] 第四实施方式的解码装置450中被输入第四实施方式的编码装置400输出的整数代码、以及参数代码。在解码装置450中输入的整数代码以及参数代码,对于与整数值的序列的N样本对应的每个代码,被输入到参数解码单元470和整数解码单元460。即,解码装置450对于与编码装置400相同的每个帧进行解码处理。

[0212] [参数解码单元470]

[0213] 参数解码单元470中,被输入在解码装置450中被输入的参数代码。参数解码单元470通过与参数决定单元420得到参数代码对应的解码处理,对参数代码进行解码,得到赖斯参数 $r$ 并输出(S470)。参数解码单元470例如通过与编码装置400的参数决定单元420进行的标量化对应的解码处理,得到与参数代码对应的赖斯参数 $r$ 的量化值作为赖斯参数 $r$ 并输出。

[0214] [整数解码单元460]

[0215] 整数解码单元460中,被输入解码装置450中所输入的整数代码、参数解码单元470输出的赖斯参数 $r$ 。整数解码单元460在赖斯参数 $r$ 为非负值的情况下,通过与编码装置400的整数编码单元410进行的方法相同的方法将赖斯参数 $r$ 舍入成整数值后,对于整数代码进行基于赖斯参数 $r$ 的格伦布-赖斯解码,得到整数序列 $x_1, x_2, \dots, x_N$ 并输出,在赖斯参数 $r$ 为负值的情况下,以与第四实施方式的整数编码单元410相同的处理将赖斯参数 $r$ 变换为编码参数 $K$ 而得到、即从赖斯参数 $r$ 通过 $K=2^{-r}$ 得到 $K$ 的值,通过与编码装置400的整数编码单元410进行的方法相同的方法,将 $K$ 的值舍入成整数值,从而得到编码参数 $K$ ,使用得到的编码参数 $K$ ,通过与第一实施方式至第三实施方式的整数解码单元的任一处理相同的处理,得到整数序列 $x_1, x_2, \dots, x_N$ 并输出(S460)。

[0216] <第四实施方式的变形例1>

[0217] 在第四实施方式中,进行了与格伦布-赖斯编码/解码组合了的编码/解码处理,但是也可以不与格伦布-赖斯编码/解码组合。将不与格伦布-赖斯编码/解码组合的方式作为第四实施方式的变形例1进行说明。编码装置401的参数决定单元421和解码装置451的参数解码单元471的动作与第四实施方式相同,编码装置401的整数编码单元411和解码装置451的整数解码单元461的动作与第四实施方式不同。

[0218] [整数编码单元411]

[0219] 整数编码单元411中,被输入在编码装置401中被输入的整数值的序列中的、每次N样本的整数序列 $x_1, x_2, \dots, x_N$ 、和参数决定单元421输出的赖斯参数 $r$ 。整数编码单元411在赖斯参数 $r$ 为非负值的情况下,使用2作为编码参数 $K$ ,通过与从第一实施方式至第三实施方式的整数编码单元的任一处理相同的处理得到整数代码并输出,在赖斯参数 $r$ 为负值的情况下,从赖斯参数 $r$ 通过 $K=2^{-r}$ 得到 $K$ 的值,通过将得到的 $K$ 的值的的小数第一位进行四舍五入等,将 $K$ 的值舍入成整数值,从而得到编码参数 $K$ ,使用得到的编码参数 $K$ ,通过与第一实施方式至第三实施方式的整数编码单元的任一处理相同的处理得到整数代码并输出(S411)。而且,在编码参数 $K$ 为1和2之间的值的情况下,例如使舍入后的整数值为2即可。

[0220] [整数解码单元461]

[0221] 整数解码单元461中,被输入在解码装置451中被输入的整数代码、和参数解码单

元471输出的赖斯参数 $r$ 。整数解码单元461,在赖斯参数 $r$ 为非负值的情况下,将2用作编码参数 $K$ ,通过与第一实施方式至第三实施方式的整数解码单元的任一处理相同的处理,得到整数序列 $x_1, x_2, \dots, x_N$ 并输出,在赖斯参数 $r$ 为负值的情况下,以与第四实施方式的整数编码单元410相同的处理,将赖斯参数 $r$ 变换为编码参数 $K$ 而得到、即从赖斯参数 $r$ 通过 $K=2^{-r}$ 得到 $K$ 的值,通过与编码装置401的整数编码单元411进行的方法相同的方法,将 $K$ 的值舍入成整数,从而得到编码参数 $K$ ,使用得到的编码参数 $K$ ,通过与第一实施方式至第三实施方式的整数解码单元的任一处理相同的处理,得到整数序列 $x_1, x_2, \dots, x_N$ 并输出(S461)。

[0222] <第四实施方式的变形例2>

[0223] 在第四实施方式中,设为在整数编码单元410中赖斯参数 $r$ 为负值的情况下,从赖斯参数 $r$ 得到编码参数 $K$ 的结构,但是也可以设为在参数决定单元420中赖斯参数 $r$ 为负值的情况下从整数序列直接得到编码参数 $K$ 的结构。将该方式作为第四实施方式的变形例2,说明与第四实施方式不同的部分。

[0224] [参数决定单元422]

[0225] 第四实施方式的参数决定单元420是与赖斯参数的值无关地、一定输出赖斯参数和与该赖斯参数对应的参数代码的结构,但是第四实施方式的变形例2的参数决定单元422,在赖斯参数 $r$ 为非负值的情况下,输出赖斯参数 $r$ 和与该赖斯参数 $r$ 对应的参数代码,在赖斯参数 $r$ 为负值的情况下,通过后述的式(6),从整数序列得到编码参数 $K$ ,输出编码参数 $K$ 和与该编码参数 $K$ 对应的参数代码(S422)。此时,参数代码,设为对应的解码装置452能够唯一地确定赖斯参数 $r$ 的值和编码参数 $K$ 的值的代码即可。例如,通过将在对赖斯参数 $r$ 进行编码得到的代码之前附加1比特的代码‘1’后的结果作为参数代码,将在对编码参数 $K$ 进行编码得到的代码之前附加1比特的代码‘0’后的结果设为参数代码等,只要在分配给赖斯参数 $r$ 的参数代码的代码群、和分配给编码参数 $K$ 的代码群中不包含相同的代码即可。

[0226] [整数编码单元412]

[0227] 整数编码单元412中,被输入在编码装置402中被输入的整数值的序列中的、每次 $N$ 样本的整数序列 $x_1, x_2, \dots, x_N$ 、和参数决定单元422输出的赖斯参数 $r$ 或者编码参数 $K$ 。整数编码单元412在被输入了赖斯参数 $r$ 的情况下,将赖斯参数 $r$ 舍入成为整数,对于整数序列 $x_1, x_2, \dots, x_N$ ,进行基于整数值的赖斯参数 $r$ 的格伦布-赖斯编码,将通过格伦布-赖斯编码得到的代码作为整数代码输出,在被输入了编码参数 $K$ 的情况下,将编码参数 $K$ 舍入成整数,使用整数值的编码参数 $K$ ,通过与从第一实施方式至第三实施方式的整数编码单元的任一处理相同的处理,得到整数代码并输出(S412)。而且,在编码参数 $K$ 为1和2之间的值的情况下,例如使舍入后的整数值为2即可。

[0228] [参数解码单元472]

[0229] 第四实施方式的参数解码单元470是对参数代码进行解码得到赖斯参数的单元,但是第四实施方式的变形例2的参数解码单元472对参数代码进行解码,得到赖斯参数 $r$ 或者编码参数 $K$ 并输出(S472)。

[0230] [整数解码单元462]

[0231] 整数解码单元462中,被输入在解码装置452中被输入的整数代码、和参数解码单元472输出的赖斯参数 $r$ 或者编码参数 $K$ 。整数解码单元462在被输入了赖斯参数 $r$ 的情况下,通过将赖斯参数 $r$ 利用与编码装置402的整数编码单元412进行的方法相同的方法舍入成为



整数值后,对于整数代码,进行基于整数值的赖斯参数 $r$ 的格伦布-赖斯解码,得到整数序列 $x_1, x_2, \dots, x_N$ 并输出,在被输入了编码参数 $K$ 的情况下,利用与编码装置402的整数编码单元412进行的方法相同的方法将编码参数 $K$ 舍入成为整数值后,使用整数值的编码参数 $K$ ,通过与第一实施方式至第三实施方式的整数解码单元的任一处理相同的处理,得到整数序列 $x_1, x_2, \dots, x_N$ 并输出(S462)。

[0232] <第五实施方式>

[0233] 在第四实施方式的编码装置中,首先求出赖斯参数 $r$ ,从求出的赖斯参数 $r$ 求出编码参数 $K$ ,但是,也可以通过对本发明的代码为最佳的分布 $p(x, 1/K)$ 估计似然的 $K$ ,从输入的整数值的序列直接估计编码参数 $K$ 。在该情况下,格伦布-赖斯代码可以解释为基于取1以下的值的 $K$ 的本发明的代码。因此,从基于所输入的整数值的序列的部分序列直接求出最佳的编码参数 $K$ ,将使用了求出的编码参数 $K$ 的第一实施方式至第三实施方式的整数编码单元的处理与格伦布-赖斯编码组合,对输入的整数值的序列进行编码也可以,进行与该编码对应的解码也可以。将该方式作为第五实施方式进行说明。

[0234] 《编码装置》

[0235] 参照图11以及图12,说明第五实施方式的编码装置500执行的编码方法的处理过程。第五实施方式的编码装置500如图11所示,包含参数决定单元520和整数编码单元510。第五实施方式的编码装置500通过执行图12所示的各步骤的处理,实现第五实施方式的编码方法。

[0236] 第五实施方式的编码装置500中与第一实施方式至第四实施方式同样,被输入非负的整数值的序列。被输入到第五实施方式的编码装置500的非负的整数值的序列每次 $N$ 样本本地输入到参数决定单元520和整数编码单元510。即,第五实施方式的编码装置500例如将由麦克风拾音得到的语音或音乐等变换为数字信号进行量化,将基于得到的整数值的绝对值的序列等的整数值的序列对于基于 $N$ 样本的每个帧进行编码。

[0237] 以下,将被输入到本实施方式的编码装置的非负的整数值仅称为“整数值”进行说明。

[0238] [参数决定单元520]

[0239] 参数决定单元520中,被输入在编码装置500中所输入的整数值的序列中的、每次 $N$ 样本的整数序列 $x_1, x_2, \dots, x_N$ 。参数决定单元520基于被输入的整数序列 $x_1, x_2, \dots, x_N$ ,得到与该整数序列对应的编码参数 $K$ 和作为表示该参数的代码的参数代码并输出(S520)。参数代码对编码参数 $K$ 进行编码得到,使得通过解码装置550对该参数代码进行解码,能够得到参数决定单元520决定的编码参数 $K$ 即可。

[0240] 参数决定单元520例如使用在所输入的整数序列 $x_1, x_2, \dots, x_N$ 中包含的各整数值,通过式(6)得到编码参数 $K$ 。基于式(4),求出相对于分布 $p(x, 1/K)$ 的整数序列 $x_1, x_2, \dots, x_N$ 的对数似然性,得到如下式那样将对数似然性进行最大化的编码参数 $K$ 并输出。

$$[0241] \quad K = \log_2 \left( 1 + \frac{1}{\frac{1}{N} \sum_{n=1}^N x_{-n}} \right) \cdots (6)$$

[0242] 式(6)中求出的编码参数K,是将关于整数序列 $x_1, x_2, \dots, x_N$ 的相对于基于式(4)的分布 $p(x, 1/K)$ 的对数似然性进行最大化的值。

[0243] 然后,参数决定单元520例如将通过式(6)得到的编码参数K进行标量量化得到代码,将得到的代码作为参数代码输出,并将与该参数代码对应的编码参数K的量化值作为编码参数K进行输出。

[0244] [整数编码单元510]

[0245] 整数编码单元510中,被输入在编码装置500中被输入的整数值的序列中的、每次N样本的整数序列 $x_1, x_2, \dots, x_N$ 、和参数决定单元520输出的编码参数K。整数编码单元510,在编码参数K为1以下的情况下,从编码参数K通过 $r = -\log_2 K$ 得到r的值,通过将得到的r的值的小数第一位进行四舍五入等,将r的值舍入成为整数值,从而得到赖斯参数r,使用得到的赖斯参数r,对整数序列 $x_1, x_2, \dots, x_N$ 进行基于赖斯参数r的格伦布-赖斯编码,将通过格伦布-赖斯编码得到的代码作为整数代码输出,在编码参数K大于1的情况下,通过将编码参数K的小数第一位进行四舍五入等,将K的值舍入成为整数值后,使用整数值的编码参数K,通过与第一实施方式至第三实施方式的整数编码单元的任一处理相同的处理,得到整数代码并输出(S510)。而且,在编码参数K为1和2之间的值的情况下,例如使舍入后的整数值为2即可。

[0246] 《解码装置》

[0247] 参照图13以及图14,说明第五实施方式的解码装置550执行的解码方法的处理过程。第五实施方式的解码装置550如图13所示,包含参数解码单元570和整数解码单元560。第五实施方式的解码装置550通过执行图14所示的各步骤的处理,实现第五实施方式的解码方法。

[0248] 第五实施方式的解码装置550中,被输入第五实施方式的编码装置500输出的整数代码、以及参数代码。被输入到解码装置550的整数代码以及参数代码,对于与整数值的序列的N样本对应的每个代码,被输入到参数解码单元570和整数解码单元560。即,解码装置550对于与编码装置500相同的每个帧进行解码处理。

[0249] [参数解码单元570]

[0250] 参数解码单元570中,被输入在解码装置550中被输入的参数代码。参数解码单元570通过与参数决定单元520得到参数代码对应的解码处理,对参数代码进行解码,得到编码参数K并输出(S570)。参数解码单元570例如通过与编码装置500的参数决定单元520进行的标量量化对应的解码处理,得到与参数代码对应的编码参数K的量化值作为编码参数K并输出。

[0251] [整数解码单元560]

[0252] 整数解码单元560中,被输入在解码装置550中被输入的整数代码、和参数解码单元570输出的编码参数K。整数解码单元560在编码参数K为1以下的情况下,以与第五实施方式的整数编码单元510相同的处理,将编码参数K变换为赖斯参数r而得到、即从编码参数K通过 $r = -\log_2 K$ 得到r的值,通过与编码装置500的整数编码单元510进行的方法相同的方法,将r的值舍入成为整数值,从而得到赖斯参数r,使用得到的赖斯参数r,对整数代码进行基于赖斯参数r的格伦布-赖斯解码,得到整数序列 $x_1, x_2, \dots, x_N$ 并输出,在编码参数K大于1的情况下,通过与编码装置500的整数编码单元510进行的方法相同的方法将编码参数K舍入成为整数值后,使用整数值的编码参数K,通过与第一实施方式至第三实施方式的整数解码单元的任一处理相同的处理,得到整数序列 $x_1, x_2, \dots, x_N$ 并输出 (S560)。

[0253] <第五实施方式的变形例1>

[0254] 在第五实施方式中,进行了与格伦布-赖斯编码/解码组合了的编码/解码处理,但是也可以不与格伦布-赖斯编码/解码组合。将不与格伦布-赖斯编码/解码组合的方式作为第五实施方式的变形例1进行说明。编码装置501的参数决定单元521和解码装置551的参数解码单元571的动作与第五实施方式相同,编码装置501的整数编码单元511和解码装置551的整数解码单元561的动作与第五实施方式不同。

[0255] [整数编码单元511]

[0256] 整数编码单元511中,被输入在编码装置501中被输入的整数值的序列中的、每次N样本的整数序列 $x_1, x_2, \dots, x_N$ 、和参数决定单元521输出的编码参数K。整数编码单元511通过将被输入的编码参数K的小数第一位进行四舍五入等,将K的值舍入成为整数值后,在整数值的编码参数K为1以下的情况下,使用2作为编码参数K,通过与第一实施方式至第三实施方式的整数编码单元的任一处理相同的处理,得到整数代码并输出,在整数值的编码参数K大于1的情况下,使用整数值的编码参数K,通过与第一实施方式至第三实施方式的整数编码单元的任一处理相同的处理,得到整数代码并输出 (S511)。

[0257] [整数解码单元561]

[0258] 整数解码单元561中,被输入在解码装置551中被输入的整数代码、和参数解码单元571输出的编码参数K。整数解码单元561将被输入的编码参数K,通过与编码装置501的整数编码单元511进行的方法相同的方法舍入成为整数值后,在整数值的编码参数K为1以下的情况下,使用2作为编码参数K,通过与第一实施方式至第三实施方式的整数解码单元的任一处理相同的处理,得到整数序列 $x_1, x_2, \dots, x_N$ 并输出,在整数值的编码参数K大于1的情况下,使用整数值的编码参数K,通过与第一实施方式至第三实施方式的整数解码单元的任一处理相同的处理,得到整数序列 $x_1, x_2, \dots, x_N$ 并输出 (S561)。

[0259] <第五实施方式的变形例2>

[0260] 在第五实施方式中,设为在整数编码单元510中编码参数K为1以下的情况下,从编码参数K得到赖斯参数r的结构,在参数决定单元520中编码参数K为1以下的情况下,也可以设为从整数序列直接得到赖斯参数r的结构。将该方式作为第五实施方式的变形例2,说明与第五实施方式不同的部分。

[0261] [参数决定单元522]

[0262] 第五实施方式的参数决定单元520,是与编码参数K的值无关地,一定输出编码参数K和与该编码参数K对应的参数代码的结构,但是第五实施方式的变形例2的参数决定单

元522,在编码参数K大于1的情况下,输出编码参数K和与该编码参数K对应的参数代码,在编码参数K为1以下的情况下,通过式(5)从整数序列得到赖斯参数r,输出赖斯参数r和与该赖斯参数r对应的参数代码(S522)。此时,与第四实施方式的变形例2同样,参数代码设为对应的解码装置552能够唯一地确定赖斯参数r的值和编码参数K的值的代码即可。

[0263] [整数编码单元512]

[0264] 整数编码单元512中,被输入在编码装置502中被输入的整数值的序列中的、每次N样本的整数序列 $x_1, x_2, \dots, x_N$ 、和参数决定单元522输出的编码参数K或者赖斯参数r。整数编码单元512在被输入了赖斯参数r的情况下,将赖斯参数r的值舍入成为整数值后,对整数序列 $x_1, x_2, \dots, x_N$ ,进行基于整数值的赖斯参数r的格伦布-赖斯编码,将通过格伦布-赖斯编码得到的代码作为整数代码输出,在被输入了编码参数K的情况下,将编码参数K舍入成为整数值后,使用整数值的编码参数K,通过与第一实施方式至第三实施方式的整数编码单元的任一处理相同的处理,得到整数代码并输出(S512)。而且,在编码参数K为1和2之间的值的情况下,例如使舍入后的整数值设为2即可。

[0265] [参数解码单元572]

[0266] 第五实施方式的参数解码单元570是将参数代码进行解码得到编码参数K的单元,而第五实施方式的变形例2的参数解码单元572对参数代码进行解码得到编码参数K或者赖斯参数r并输出(S572)。

[0267] [整数解码单元562]

[0268] 整数解码单元562中,被输入在解码装置552中被输入的整数代码、和参数解码单元572输出的编码参数K或者赖斯参数r。整数解码单元562在被输入了赖斯参数r的情况下,通过与编码装置502的整数编码单元512进行的方法相同的方法,将赖斯参数r的值舍入作为整数值,对整数代码,进行基于整数值的赖斯参数r的格伦布-赖斯解码,得到整数序列 $x_1, x_2, \dots, x_N$ 并输出,在被输入了编码参数K的情况下,将编码参数K通过与编码装置502的整数编码单元512进行的方法相同的方法舍入成为整数值后,使用整数值的编码参数K,通过与第一实施方式至第三实施方式的整数解码单元的任一处理相同的处理,得到整数序列 $x_1, x_2, \dots, x_N$ 并输出(S562)。

[0269] <第四实施方式以及第五实施方式的其它变形>

[0270] 而且,赖斯参数r和编码参数K,通过 $K=2^r$ 以及 $r=-\log_2 K$ 可相互进行变换,因此在编码装置的整数编码单元和解码装置的整数解码单元中,只要能够使用相同的参数的相同的值,则将哪个参数设为编码/解码的对象和将上述的变换怎样组合都可以。而且,只要能够在编码装置的整数编码单元和解码装置的整数解码单元中使用相同的参数的相同的值,则将赖斯参数r和编码参数K的哪个值作为舍入的处理的对象都可以,将参数的编码前后(即量化前后)的哪个值作为舍入的处理的对象也可以,同时进行编码中的值的量化和舍入的处理也可以。总之,第四实施方式以及第五实施方式以及在这些变形例1和变形例2中说明的最终是一例,对于基于N样本的每个帧,只要在编码装置的整数编码单元和解码装置的整数解码单元中能够进行使用相同的参数的相同的值的编码处理和解码处理即可。

[0271] <第六实施方式>

[0272] 在第四实施方式以及第五实施方式中,说明了对于基于输入的整数值的序列的部分序列,使用求出的赖斯参数r或者编码参数K,进行从第一实施方式至第三实施方式的编

码/解码,实现本发明的编码和解码的方式。在第六实施方式中,说明使用赖斯参数 $r$ 和编码参数 $K$ 的至少其中一个,实现本发明的编码和解码的方式。

[0273] 《编码装置》

[0274] 参照图11以及图12,说明第六实施方式的编码装置600执行的编码方法的处理过程。第六实施方式的编码装置600如图11所示,与编码装置400或编码装置500同样,包含参数决定单元620和整数编码单元610。第六实施方式的编码装置600与编码装置400或编码装置500同样,通过执行图12所示的各步骤的处理,实现第六实施方式的编码方法。

[0275] 在第六实施方式的编码装置600中,与编码装置400或编码装置500同样,被输入非负的整数值的序列。被输入到第六实施方式的编码装置600的非负的整数值的序列每次 $N$ 本地被输入到参数决定单元620和整数编码单元610。

[0276] 以下,仅将被输入到本实施方式的编码装置的非负的整数值称为“整数值”进行说明。

[0277] [参数决定单元620]

[0278] 参数决定单元620中,被输入在编码装置600中被输入的整数值的序列中的、每次 $N$ 样本的整数序列 $x_1, x_2, \dots, x_N$ 。参数决定单元620基于被输入的整数序列 $x_1, x_2, \dots, x_N$ ,得到与该整数序列对应的赖斯参数 $r$ 和编码参数 $K$ 的至少其中一个、和表示该赖斯参数 $r$ 和编码参数 $K$ 的至少其中一个的参数代码并输出(S620)。即,参数决定单元620进行:对于基于被输入的非负的整数值的序列中包含的基于每个规定样本的整数值的序列即每个整数序列,基于整数序列中包含的整数值,通过式(5)求出赖斯参数 $r$ 的处理、和基于整数序列中包含的整数值,通过式(6)求出编码参数 $K$ 的处理的至少其中一个;和对赖斯参数 $r$ 和编码参数 $K$ 的至少其中一个进行编码得到参数代码的处理。

[0279] [整数编码单元610]

[0280] 整数编码单元610中,被输入在编码装置600中被输入的整数值的序列中的、每次 $N$ 样本的整数序列 $x_1, x_2, \dots, x_N$ 、和参数决定单元620输出的赖斯参数 $r$ 和编码参数 $K$ 的至少其中一个。整数编码单元610在被输入了编码参数 $K$ 的情况下,如果编码参数 $K$ 为1以下,则与第五实施方式同样,进行格伦布-赖斯编码,将通过格伦布-赖斯编码得到的代码作为整数代码进行输出,如果编码参数 $K$ 大于1,则与第五实施方式同样,通过与第一实施方式至第三实施方式的整数编码单元的任一处理相同的处理,得到整数代码并输出,在被输入了赖斯参数 $r$ 的情况下,如果赖斯参数 $r$ 为非负值,则与第四实施方式同样,进行格伦布-赖斯编码,将通过格伦布-赖斯编码得到的代码作为整数代码输出,如果赖斯参数 $r$ 为负值,则与第四实施方式同样,通过与第一实施方式至第三实施方式的整数编码单元的任一处理相同的处理,得到整数代码并输出(S610)。即,整数编码单元610对于基于每规定样本的整数值的序列即每个整数序列,使用参数决定单元620得到的编码参数 $K$ 、或者从参数决定单元620得到的赖斯参数 $r$ 通过 $K=2^{-r}$ 得到的编码参数 $K$ 作为编码参数 $K$ ,得到代码A或代码B的代码。特别地,也可以在与整数序列对应的编码参数 $K$ 为1以下的情况下,或者在与整数序列对应的赖斯参数 $r$ 为非负的情况下,对整数序列进行格伦布-赖斯编码,得到代码串。

[0281] 《解码装置》

[0282] 参照图13以及图14,说明第六实施方式的解码装置650执行的解码方法的处理过程。第六实施方式的解码装置650如图13所示,与解码装置450或解码装置550同样,包含参

数解码单元670和整数解码单元660。第六实施方式的解码装置650与解码装置450或解码装置550同样,通过执行图14所示的各步骤的处理,实现第六实施方式的解码方法。

[0283] 第六实施方式的解码装置650中,被输入第六实施方式的编码装置600输出的整数代码、以及参数代码。被输入到解码装置650的整数代码以及参数代码,对于与整数值的序列的N样本对应的每个代码,被输入到参数解码单元670和整数解码单元660。即,参数解码单元670和整数解码单元660中被输入每规定单位的参数代码和代码串。

[0284] [参数解码单元670]

[0285] 参数解码单元670通过与参数决定单元620得到参数代码对应的解码处理,对参数代码进行解码,得到赖斯参数 $r$ 和编码参数 $K$ 的至少其中一个并输出(S670)。即,参数解码单元670对于每个规定单位,对参数代码进行解码,得到赖斯参数 $r$ 和编码参数 $K$ 的至少其中一个。

[0286] [整数解码单元660]

[0287] 整数解码单元660中被输入在解码装置650中被输入的整数代码、参数解码单元670输出的赖斯参数 $r$ 和编码参数 $K$ 中的至少其中一个。整数解码单元660在被输入了编码参数 $K$ 的情况下,如果编码参数 $K$ 为1以下,则与第五实施方式同样,进行格伦布-赖斯解码,得到整数序列 $x_1, x_2, \dots, x_N$ 并输出,如果编码参数 $K$ 大于1,则与第五实施方式同样,通过与第一实施方式至第三实施方式的整数解码单元的其中一个处理相同的处理,得到整数序列 $x_1, x_2, \dots, x_N$ 并输出,在被输入了赖斯参数 $r$ 的情况下,如果赖斯参数 $r$ 为非负值,则与第四实施方式同样,进行格伦布-赖斯解码,得到整数序列 $x_1, x_2, \dots, x_N$ 并输出,如果赖斯参数 $r$ 为负值,则与第四实施方式同样,通过与第一实施方式至第三实施方式的整数解码单元的其中一个处理相同的处理,得到整数序列 $x_1, x_2, \dots, x_N$ 并输出(S660)。即,整数解码单元660对于每个规定单位,将参数解码单元670得到的编码参数 $K$ 、或者从参数解码单元670得到的赖斯参数 $r$ 通过 $K=2^{-r}$ 得到的编码参数 $K$ 使用作为编码参数 $K$ ,得到由整数值0的2个至 $2^{K-1}$ 个连续、或基于0个至 $2^{K-1}-1$ 个连续的整数值0和1个0以外的整数值 $x_n$ 的组构成的、基于整数值的序列。特别地,也可以在编码参数 $K$ 为1以下的情况下、或者在赖斯参数 $r$ 为非负的情况下,将代码串进行格伦布-赖斯解码,得到基于整数值的序列。

[0288] 根据本实施方式的发明,如果知道与整数值的序列或部分序列对应的赖斯参数 $r$ 相当的值,则通过对该序列使用最佳的编码参数 $K$ ,能够以较少的平均比特数进行编码。而且,能够将那样的代码进行解码。

[0289] <补记>

[0290] 本发明的装置例如作为单一的硬件实体,具有:可连接键盘等的输入单元、可连接液晶显示器等的输出单元、可连接能够对硬件实体的外部进行通信的通信装置(例如通信电缆)的通信单元、CPU(也可以具有中央处理单元(Central Processing Unit)、闪速存储器或寄存器等)、作为存储器的RAM或ROM、作为硬盘的外部存储装置、以及进行连接以在这些输入单元、输出单元、通信单元、CPU、RAM、ROM、外部存储装置之间进行数据交换的总线。另外,根据需要,在硬件实体中也可以设置能够读写CD-ROM等的记录介质的装置(驱动器)等。作为具有这样的硬件资源的物理的实体,有通用计算机等。

[0291] 在硬件实体的外部存储装置中,存储了用于实现上述的功能所需要的程序以及在该程序的处理中所需要的数据等(不限于外部存储装置,例如也可以预先读出程序,使其存

储在作为专用存储装置的ROM)。而且,通过这些程序的处理所得到的数据等,被适当地存储在RAM或外部存储装置等。

[0292] 在硬件实体中,在外部存储装置(或者ROM等)所存储的各程序和该各程序的处理所需要的数据根据需要被读入到存储器,适当地在CPU中进行解释执行/处理。其结果,CPU实现规定的功能(作为上述…单元、…部件等表示的各构成要件)。

[0293] 本发明没有被限定于上述的实施方式,在不脱离本发明的宗旨的范围内能够适当进行变更。而且,上述实施方式中说明的处理不仅根据记载的顺序时序地执行,也可以根据执行处理的装置的处理能力或者需要而并行或者单独地执行。

[0294] 如已述那样,在通过计算机实现上述实施方式中说明的硬件实体(本发明的装置)中的处理功能的情况下,硬件实体应该具有的功能性的处理内容通过程序被记述。然后,通过在计算机上执行该程序,在计算机上实现上述硬件实体中的处理功能。

[0295] 记述了该处理内容的程序,能够被预先记录在计算机可读的记录介质中。作为由计算机可读的记录介质,例如磁记录装置、光盘、光磁记录介质、半导体存储器等什么样的装置都可以。具体地说,例如,作为磁记录装置,可以使用硬盘装置、软盘、磁带等,作为光盘,可以使用DVD(Digital Versatile Disc,数字通用光盘)、DVD-RAM(Random Access Memory,随机存取存储器)、CD-ROM(Compact Disc Read Only Memory,压缩光盘只读存储器)、CD-R(Recordable,可记录)/RW(ReWritable,可改写)等,作为光磁记录介质,可以使用MO(Magneto-Optical disc,磁光盘)等,作为半导体存储器,可以使用EEP-ROM(Electronically Erasable and Programmable-Read Only Memory,电可擦可编程只读存储器)等。

[0296] 而且,例如通过贩卖、转让、出租记录了该程序的DVD、CD-ROM等可移动型记录介质等来进行该程序的流通。进而,也可以将该程序预先存储在服务器计算机的存储装置中,经由网络,将该程序通过从服务器计算机转发到其它计算机,使该程序流通。

[0297] 执行这样的程序的计算机,例如,首先将移动式记录介质中记录的程序或由服务器计算机转发的程序暂时存储在自身的存储装置中。在处理的执行时,该计算机读取自身的记录介质中存储的程序,执行根据读取出的程序的处理。另外,作为该程序的另一执行方式,也可以计算机从移动式记录介质中直接读取程序,执行根据读取出的程序的处理,而且,每次从服务器计算机对该计算机转发程序时,逐次执行根据接受的程序的处理。也可以是不从服务器计算机进行对该计算机的程序的转发,而仅通过其执行指示和结果取得来实现处理功能的、所谓ASP(Application Service Provider;应用服务提供商)式的服务,执行上述处理的结构。而且,本方式中的程序中假设包含供电子计算机进行的处理用的信息,即按照程序的信息(虽然不是对于计算机的直接指令,但是具有规定计算机的处理的性质的数据等)。

[0298] 而且,在该方式中,通过在计算机上执行规定的程序而构成硬件实体,但也可以将这些处理内容的至少一部分由硬件实现。

[0299] 上述的本发明的实施方式的记载是以例证和记载为目的进行提示的。没有穷举的意思,也没有将发明限定于公开的严密形式的意思。变形和变化可以根据上述的教导而进行。实施方式,是为了提供本发明的原理的最佳的例证,并且为了该领域的技术人员对本发明以各种实施方式且附加各种变形进行利用以适合深思熟虑的实际的使用,而选择并表现

的方式。所有这样的变形或变化都在由附加的权利要求书所确定的本发明的范围内,所述附加权利要求书是按照公正合法公平给出的范围被解释的。



输入x	二进制表现	代码 (r = 0)	代码 (r = 1)	代码 (r = 2)
0	0000	1	1 0	1 00
1	0001	01	1 1	1 01
2	0010	001	01 0	1 10
3	0011	0001	01 1	1 11
4	0100	00001	001 0	01 00
5	0101	000001	001 1	01 01
6	0110	0000001	0001 0	01 10
7	0111	00000001	0001 1	01 11
8	1000	000000001	00001 0	001 00
9	1001	0000000001	00001 1	001 01
:	:	:	:	:

图1

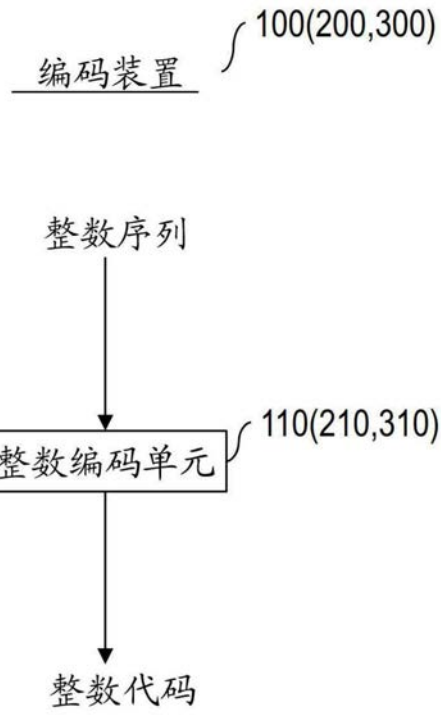


图2

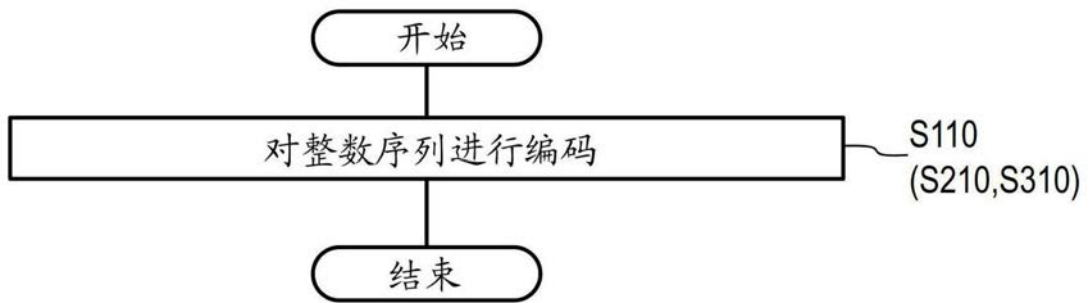


图3

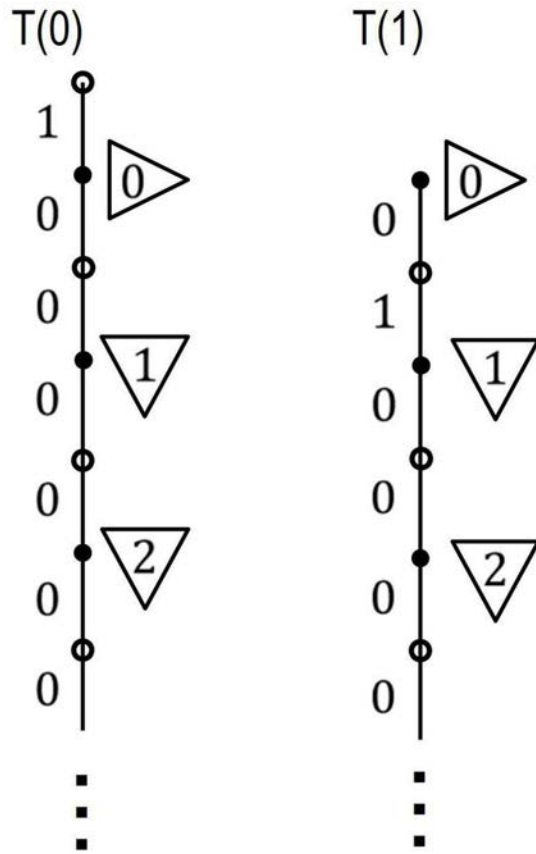


图4



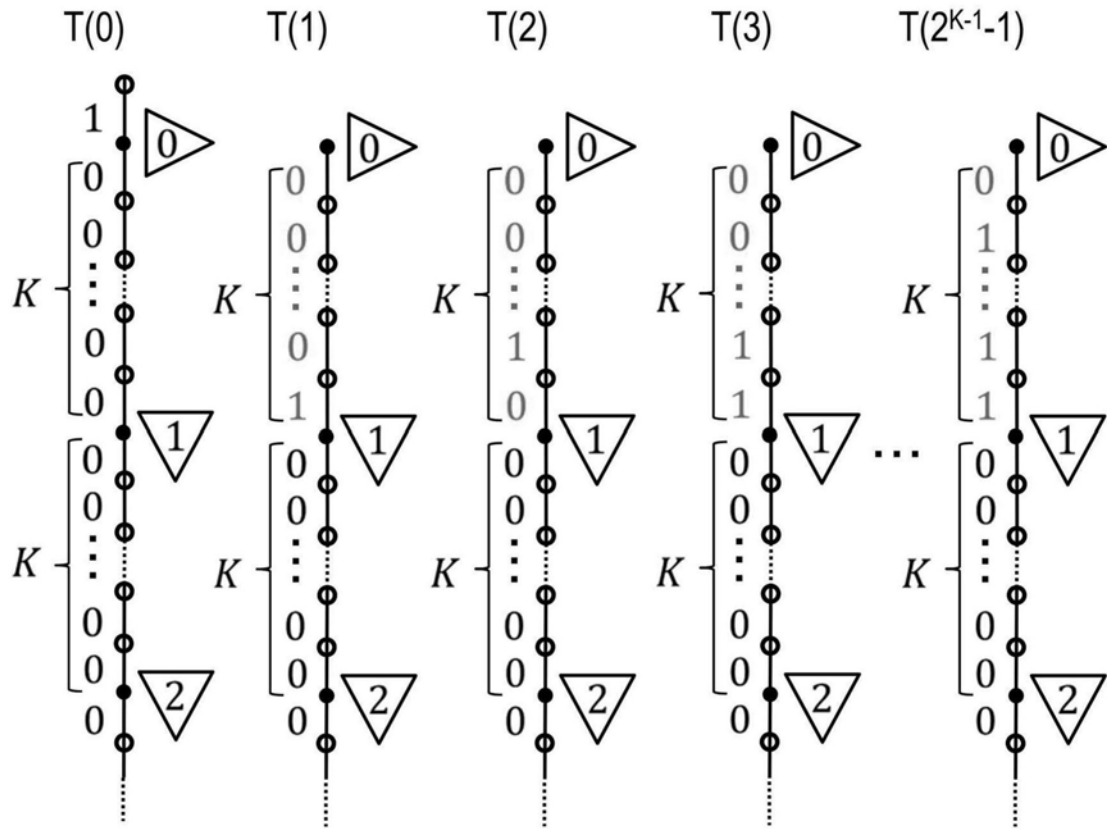


图6

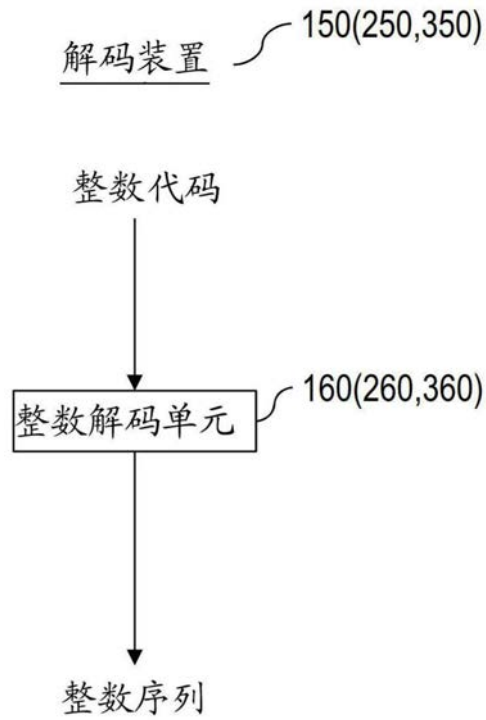


图7

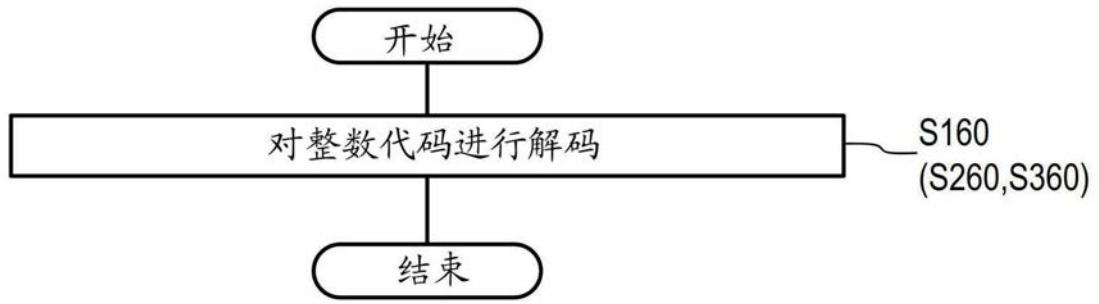


图8

整数值的 部分序列	代码
1	01
2	01 00
3	01 00 00
4	01 00 00 00
:	:

图9A

整数值的 部分序列	代码
00	1
01	1 00
02	1 00 00
03	1 00 00 00
04	1 00 00 00 00
:	:

图9B

整数值的 部分序列	代码
1	001
2	001 000
3	001 000 000
4	001 000 000 000
:	:

图10A

整数值的 部分序列	代码
01	010
02	010 000
03	010 000 000
04	010 000 000 000
:	:

图10B

整数值的 部分序列	代码
001	011
002	011 000
003	011 000 000
004	011 000 000 000
:	:

图10C



整数值的部分序列	代码
0000	1
0001	1 000
0002	1 000 000
0003	1 000 000 000
0004	1 000 000 000 000
:	:

图10D

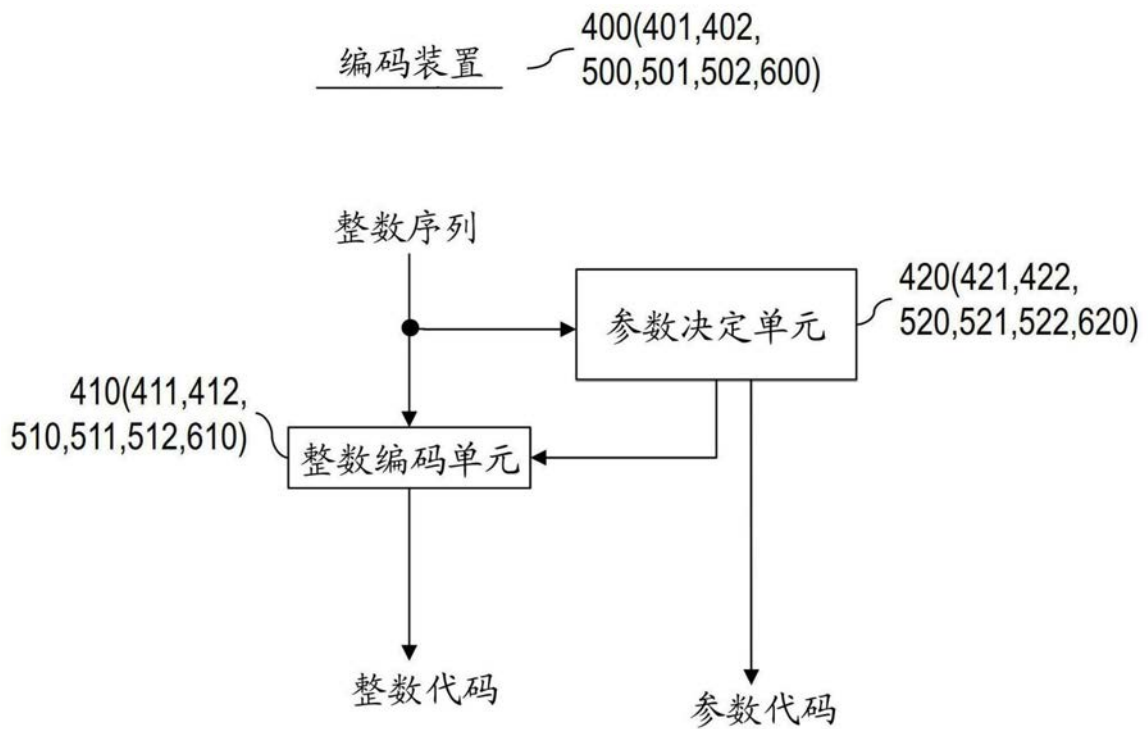


图11

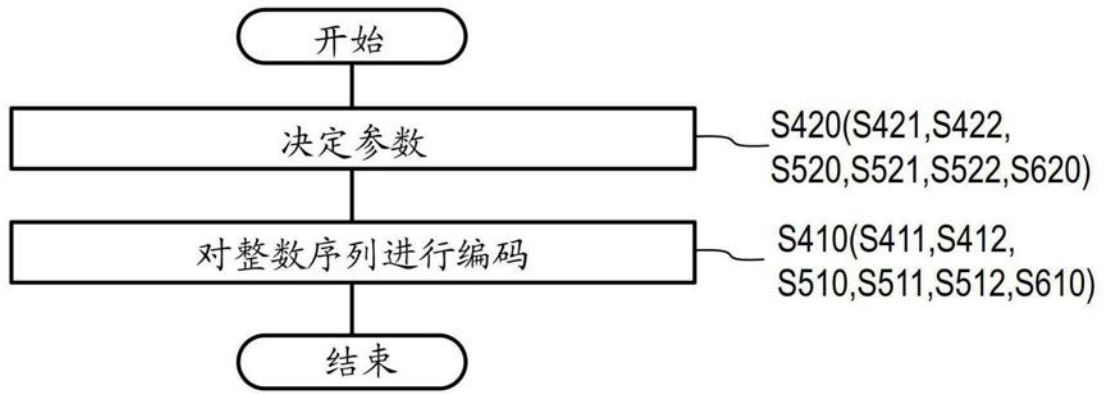


图12

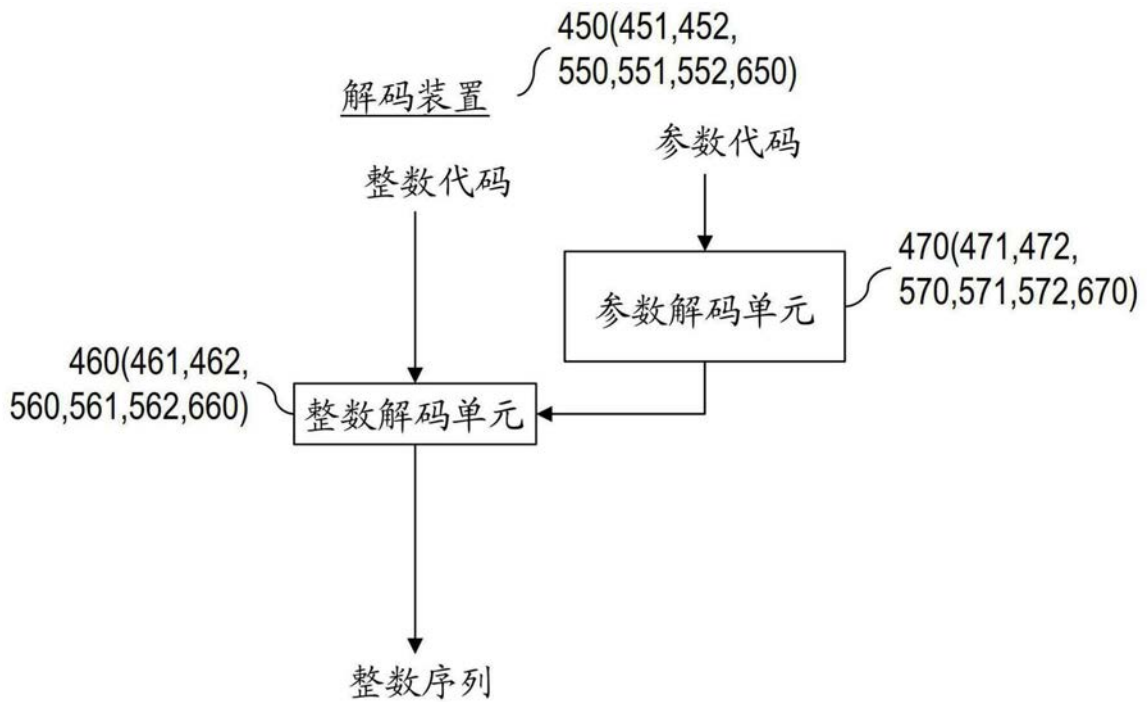


图13

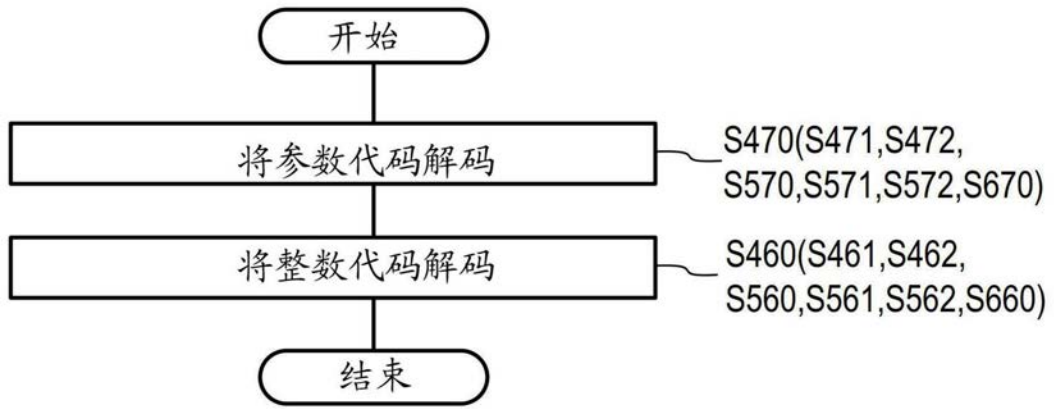


图14