

(19) 日本国特許庁(JP)

(12) 特許公報(B2)

(11) 特許番号

特許第6885466号
(P6885466)

(45) 発行日 令和3年6月16日(2021.6.16)

(24) 登録日 令和3年5月17日(2021.5.17)

(51) Int. Cl.		F I	
HO3M 7/40 (2006.01)		HO3M 7/40	
G1OL 19/00 (2013.01)		G1OL 19/00	250
G1OL 19/002 (2013.01)		G1OL 19/002	

請求項の数 15 (全 36 頁)

(21) 出願番号	特願2019-532376 (P2019-532376)	(73) 特許権者	000004226 日本電信電話株式会社 東京都千代田区大手町一丁目5番1号
(86) (22) 出願日	平成30年4月20日(2018.4.20)	(74) 代理人	100121706 弁理士 中尾 直樹
(86) 国際出願番号	PCT/JP2018/016227	(74) 代理人	100128705 弁理士 中村 幸雄
(87) 国際公開番号	W02019/021552	(74) 代理人	100147773 弁理士 義村 宗洋
(87) 国際公開日	平成31年1月31日(2019.1.31)	(72) 発明者	杉浦 亮介 東京都千代田区大手町一丁目5番1号 日本電信電話株式会社内
審査請求日	令和1年11月18日(2019.11.18)	(72) 発明者	鎌本 優 東京都千代田区大手町一丁目5番1号 日本電信電話株式会社内
(31) 優先権主張番号	特願2017-143403 (P2017-143403)		
(32) 優先日	平成29年7月25日(2017.7.25)		
(33) 優先権主張国・地域又は機関	日本国(JP)		

最終頁に続く

(54) 【発明の名称】 符号化装置、復号装置、符号化方法、復号方法、符号化プログラム、復号プログラム

(57) 【特許請求の範囲】

【請求項1】

非負の整数値 x 、 n 、 $n \in \{1, 2, \dots, N\}$ による系列（以下、「整数系列」という）を入力として、前記整数系列に含まれる各整数値を予め定めた順に符号化して符号列を得て出力する符号化装置であって、

K を 2 以上の整数として、

前記予め定めた順の整数値による部分系列が、整数値 0 の 2 ^{$K-1$} 個のうちの予め定めた何れかの個数である L 個の連続である場合には、該 L 個の連続に対応する符号として 1 ビットの符号（以下、「符号 A」という）を得、

前記予め定めた順の整数値による部分系列が、0 個から $L-1$ 個連続する整数値 0 と、1 個の 0 以外の整数値 x 、 n と、による組、である場合には、該組に対応する符号として $K \times x$ ビットまたは $K \times x + 1$ ビットの符号（以下、「符号 B」という）を得る処理を、

前記予め定めた順の N 個目の整数値が処理されるまで行い、得た符号を連結したものを前記符号列として出力する整数符号化部

を含む符号化装置。

【請求項2】

請求項1に記載の符号化装置であって、

ビット値 "x" はビット値 "1" でありビット値 "y" はビット値 "0" である、または、ビット値 "x" はビット値 "0" でありビット値 "y" はビット値 "1" である、として、

前記符号 A は、ビット値が "x" である 1 ビットの符号であり、

前記符号 B は、1ビット目から K ビット目までに少なくとも 1 つのビット値 " x " と少なくとも 1 つのビット値 " y " を含み、かつ、最後から $K \times (x \text{ n } 1)$ ビットはビット値が " y " である

符号化装置。

【請求項 3】

請求項 2 に記載の符号化装置であって、

L は $2^{K'}$ であり、

整数系列における整数値 0 の 0 個から $2^{K'}$ 個の $2^{K'}$ 通りの連続個数のそれぞれは、

符号 B が $K \times x \text{ n } + 1$ ビットであるときの、

該符号 B の 1 ビット目から $K + 1$ ビット目までの $K + 1$ ビットで構成され、1 ビット目のビット値が " x " であり、2 ビット目から $K + 1$ ビット目までのビット値が " y " である、1 通りの部分符号と、

符号 B が $K \times x \text{ n}$ ビットであるときの、

該符号 B の 1 ビット目から K ビット目までの K ビットで構成され、1 ビット目のビット値が " y " であり、2 ビット目から K ビット目までの K 1 ビット中の少なくとも 1 ビットのビット値が " x " である、 $2^{K'}$ 通りの部分符号と、

の $2^{K'}$ 通りの部分符号のうちの何れかと一対一に対応する

符号化装置。

【請求項 4】

非負の整数値による系列を入力として、N サンプルによる所定単位である非負の整数値 x_n 、 $n \in \{1, 2, \dots, N\}$ による系列（以下、「整数系列」という）ごとに符号化する符号化装置であって、

前記符号化装置は、前記整数系列に含まれる各整数値を予め定めた順に符号化して符号列を得て出力するものであり、

前記整数系列ごとに、

前記予め定めた順の整数値による部分系列が、整数値 0 の 2 個から $2^{K'}$ 個のうちの予め定めた何れかの個数である L 個の連続である場合には、該 L 個の連続に対応する符号として L ビットの符号（以下、「符号 A」という）を得、

前記予め定めた順の整数値による部分系列が、0 個から L 1 個連続する整数値 0 と、1 個の 0 以外の整数値 x_n と、による組、である場合には、該組に対応する符号として $K \times x \text{ n}$ ビットまたは $K \times x \text{ n } + 1$ ビットの符号（以下、「符号 B」という）を得る処理を、

前記予め定めた順の N 個目の整数値が処理されるまで行い、得た符号を連結したものを前記符号列として出力する整数符号化部を含み、

前記整数系列ごとに、

前記整数系列に含まれる整数値に基づいて次式によりライスパラメータ r を求める処理と、

【数 7】

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

前記整数系列に含まれる整数値に基づいて次式により符号化パラメータ K' を求める処理と、の少なくとも何れかと、

【数 8】

$$K' = \log_2 \left(1 + \frac{1}{\frac{1}{N} \sum_{n=1}^N x_n} \right)$$

ライスパラメータ r と符号化パラメータ K' の少なくとも何れかを符号化してパラメータ符号を得る処理と、

を行うパラメータ決定部を更に含み、

前記整数符号化部は、前記整数系列ごとに、前記パラメータ決定部が得た符号化パラメータ K' 、または、前記パラメータ決定部が得た前記ライスパラメータ r から $K' = 2^r$ により得た符号化パラメータ K' 、を丸めて整数値にした値を上記 K として用いる符号化装置。

【請求項 5】

請求項 4 に記載の符号化装置であって、

前記整数符号化部は、前記整数系列に対応する前記符号化パラメータ K' が 1 以下である場合、または、前記整数系列に対応する前記ライスパラメータ r が非負である場合、には、前記整数系列をゴロムライス符号化して符号列を得る符号化装置。

10

【請求項 6】

符号列を入力として、前記符号列に含まれる符号を順に復号して整数値 x_n 、 $n \in \{1, 2, \dots, N\}$ による系列を得て出力する復号装置であって、

K を 2 以上の整数として、

前記符号列に含まれる符号が 1 ビットの符号（以下「符号 A」という）である場合には、該符号 A に対応する整数値 0 の 2 個から 2^{K-1} 個のうちの予め定めた何れかの個数である L 個の連続を得、

前記符号列に含まれる符号が $K \times x_n$ ビットまたは $K \times x_n + 1$ ビットの符号（以下、「符号 B」という）である場合には、該符号 B に対応する、0 個から $L - 1$ 個連続する整数値 0 と、1 個の 0 以外の整数値 x_n と、による組を得る処理を、

20

前記符号列に含まれる符号がなくなるまで行い、得た整数値による系列を前記整数値 x_n による系列として出力する整数復号部

を含む復号装置。

【請求項 7】

請求項 6 に記載の復号装置であって、

ビット値 "x" はビット値 "1" でありビット値 "y" はビット値 "0" である、または、ビット値 "x" はビット値 "0" でありビット値 "y" はビット値 "1" である、として、

前記符号 A は、ビット値が "x" である 1 ビットの符号であり、

前記符号 B は、1 ビット目から K ビット目までに少なくとも 1 つのビット値 "x" と少なくとも 1 つのビット値 "y" を含み、かつ、最後から $K \times (x_n - 1)$ ビットはビット値が "y" である

30

復号装置。

【請求項 8】

請求項 6 または 7 に記載の復号装置であって、

前記整数復号部は、前記入力された符号列の最初から順に、

前記入力された符号列を、連続する K ビット（以下、「 K ビット列」という）ずつ読みこむことと、

前記読み込まれた K ビット列が "y" を含むものであるとき、該 "y" を含む K ビット列の直前のビットが、前記符号 A または前記符号 B の終端のビットであると識別することと、

40

を行うことで、

前記入力された符号列に含まれる前記符号 A と前記符号 B を特定する

復号装置。

【請求項 9】

請求項 7 または 8 に記載の復号装置であって、

L は 2^{K-1} であり、

符号 B が $K \times x_n + 1$ ビットであるときの、

該符号 B の 1 ビット目から $K+1$ ビット目までの $K+1$ ビットで構成され、1 ビット目のビット値が "x" であり、2 ビット目から $K+1$ ビット目までのビット値が "y" である、1 通りの部分符号と、

50

符号 B が $K \times x \ n$ ビットであるときの、
 該符号 B の 1 ビット目から K ビット目までの K ビットで構成され、1 ビット目のビット値が " y " であり、2 ビット目から K ビット目までの $K-1$ ビット中の少なくとも 1 ビットのビット値が " x " である、 2^{K-1} 通りの部分符号と、
 の 2^{K-1} 通りの部分符号のそれぞれは、
 前記整数復号部により得られた整数値による系列における整数値 0 の 0 個から 2^{K-1} 個の 2^{K-1} 通りの連続個数のうちの何れかと一対一に対応する
 復号装置。

【請求項 1 0】

所定単位ごとのパラメータ符号と符号列を入力として、前記符号列に含まれる符号を順に復号して整数値 x_n 、 $n \in \{1, 2, \dots, N\}$ による系列を得て出力する復号装置であって、

所定単位ごとに、

前記符号列に含まれる符号が 1 ビットの符号（以下「符号 A」という）である場合には、該符号 A に対応する整数値 0 の 2 個から 2^{L-1} 個のうちの予め定めた何れかの個数である L 個の連続を得、

前記符号列に含まれる符号が $K \times x \ n$ ビットまたは $K \times x \ n+1$ ビットの符号（以下、「符号 B」という）である場合には、該符号 B に対応する、0 個から $L-1$ 個連続する整数値 0 と、1 個の 0 以外の整数値 x_n と、による組を得る処理を、

前記符号列に含まれる符号がなくなるまで行い、得た整数値による系列を前記整数値 x_n による系列として出力する整数復号部を含み、

所定単位ごとに、前記パラメータ符号を復号してライスパラメータ r と符号化パラメータ K' の少なくとも何れかを得るパラメータ復号部を更に含み、

前記整数復号部は、所定単位ごとに、前記パラメータ復号部が得た符号化パラメータ K' 、または、前記パラメータ復号部が得た前記ライスパラメータ r から $K' = 2^r$ により得た符号化パラメータ K' 、を丸めて整数値にした値を上記 K として用いる

復号装置。

【請求項 1 1】

請求項 1 0 に記載の復号装置であって、

前記整数復号部は、前記符号化パラメータ K' が 1 以下である場合、または、前記ライスパラメータ r が非負である場合、には、前記符号列をゴロムライス復号して整数値による系列を得る

復号装置。

【請求項 1 2】

符号化装置が、非負の整数値 x_n 、 $n \in \{1, 2, \dots, N\}$ による系列（以下、「整数系列」という）を入力として、前記整数系列に含まれる各整数値を予め定めた順に符号化して符号列を得て出力する符号化方法であって、

K を 2 以上の整数として、

前記予め定めた順の整数値による部分系列が、整数値 0 の 2 個から 2^{K-1} 個のうちの予め定めた何れかの個数である L 個の連続である場合には、該 L 個の連続に対応する符号として 1 ビットの符号（以下、「符号 A」という）を得、

前記予め定めた順の整数値による部分系列が、0 個から $L-1$ 個連続する整数値 0 と、1 個の 0 以外の整数値 x_n と、による組、である場合には、該組に対応する符号として $K \times x \ n$ ビットまたは $K \times x \ n+1$ ビットの符号（以下、「符号 B」という）を得る処理を、

前記予め定めた順の N 個目の整数値が処理されるまで行い、得た符号を連結したものを前記符号列として出力する整数符号化ステップ

を実行する符号化方法。

【請求項 1 3】

復号装置が、符号列を入力として、前記符号列に含まれる符号を順に復号して整数値 x_n 、 $n \in \{1, 2, \dots, N\}$ による系列を得て出力する復号方法であって、

K を 2 以上の整数として、

10

20

30

40

50

前記符号列に含まれる符号が1ビットの符号（以下「符号A」という）である場合には、該符号Aに対応する整数値0の2個から 2^{K1} 個のうちの予め定めた何れかの個数であるL個の連続を得、

前記符号列に含まれる符号が $K \times x_n$ ビットまたは $K \times x_{n+1}$ ビットの符号（以下、「符号B」という）である場合には、該符号Bに対応する、0個からL-1個連続する整数値0と、1個の0以外の整数値 x_n と、による組を得る処理を、

前記符号列に含まれる符号がなくなるまで行い、得た整数値による系列を前記整数値 x_n による系列として出力する整数復号ステップ

を実行する復号方法。

【請求項14】

請求項1ないし5の何れか1項に記載の符号化装置としてコンピュータを機能させるための符号化プログラム。

【請求項15】

請求項6ないし11の何れか1項に記載の復号装置としてコンピュータを機能させるための復号プログラム。

【発明の詳細な説明】

【技術分野】

【0001】

本発明は、音声や音響の時系列デジタル信号のサンプル系列などの整数値から成るサンプル系列を符号化、復号する技術に関する。

【背景技術】

【0002】

圧縮を目的としてサンプル系列を符号化する技術として、サンプル値を量子化することにより得た有限精度の値（以下、これを整数値と呼ぶ）を可逆符号化することにより、サンプル系列の記述に用いるビット長を削減する技術がある。この技術においては、どの整数値に対してどの長さの符号を割り当てるかが、圧縮の性能に直結する。この事実は、画像信号のサンプル列を符号化復号する画像符号化や音響信号のサンプル列を符号化復号する音響符号化などの、サンプル系列の符号化や復号の工学的応用先においても例外ではない。

【0003】

一般的に可変長の可逆符号化においては、復号可能性の制約により、整数値に割り当てる符号の構成に制約がある。具体的には、ある整数値に対して短い符号を割り当てるとすると、復号可能な符号にするには他の整数値に対して長い符号を割り当てなければならない、という制約である。従って、圧縮性能を高くするためには、符号の構成（各整数値への符号の割り当て）は整数値の系列中の値の分布に適したものとする必要がある。具体的には、出現確率が高い整数値に対して短い符号を割り当て、出現確率が低い整数値に対しては長い符号を割り当てることで、整数値の系列の圧縮後のビット長の期待値を小さくすることができる。

【0004】

上記のような可逆符号化において最も単純な可変長符号の一つとして用いられてきたのがGolomb Rice符号（ゴロムライス符号）である。Golomb Rice符号は、整数値の系列がラプラス分布に属する場合、つまり整数値の出現確率が値の大きさに対して指数的に低くなっている場合において、最小な期待ビット長（最小のビット長）を達成することが知られている。

【0005】

このGolomb Rice符号は非常に単純な構成であることから広く用いられている。Golomb Rice符号は、例えば、音響信号をAD変換してPCMにした整数値の系列やその系列をフーリエ変換して周波数スペクトルに変換した系列などに含まれる各サンプル値を量子化値で除算して得た整数値の絶対値の系列を符号化するとき用いられる。或いは、Golomb Rice符号は、上記整数値の系列の各サンプル値 x を以下の式(1)により求まる x' にそれぞれ置

10

20

30

40

50

き換えた非負整数値からなる系列を符号化するとき用いられる。

【 0 0 0 6 】

【数 1】

$$x' = \begin{cases} 2|x| - 1 & (\text{if } x > 0) \\ 2|x| & (\text{otherwise}) \end{cases} \dots(1)$$

【 0 0 0 7 】

従って、以下では非負整数値を符号化・復号する場合について説明するが、式(1)などにより整数値全体を非負整数値に一对一に変換できるという上記の事実から、符号化処理の前段で式(1)などにより整数値全体を非負整数値に変換したり、復号処理の後段で式(1)の逆変換などにより非負整数値を整数値全体に変換したりすることで、整数値全体の符号化・復号にも適応可能である。

【 0 0 0 8 】

Golomb Rice符号化は、Riceパラメータ(ライスパラメータ) r の値を適切に選択することにより、図1の表のように、非負整数値の入力 x に対する符号のビット長 $B(x)$ を

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

とし、以下の片側ラプラス分布に従う非負整数値の系列に対して近似的に最適な期待ビット長を与えるように符号化できるものである。

【 0 0 0 9 】

【数 3】

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

【 0 0 1 0 】

ただし、Riceパラメータ r は非負の整数値である。つまり、Riceパラメータ r を小さな非負の整数値とすれば、値の偏りの大きい系列を少ない平均ビット数で符号化でき、 r を大きな非負の整数値とすれば、値の偏りの小さい系列を少ない平均ビット数で符号化できる。

【 0 0 1 1 】

しかしGolomb Rice符号化はRiceパラメータ r を負値にすることができないことから、例えば分布 $p(x, 0.1)$ に従う値の系列など、系列の値の偏りが分布 $p(x, 1)$ よりも大きくなるにつれ、Riceパラメータ r をどのように設定しても圧縮の効率は低下していく。

【 0 0 1 2 】

例えば、音響信号をAD変換してPCMにするときには、大音量まで記録できるように表現可能な数値の範囲を設定しているため、多くの信号の系列に含まれる値は表現可能な数値の範囲の絶対値が小さな値のところ偏る傾向にある。特に、音声信号は発話の無い時間区間があるため、零値や背景雑音だけの非常に小さな値に偏る。また、このような信号の系列に含まれる各値を量子化値で除算して得た整数値の系列は、量子化により、元の系列よりも更に零値を含む小さな値に偏っている。

【 0 0 1 3 】

零値に偏った整数値の系列を符号化する先行技術としては非特許文献1の技術がある。非特許文献1の技術は、入力された整数値の系列に含まれる零値は、零値が連続する個数を予め決めた固定ビット数で符号化して、入力された整数値の系列に含まれる零値以外の値は、その値から1を減算して得た値(値1)をGolomb Rice符号化するものである。

【 0 0 1 4 】

また、零値を含む小さな値に偏った整数値の系列を符号化する先行技術としては非特許文献2の技術がある。非特許文献2の技術は、2つの整数値による組と1つの整数値を全単射なように予め紐付けておき、整数値の系列に含まれる2つの整数値による組を1つの整数値に変換し、変換後の整数値の系列をGolomb Rice符号化するものである。

【 先行技術文献 】

【 非特許文献 】

【 0 0 1 5 】

【 非特許文献 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.

【 非特許文献 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.

【 発明の概要 】

【 発明が解決しようとする課題 】

【 0 0 1 6 】

非特許文献1の技術によれば、零値については少ない平均ビット数で符号化することができる。しかし、非特許文献1の技術では零値以外についてはGolomb Rice符号化とほぼ同じ符号化を行うので、非特許文献1の技術には、零値ではない小さな値が分布 $p(x,1)$ よりも大きく偏っている場合には、圧縮の性能が低下するという課題がある。

【 0 0 1 7 】

また、非特許文献1の技術は数学的に表現される分布に対応した符号化方法ではない。従って、非特許文献1の技術には、整数値の系列における値の分布を知ったところで、零値の連続する個数を表現するために用いるビット数としてどれを選択して符号化すれば少ない平均ビット数で符号化できるのかがわからないという課題もある。

【 0 0 1 8 】

一方、非特許文献2の技術によれば、分布 $p(x,1)$ よりも値の偏っている整数値の系列を少ない平均ビット数で符号化することができる。しかし、非特許文献2の技術は、1つの整数値に対する平均ビット数を0.5ビットより小さくできず、理論上最適な平均ビット数が0.5ビット未満となるような偏りの大きい整数値の系列に対しては、圧縮の性能が低下するという課題がある。

【 0 0 1 9 】

そこで本発明では、零値ではない小さな値も含み、その分布が小さな値に大きく偏っている整数値の系列であっても、少ない平均ビット数で符号化することができる符号化復号技術を提供することを目的とする。

【 課題を解決するための手段 】

【 0 0 2 0 】

本発明の一態様は、入力された非負の整数値 x_n 、 $n = \{1, 2, \dots, N\}$ による系列（以下、「整数系列」という）について、 K を2以上の整数として、前記整数系列に含まれる整数値0の2個から 2^{K-1} 個のうちの予め定めた何れかの個数である L 個の連続に対応する符号として1ビットの符号（以下、「符号A」という）を得、前記整数系列に含まれる、0個から $L-1$ 個連続する整数値0と、1個の0以外の整数値 x_n と、による組、に対応する符号として $K \times x_n$ ビットまたは $K \times x_{n+1}$ ビットの符号（以下、「符号B」という）を得る整数符号化部を含む。

【 0 0 2 1 】

本発明の一態様は、入力された符号列について、 K を2以上の整数として、前記入力された符号列に含まれる1ビットの符号（以下「符号A」という）から、整数値0の2個から 2^{K-1} 個のうちの予め定めた何れかの個数である L 個の連続を得、前記入力された符号列に含まれる $K \times x_n$ ビットまたは $K \times x_{n+1}$ ビットの符号（以下、「符号B」という）から、0個から $L-1$ 個連続する整数値0と、1個の0以外の整数値 x_n と、による組を得ることにより整数

値 x_n 、 $n \in \{1, 2, \dots, N\}$ による系列を得る整数復号部を含む。

【0022】

本発明の一態様は、非負値である整数値 x_n 、 $n \in \{1, 2, \dots, N\}$ の系列（以下、「整数系列」という）を表す符号列のデータ構造であって、 K を2以上の整数として、前記符号列は、前記整数系列から前記符号列を得る符号化装置、または \wedge および、前記符号列から前記整数系列を得る復号装置、で用いられるものであり、前記整数系列に含まれる整数値0の2個から 2^{K-1} 個のうちの予め定めた何れかの個数である L 個の連続に対応する符号である1ビットの符号（以下、「符号A」という）と、前記整数系列に含まれる、0個から $L-1$ 個連続する整数値0と、1個の0以外の整数値 x_n と、による組、に対応する符号である $K \times x_n$ ビットまたは $K \times x_n + 1$ ビットの符号（以下、「符号B」という）と、を含む。

10

【発明の効果】

【0023】

本発明によれば、零値ではない小さな値も含んで小さな値に大きく偏っている整数値の系列や、非特許文献2の技術では対応しきれないくらいに値の偏った整数値の系列であっても、少ない平均ビット数で符号化することができる。また、そのような符号を復号することができる。

【図面の簡単な説明】

【0024】

【図1】非負整数値の入力 x に対するGolomb Rice符号を示す図である。

【図2】符号化装置100（200, 300）の構成の一例を示すブロック図である。

20

【図3】符号化装置100（200, 300）の動作の一例を示すフローチャートである。

【図4】 $K=2$ のときの符号木の一例を示す図である。

【図5】 $K=3$ のときの符号木の一例を示す図である。

【図6】 K を一般化したときの符号木の一例を示す図である。

【図7】復号装置150（250, 350）の構成の一例を示すブロック図である。

【図8】復号装置150（250, 350）の動作の一例を示すフローチャートである。

【図9A】 $K=2$ のときの対応表の一例を示す図である。

【図9B】 $K=2$ のときの対応表の一例を示す図である。

【図10A】 $K=3$ のときの対応表の一例を示す図である。

30

【図10B】 $K=3$ のときの対応表の一例を示す図である。

【図10C】 $K=3$ のときの対応表の一例を示す図である。

【図10D】 $K=3$ のときの対応表の一例を示す図である。

【図11】符号化装置400（401, 402, 500, 501, 502, 600）の構成の一例を示すブロック図である。

【図12】符号化装置400（401, 402, 500, 501, 502, 600）の動作の一例を示すフローチャートである。

【図13】復号装置450（451, 452, 550, 551, 552, 650）の構成の一例を示すブロック図である。

【図14】復号装置450（451, 452, 550, 551, 552, 650）の動作の一例を示すフローチャートである。

40

【発明を実施するための形態】

【0025】

以下、本発明の実施の形態について、詳細に説明する。なお、同じ機能を有する構成部には同じ番号を付し、重複説明を省略する。

【0026】

<第一実施形態>

符号化装置

図2及び図3を参照して、第一実施形態の符号化装置100が実行する符号化方法の処理手続きを説明する。第一実施形態の符号化装置100は、図2に示す通り、整数符号化

50

部 1 1 0 を含む。第一実施形態の符号化装置 1 0 0 が図 3 に示す各ステップの処理を実行することにより、第一実施形態の符号化方法が実現される。

【 0 0 2 7 】

第一実施形態の符号化装置 1 0 0 には、非負の整数値の系列が入力される。この非負の整数値の系列としては、例えば、マイクロホンで収音した音声や音楽などをデジタル信号に変換して得た信号やカメラで撮像した画像や映像をデジタル信号に変換して得た信号などの信号の一部あるいは全部を既存の技術により量子化して有限精度の値にして得た整数値の絶対値による系列、または、整数値を x として式(1)により得た x' による系列を入力してもよい。

【 0 0 2 8 】

第一実施形態の符号化装置 1 0 0 は、予め定めた複数の符号木を予め定めた規則に則って用い、非負の整数値の系列を可変長符号化することで、ラプラス分布よりも偏りの大きな分布の非負の整数値の系列に対して Golomb Rice 符号化よりも短いビット長となる符号化処理を実現するものである。ここで符号木とは、入力された非負の整数値に対してどの符号を割り当てるかの予め定めた規則を表すグラフのことを指す。以下では、符号化装置 1 0 0 に入力された非負の整数値を単に「整数値」と呼んで説明する。

【 0 0 2 9 】

[整数符号化部 1 1 0]

整数符号化部 1 1 0 には、符号化装置 1 0 0 に入力された整数値の系列のうちの、 N サンプル (N は自然数) ずつの整数値による系列が入力される。入力された整数値による系列を整数系列 x_1, x_2, \dots, x_N とする。整数符号化部 1 1 0 は、予め定めたまたは図示しない手段により入力された 2 以上の自然数である符号化パラメータ K に基づき、整数系列 x_1, x_2, \dots, x_N を以下の符号木を用いた符号化処理で符号化して符号を得、得た符号を整数符号として出力する ($S 1 1 0$)。以下では、整数系列 x_1, x_2, \dots, x_N の各整数が x_n (n は系列中のサンプル番号であり、 $n \in \{1, 2, \dots, N\}$) であるとして説明を行う。

【 0 0 3 0 】

[[符号化処理に用いる符号木]]

まず、整数符号化部 1 1 0 における符号化処理に用いる符号木を例示する。

【 0 0 3 1 】

整数符号化部 1 1 0 には、図 4 から図 6 に例示するような 2^{K-1} 個 (2 の $K-1$ 乗個) の符号木が予め記憶されている。図 4 は $K=2$ のときの符号木の例であり、 $T(0)$ と $T(1)$ のそれぞれが符号木である。図 5 は $K=3$ のときの符号木の例であり、 $T(0)$ から $T(3)$ のそれぞれが符号木である。図 6 は K を一般化したときの符号木の例であり、 $T(0)$ から $T(2^{K-1}-1)$ のそれぞれが符号木である。以下では、 T に付された括弧内の番号を符号木番号と呼ぶ。

【 0 0 3 2 】

[[符号木における入力された整数値と当該整数値に対応する符号との関係]]

次に、各符号木における入力された整数値と当該整数値に対応する符号との関係を説明する。

【 0 0 3 3 】

各符号木のノードの右側にある三角で囲まれている整数値は入力された整数値に対応している。また、各符号木のエッジの左側にあるビット値 '0' 及び '1' は入力された整数値に割り当てる符号の規則に対応している。この規則は、符号木の上端のノードから入力された整数値に対応するノードまでのエッジの左側にあるビット値を並べたものを入力された整数値に割り当てる符号として得る規則である。

【 0 0 3 4 】

2^{K-1} 個の符号木のうちの 1 つの符号木は、図 4 から図 6 の $T(0)$ に対応するものであり、入力された整数値 x_n に対応する符号として、ビット数が $K \times x_n + 1$ ビットであり、最初の 1 ビット目のビット値が '1' であり、2 ビット目から $K \times x_n + 1$ ビット目までのビット値が '0' である符号を得る規則に対応する符号木である。なお、この符号木によれば、入力さ

10

20

30

40

50

れた整数値 x_n が 0 である場合には、符号のビット数 $K \times x_{n+1}$ は 1 ビットであるため、ビット値が ' 1 ' である 1 ビットの符号を得ることになる。

【 0 0 3 5 】

2^{K+1} 個の符号木のうちの上記以外の $2^{K+1} - 1$ 個の符号木は、図 4 から図 6 の $T(1)$ から $T(2^{K+1} - 1)$ までに対応するものであり、入力された整数値 x_n に対応する符号として、ビット数が $K \times x_n$ ビットであり、最初の 1 ビット目のビット値が ' 0 ' であり、2 ビット目から K ビット目までの $K - 1$ ビット中の少なくとも 1 ビットのビット値が ' 1 ' であり、 $K + 1$ ビット目から $K \times x_n$ ビット目までの $K \times (x_n - 1)$ ビットのビット値が ' 0 ' である、すなわち、最後から $K \times (x_n - 1)$ ビットのビット値が ' 0 ' である符号を得る規則に対応する符号木である。ここで、2 ビット目から K ビット目までの $K - 1$ ビットは $2^{K+1} - 1$ 個の符号木のそれぞれに一対一に対応するビット群である。なお、これら符号木によれば、入力された整数値 x_n が 0 である場合には、符号のビット数 $K \times x_n$ は 0 ビットであるため、符号を得ないことになる。

【 0 0 3 6 】

例えば、図 4 の $K=2$ の場合における入力された整数値と当該整数値に対応する符号の規則は、入力された整数値 0 を符号木 $T(0)$ で符号化する場合には、入力された整数値 0 に対応する符号として ' 1 ' を得、入力された整数値 1 を符号木 $T(0)$ で符号化する場合には、入力された整数値 1 に対応する符号として ' 100 ' を得、入力された整数値 2 を符号木 $T(0)$ で符号化する場合には、入力された整数値 2 に対応する符号として ' 10000 ' を得、入力された整数値 0 を符号木 $T(1)$ で符号化する場合には、入力された整数値 0 に対応する符号を得ず、入力された整数値 1 を符号木 $T(1)$ で符号化する場合には、入力された整数値 1 に対応する符号として ' 01 ' を得、入力された整数値 2 を符号木 $T(1)$ で符号化する場合には、入力された整数値 2 に対応する符号として ' 0100 ' を得る、という規則である。

【 0 0 3 7 】

[[複数の符号木を用いる規則]]

次に、複数の符号木を用いる規則について説明する。整数符号化部 1 1 0 には、複数の符号木を用いる規則も予め記憶されている。

【 0 0 3 8 】

図 4 から図 6 の各符号木のノードの右側にある整数値を囲む三角の向きは、複数の符号木を用いる規則に対応している。

【 0 0 3 9 】

右向きの三角は、入力された整数値の次の整数値を符号化する際には、右隣の符号木（符号木番号が 1 つ大きい符号木）を用いる規則を表す。なお、その際、最も右にある符号木（符号木番号が最も大きい符号木）の右隣は最も左にある符号木（符号木番号が最も小さい符号木）であるとする。なお、この規則はあくまでも一例であって、複数の符号木を巡回的に用いる規則であればどのような規則であってもよい。すなわち、各符号木について、右向きの三角のときに次の整数値の符号化には異なる符号木を用いる規則であって、全ての符号木について、右向きの三角のときに次の整数値の符号化に用いる符号木が重複しない規則であれば、どのような規則であってもよい。なお、複数の符号木を巡回的に用いる規則においては、用いる符号木の最大個数は 2^{K+1} 個である。

【 0 0 4 0 】

下向き三角は、入力された整数値の次の整数値を符号化する際には、符号木 $T(1)$ を用いる規則を表す。なお、この規則もあくまでも一例であって、下向き三角のときには、各下向き三角について予め定めた何れか 1 つの符号木を次の整数値の符号化に用いる符号木とする規則であれば、どのような規則であってもよい。例えば、下向き三角のときに、全て下向き三角について予め定めた何れか 1 つの同じ符号木を次の整数値の符号化に用いる符号木とする規則としてもよい。また、平均ビット数を短くするためには、下向き三角のときに、図 4 から図 6 の $T(0)$ に対応する符号木以外の符号木を、次の整数値の符号化に用いる符号木とする規則とするのがよい。

【 0 0 4 1 】

なお、平均ビット数を最も短くするためには、右向き三角のときに、 2^{K+1} 個の全ての符

10

20

30

40

50

号木を巡回的に用い、かつ、全ての符号木について、下向き三角のときに、図4から図6のT(0)に対応する符号木において右向き三角のときに次の整数値の符号化に用いる符号木を、次の整数値の符号化に用いる符号木とする規則とするのがよい。

【0042】

例えば、図4のK=2の場合の符号木を用いた符号化の規則は、入力された整数値0を符号木T(1)で符号化する場合には、入力された整数値0に対応する符号は得ずに次の整数値を符号化する際に用いる符号木をT(0)とする規則であり、入力された整数値2を符号木T(0)で符号化する場合には、入力された整数値2に対応する符号として'10000'を得て、次の整数値を符号化する際に用いる符号木をT(1)とする規則である。

【0043】

複数の符号木のうちの何れの符号木を符号化処理の最初に用いてもよいが、何れの符号木を符号化処理の最初に用いるかの規則も、複数の符号木を用いる規則として予め決めて記憶しておく。例えば、図4のK=2の場合において、整数系列 x_1, x_2, \dots, x_N を整数値 x_1 から順に符号化する場合には、整数値 x_1 をT(0)で符号化する規則としてもよいし、整数値 x_1 をT(1)で符号化する規則としてもよいが、T(0)とT(1)の何れを整数値 x_1 の符号化に用いるかは予め決めて記憶しておく。

【0044】

[[符号木を用いた符号化処理]]

整数符号化部110は、上述した複数の符号木を用いて、複数の符号木を用いる上述した規則に従って、入力された整数系列 x_1, x_2, \dots, x_N に含まれる各整数値を予め定めた順に符号化して符号を得、得た符号を整数符号として出力する。例えば、予め定めた順が系列中のサンプル番号順であり、最初に用いる符号木がT(0)である場合に整数符号化部110が行う符号化手順は下記の通りである。

(符号化手順)

1. サンプル番号 n ,符号木番号の初期値をそれぞれ1,0とする。
2. $n=N$ となるまで以下の(i)~(iv)を繰り返す。

(i) 符号木番号が0なら符号'1'を得る。

(ii) 整数値 $x_n=0$ なら、符号木番号に1を加算したものを新たな符号木番号とする。この際、1を加算して符号木番号が 2^{K-1} となった場合には0を新たな符号木番号とする。

(iii) 整数値 $x_n \neq 0$ なら、1ビットの'0'と符号木に一対一に対応する対応するK-1ビットとによるKビットの符号を得、さらに、 $K \times (x_n - 1)$ ビットの符号'0'を得て、1を新たな符号木番号とする。

(iv) n に1を加算したものを新たな n とする。

3. 1~2により得た符号のうちの最初に得た符号'1'を取り除く。

(あるいは始めの1回の2.(i)操作を行わない。)

4. 1~3により得た符号の最後に符号'1'(終端符号)を加える。

5. 1~4により得た符号を整数符号として出力する。

【0045】

なお、符号化手順のうちの3を行うのは必須ではない。最初に用いる符号木がT(0)である場合には、 x_1 の値がどのような値であっても、1~2により得た符号のうちの最初に得た符号は'1'である。従って、最初に得た符号'1'を取り除いたものを整数符号として符号化装置100が出力しても、復号装置150が入力された整数符号の最初に符号'1'を加えれば、符号化装置100の整数符号化部110が行った符号化処理に対応する復号処理を復号装置150が行うことが可能である。このことから、符号化手順の3では最初に得た符号'1'を取り除いている。しかし、必ずしも符号化手順の3を行う必要はなく、1~2により得た符号のうちの最初に得た符号'1'を取り除かないでもよい。この場合には、対応する復号装置150が、入力された整数符号の最初に符号'1'を加える処理を行わないようにすればよく、符号化手順のうちの3は行わないでもよい。

【0046】

また、符号化装置100から復号装置150への整数符号の伝え方によっては、符号化

10

20

30

40

50

手順のうちの4は行わないでもよい。復号装置150における復号の手順は後述するが、整数符号化部110が行った符号化処理に対応する復号処理を復号装置150が行うためには、復号装置150が整数符号の終端を特定できる必要がある。符号化装置100から復号装置150への整数符号の伝える構成として、例えば、符号化装置100内または符号化装置100の後段において整数系列 x_1, x_2, \dots, x_N について得た整数符号を周知の packets 化の手法を用いて packets に格納して復号装置150に伝える構成を採用することができる。この場合には、復号側では、復号装置150内または復号装置150の前段において packets から整数符号を取り出す構成を採用することになるので、復号装置150が整数符号の終端を特定できる。従って、このような場合には、符号化手順のうちの4は行わないでもよい。符号化装置100から復号装置150への整数符号の伝える構成としては、整数系列 x_1, x_2, \dots, x_N について得た整数符号の後ろに別の符号を連結させたものを上述した packets に格納して復号装置150に伝える構成や、上述した packets 化を行わずに整数系列 x_1, x_2, \dots, x_N について得た整数符号の後ろに別の符号を連結させて復号装置150に伝える構成などを採用することもある。これらの構成において、符号化手順のうちの4を行わなかったとすると、後ろに連結させた符号が '0000' であるときなどには、N番目の整数値に対応する符号がどこまでであり、後ろに連結させた符号がどこからであるか、を復号装置150が判別できなくなってしまう。そこで、これらの構成においては、予め定めた同じNの値を符号化装置100と復号装置150に記憶しておくことなどによって、整数符号がN個の整数値に対応するものであることを復号装置150が把握可能なようにするとともに、符号化手順のうちの4も行って、整数符号の終端を復号装置150が特定できるようにする必要がある。

【0047】

なお、上述した説明の符号 '0' は2値のビット値のうちの何れか一方のビット値、上述した説明の符号 '1' は2値のビット値のうちの他方のビット値、の例である。すなわち、当然のことながら、上述した説明の符号 '0' に代えて符号 '1' を用い、符号 '1' に代えて符号 '0' を用いてもよい。

【0048】

[[符号化処理の例]]

符号化処理の例として、 $K=2$ で最初に用いる符号木が $T(0)$ である場合の、入力された整数系列 x_1, x_2, x_3, x_4, x_5 が0, 0, 2, 0, 0である場合の符号化処理を説明する。まず、始めの整数値 $x_1=0$ は $T(0)$ で符号化するので符号 '1' を得、符号木番号0に1を加算したものを新たな符号木番号とするので、次の整数値 $x_2=0$ の符号化に用いる符号木を $T(1)$ とする。次の整数値 $x_2=0$ は $T(1)$ で符号化するので符号は得ず、符号木番号1に1を加算したものを符号木番号とするが、1を加算した符号木番号が 2^k となったので0を新たな符号木番号とし、次の整数値 $x_3=2$ の符号化に用いる符号木を $T(0)$ とする。3番目の整数値 $x_3=2$ は $T(0)$ で符号化するので符号 '10000' を得、新たな符号木番号を1とする規則であるので、次の整数値 $x_4=0$ の符号化に用いる符号木を $T(1)$ とする。4番目の整数値 $x_4=0$ は $T(1)$ で符号化するので何も符号を追加せず、符号木を $T(1)$ から $T(0)$ へ遷移させる。最後の整数値0は $T(0)$ で符号化するので符号 '1' を得る。ここまでで得られた符号は '1100001' であるが、上述した符号化手順の3を行い上述した符号化手順の4を行わない場合には、最初に得た符号 '1' を省略し、最終的な整数符号 '100001' を得る。

【0049】

復号装置

図7及び図8を参照して、第一実施形態の復号装置150が実行する復号方法の処理手続きを説明する。第一実施形態の復号装置150は、図7に示す通り、整数復号部160を含む。第一実施形態の復号装置150が図8に示す各ステップの処理を実行することにより、第一実施形態の復号方法が実現される。

【0050】

第一実施形態の復号装置150には、第一実施形態の符号化装置100が出力した整数符号が入力される。第一実施形態の復号装置150は、入力された整数符号を、第一実施

形態の符号化装置 100 に対応する予め定めた複数の符号木を第一実施形態の符号化装置 100 に対応する予め定めた規則に則って用いた復号処理で復号して非負の整数値の系列を得ることで、第一実施形態の符号化装置 100 に入力された非負の整数値の系列そのものを復元するものである。以下では、上述した符号化装置 100 の説明と同様に、非負の整数値を単に「整数値」と呼んで説明する。

【0051】

[整数復号部 160]

整数復号部 160 には、復号装置 150 に入力された整数符号が入力される。整数復号部 160 は、符号化装置 100 の整数符号化部 110 が用いた符号化パラメータ K と同じ値であり、予め定めたまたは図示しない手段により入力された 2 以上の自然数である符号化パラメータ K に基づき、入力された整数符号を以下の符号木を用いた復号処理で復号して整数値の系列を得、得た整数系列 x_1, x_2, \dots, x_N を出力する (S160)。以下では、整数系列 x_1, x_2, \dots, x_N の各整数が x_n (n は系列中のサンプル番号であり、 $n \in \{1, 2, \dots, N\}$) であるとして説明を行う。

【0052】

復号処理に用いる符号木は、第一実施形態の符号化装置 100 の整数符号化部 110 が用いる符号木と同じであり、整数復号部 160 には、図 4 から図 6 に例示するような 2^{K-1} 個 (2 の $K-1$ 乗個) の符号木が予め記憶されている。復号処理における符号木における入力された符号と当該符号に対応する整数値との関係及び複数の符号木を用いる規則は、第一実施形態の符号化装置 100 の整数符号化部 110 の符号化処理における規則に対応するものであり、第一実施形態の符号化装置 100 の整数符号化部 110 の符号化処理における規則の「入力された整数値に割り当てる符号」を「入力された符号」と読み替え、「入力された整数値」を「入力された符号に割り当てる整数値」と読み替えた規則であり、整数復号部 160 に予め記憶されている。なお、復号処理に用いるパラメータ K の値は、第一実施形態の符号化装置 100 の整数符号化部 110 が用いたパラメータ K と同じ値であり、整数復号部 160 に予め記憶しておく。また、復号処理において何れの符号木を最初に用いるかの規則も、第一実施形態の符号化装置 100 の整数符号化部 110 が用いた規則と同じ規則であり、整数復号部 160 に予め記憶しておく。

【0053】

整数復号部 160 は、上述した複数の符号木を用いて、複数の符号木を用いる上述した規則に従って、入力された整数符号に対応する整数値による列を得て、得られた整数値による列の各整数を予め定めた順としたものを、整数系列 x_1, x_2, \dots, x_N として得て出力する。例えば、予め定めた順が系列中のサンプル番号順であり、最初に用いる符号木が $T(0)$ である場合に整数復号部 160 が行う復号手順は、例えば、下記の復号手順 1 や復号手順 2 の通りである。

【0054】

(復号手順 1 - 1) 符号化手順 4 による終端符号の追加を行っていない場合、すなわち、整数符号の終端を復号装置 150 が特定可能な場合

1. 符号化手順 3 で符号 '1' を取り除いた場合は、整数符号の最初に符号 '1' を追加する。

2. 符号木番号の初期値を 0 とする。

3. 読み込んでない整数符号がなくなるまで以下 (i) ~ (ii) を繰り返し、読み込んでない整数符号がなくなったら 4 に進む。

(i) 入力された整数符号を順に、符号木番号に対応する符号木に読み込んだ符号に対応する符号がある間 (すなわち、符号木番号に対応する符号木に読み込んだ符号に対応する符号がなくなる 1 つ前のビットまで)、読み込む。

(ii) 符号木番号に対応する符号木から、読み込んだ符号に対応する符号に対応する整数値を得て、当該符号木の整数値についての符号木の遷移の規則に従って新たな符号木番号を得る。

4. 3 の (ii) で得た整数値による系列を整数系列 x_1, x_2, \dots, x_N として出力する。

【 0 0 5 5 】

(復号手順 1 - 2) 符号化手順4による終端符号の追加を行った場合、すなわち、整数符号の終端を復号装置 1 5 0 が特定できない場合

1. 符号化手順3で符号 ' 1 ' を取り除いた場合は、入力された符号列 (整数符号と後続する別符号による符号列) の最初に符号 ' 1 ' を追加する。
2. サンプル番号 n , 符号木番号の初期値をそれぞれ 1, 0 とする。
3. $n=N$ となるまで以下 (i) ~ (ii) を繰り返す。
 - (i) 入力された符号列を順に、符号木番号に対応する符号木に読み込んだ符号に対応する符号がある間 (すなわち、符号木番号に対応する符号木に読み込んだ符号に対応する符号がなくなる 1 つ前のビットまで)、読み込む。
 - (ii) 符号木番号に対応する符号木から、読み込んだ符号に対応する符号に対応する整数値を得て、当該符号木の整数値についての符号木の遷移の規則に従って新たな符号木番号を得て、 n に 1 を加算したものを新たな n とする。
4. 3の(ii)で得た整数値による系列を整数系列 x_1, x_2, \dots, x_N として出力する。

10

【 0 0 5 6 】

(復号手順 2)

1. 符号化手順4で終端符号の追加を行っていない場合は、入力された符号列の最後に終端符号 ' 1 ' を追加する。また、符号化手順3で符号 ' 1 ' を取り除かなかった場合は、入力された符号列の最初に符号 ' 1 ' を取り除く。
2. サンプル番号 n , 符号木番号の初期値をそれぞれ 1, 0 とする。
3. $n = N$ となるまで以下 (i) ~ (v) を繰り返す。
 - (i) K ビット全てが符号 ' 0 ' である間、入力された符号列を順に K ビットずつ読み、その間読んだ符号 ' 0 ' の個数を K で割った数を L とする。
 - (ii) 符号木番号が 0 なら、整数値 L を得る。
 - (iii) 符号木番号が 0 でないなら、整数値 $(L+1)$ を得る。
 - (iv) 入力された符号列の次の 1 ビットが符号 ' 1 ' なら、その 1 ビットを読み、 (2^{k-1}) 個の整数値 0 を得、0 を新たな符号木番号にして、 n に 2^{k-1} を加算したものを新たな n とする。
 - (v) 入力された符号列の次の 1 ビットが符号 ' 0 ' なら、その 1 ビットを含む K ビットを読み、その K ビットの符号に対応する数を新たな符号木番号とし、(新しい符号木番号 1) 個の整数値 0 を得、 n に新しい符号木番号を加算したものを新たな n とする。
4. 得られた整数値が N 個よりも多い場合、 $N+1$ 個目の整数値以降は取り除く。

20

30

【 0 0 5 7 】

[[復号処理の例]]

復号処理の例として、 $K=2$ で最初に用いる符号木が $T(0)$ である場合の、上述した符号化処理の例で得た整数符号 ' 100001 ' を上述した復号手順 1 - 1 で復号する場合の復号処理を説明する。まず、整数符号 ' 100001 ' の先頭に符号 ' 1 ' を追加して整数符号を ' 1100001 ' とする。次に、整数符号 ' 1100001 ' の最初の 1 ビットである ' 1 ' は符号木 $T(0)$ に対応する符号があるものの、最初の 2 ビットである ' 11 ' は符号木 $T(0)$ に対応する符号がないことから、整数符号の最初の 1 ビットである ' 1 ' を読み込み、' 1 ' に対応する整数値 0 を得て、次に用いる符号木を $T(1)$ とする。次に、整数符号の残りの ' 100001 ' の最初の 1 ビットである ' 1 ' は符号木 $T(1)$ に対応する符号がないことから、最初の 1 ビットも読み込まず、0 ビットに対応する整数値 0 を得て、次に用いる符号木を $T(0)$ とする。次に、整数符号の残りの ' 100001 ' の最初の 1 ビットである ' 1 ' から最初の 5 ビットである ' 10000 ' までに対応する符号が符号木 $T(0)$ にあるものの、整数符号の残りの ' 100001 ' の最初の 6 ビットである ' 100001 ' は対応する符号が符号木 $T(0)$ がないことから、整数符号の残りの ' 100001 ' の最初の 5 ビットである ' 10000 ' を読み込み、' 10000 ' に対応する整数値 2 を得て、次に用いる符号木を $T(1)$ とする。次に、整数符号の残りの ' 1 ' は対応する符号が符号木 $T(1)$ がないことから、最初の 1 ビットも読み込まず、0 ビットに対応する整数値 0 を得て、次に用いる符号木を $T(0)$ とする。次に、整数符号の残りの ' 1 ' は対応する符号が符号木 $T(0)$ にあり、この ' 1 ' 以降には読み込んでない整数符号がないことから、' 1 ' に

40

50

対応する整数値0を得て、これらの処理により得た整数値0, 0, 2, 0, 0による系列を整数系列 x_1, x_2, x_3, x_4, x_5 として出力する。

【0058】

< 発明の原理の説明 >

ここで本発明の原理を説明する。

【0059】

第一実施形態の符号化装置100によれば、全ての非負整数値に対して符号を割り当てる事が可能であり、当該符号を第一実施形態の復号装置150が復号することにより非負整数値そのものを復元可能とするものである。図4から図6に示した通り、非負整数値 x_n に対応する符号は $K \times x_n$ ビットまたは $K \times x_n + 1$ ビットである。このことから、第一実施形態で用いた本発明の符号は、以下の偏りの大きな片側ラプラス分布に従う非負整数値の系列に対して近似的に最適な期待ビット長を与える符号であり、小さな値に大きく偏った非負整数値の系列に対する平均ビット数をGolomb Rice符号よりも少なくすることができる符号である。

【0060】

【数4】

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

【0061】

式(3)で示したGolomb Rice符号が最適な分布 $p(x, 2^r)$ と、式(4)で示した本発明の符号が最適な分布 $p(x, 1/K)$ と、の対応関係から、本発明の符号化パラメータ K の逆数の対数はRiceパラメータ r に対応していることがわかる。また、本発明の符号化パラメータ K が正の整数値であることから、符号化パラメータ K の逆数の対数は負値となるので、符号化パラメータ K は負値に拡張したRiceパラメータに対応するパラメータとして解釈することができる。つまり、非負整数値の系列に最適なRiceパラメータの推定を行うことにより、非負整数値の系列に割り当てる本発明の符号における最適なパラメータ K を推定することができるということである。

【0062】

第一実施形態の符号化装置100と復号装置150で用いた本発明の符号木における整数値と当該整数値に対応する符号との関係、及び、複数の符号木を用いる規則、は、非負整数値 x_n に対応する符号を $K \times x_n$ ビットまたは $K \times x_n + 1$ ビットとすることだけでなく、復号可能な符号とすることも考慮して設計したものである。具体的には、整数値0の2個から 2^{k-1} 個のうちの予め定めた何れかの個数の連続に対しては1ビットの符号である'1'を割り当て、整数値0の0個から予め定めた最大で 2^{k-1} 個の連続とそれに後続する1つの非負整数値 $x_n = 0$ に対しては、最初の K ビットの中に少なくとも1ビットのビット値'1'が含まれ、かつ、以降のビット値が'0'である符号を割り当てるという制約を設けている。この制約を設けていることにより、複数の整数値からなる整数系列に対応する整数符号を復号装置150で読み込む際に、 K ビット中にビット値'1'が含まれる部分に、整数符号中のある符号と1つ後の符号との境界が含まれることが分かるようにしている。なお、上述した制約を満たすため、整数値0の0個から予め定めた最大で 2^{k-1} 個の連続とそれに後続する1つの非負整数値 x_n に対する符号のビット数は $K \times x_n$ ビットではなく、 $K \times x_n + 1$ ビットとしている。

【0063】

< 第二実施形態 >

第一実施形態では、予め定めた符号木を用いて本発明の符号化と復号を実現する形態を説明したが、予め定めた符号木に代えて、予め定めた対応表を用いることでも、第一実施形態と等価な符号化処理と復号処理を実現することが可能である。この予め定めた対応表を用いる形態を第二実施形態として説明する。

【0064】

10

20

30

40

50

符号化装置

図 2 及び図 3 を参照して、第二実施形態の符号化装置 200 が実行する符号化方法の処理手続きを説明する。第二実施形態の符号化装置 200 は、図 2 に示す通り、第一実施形態の符号化装置 100 と同様、整数符号化部 210 を含む。第二実施形態の符号化装置 200 が、第一実施形態の符号化装置 100 と同様、図 3 に示す各ステップの処理を実行することにより、第二実施形態の符号化方法が実現される。

【0065】

第二実施形態の符号化装置 200 には、第一実施形態と同様に、非負の整数値の系列が入力される。第二実施形態の符号化装置 200 は、予め定められた対応表に従って、非負の整数値の系列を可変長符号化することで、ラプラス分布よりも偏りの大きな分布の非負の整数値の系列に対して Golomb Rice 符号化よりも短いビット長となる符号化処理を実現するものである。以下では、符号化装置 200 に入力された非負の整数値を単に「整数値」と呼んで説明する。

【0066】

[整数符号化部 210]

整数符号化部 210 には、符号化装置 200 に入力された整数値の系列のうちの、N サンプルずつの整数系列 x_1, x_2, \dots, x_N が入力される。第二実施形態の整数符号化部 210 は、2 以上の自然数である符号化パラメータ K に対応して予め作成して整数符号化部 210 内に記憶された対応表に基づいて、入力された整数系列 x_1, x_2, \dots, x_N を符号化して符号を得、得た符号を整数符号として出力する (S210)。

【0067】

[[符号化処理に用いる対応表]]

予め記憶された対応表は、零値の連続 (0 個以上の連続) と 1 つの非零値との組合せによる整数値の部分系列と、当該部分系列に対応する符号と、の対応関係の規則を格納したものである。 $K=2$ と $K=3$ における対応表の例を、それぞれ図 9 (A, B) と図 10 (A, B, C, D) に挙げる。対応表に格納された規則は、 2^{k-1} 個連続した零値による部分系列にはビット値が '1' である 1 ビットの符号が割り当てられ、それ以外の部分系列、すなわち、0 個から $2^{k-1}-1$ 個連続した零値とそれに後続する 1 つの非零値 x_n による部分系列には、それぞれの零値の連続と非零値 x_n の組合せに対して $K \times x_n$ ビットまたは $K \times x_{n+1}$ ビットの符号が割り当てられる規則である。

【0068】

例示した対応表では、0 個から $2^{k-1}-1$ 個連続した零値とそれに後続する 1 つの非零値 x_n による部分系列のそれぞれには、始めの K ビット内に少なくとも 1 ビットの '1' が存在し、残りの $K \times (x_n - 1)$ ビットまたは $K \times (x_n - 1) + 1$ ビットは全て '0' である符号が割り当てられる規則である。より詳しくは、この規則では、0 個から $2^{k-1}-1$ 個連続した零値と 1 つの非零値による部分系列には、1 ビット目のビット値が '1' であり、2 ビット目から K ビット目までのビット値が全て '0' であり、 $K+1$ ビット目から $K \times x_{n+1}$ ビット目までのビット値が全て '0' である $K \times x_{n+1}$ ビットの符号、または、1 ビット目のビット値が '0' であり、2 ビット目から K ビット目までの $K-1$ ビット中の少なくとも 1 ビットのビット値が '1' であり、 $K+1$ ビット目から $K \times x_n$ ビット目までのビット値が全て '0' である $K \times x_n$ ビットの符号、が割り当てられる。更に詳しくは、この規則では、 $K \times x_{n+1}$ ビットの符号のうちの 1 ビット目のビット値が '1' であり 2 ビット目から $K+1$ ビット目までが '0' である 1 ビット目から $K+1$ ビット目までの部分符号 1 通りと、 $K \times x_n$ ビットの符号のうちの 1 ビット目のビット値が '0' であり 2 ビット目から K ビット目までの $K-1$ ビット中の少なくとも 1 ビットのビット値が '1' である 1 ビット目から K ビット目までの部分符号 2^{k-1} 通りと、による 2^{k-1} 通りの部分符号と、0 個から $2^{k-1}-1$ 個の零値の連続数 2^{k-1} 通りと、が一対一に対応している。

【0069】

[[対応表を用いた符号化処理]]

整数符号化部 210 は、上述した対応表を用いて、入力された整数系列 x_1, x_2, \dots

x Nに含まれる各整数値を予め定めた順に符号化して符号を得、得た符号を整数符号として出力する。例えば、予め定めた順が系列中のサンプル番号順である場合に整数符号化部 2 1 0 が行う符号化手順は下記の通りである。

(符号化手順)

1. サンプル番号 n を 1 とする。

2. $n=N$ となるまで、以下の (i) ~ (ii) を繰り返す。

(i) x n 以降の 2^{k-1} 個の整数値の中に非零値がある場合には、 x n 以降の最初の非零値までの部分系列に対応する符号を符号表から得て、 x n 以降の最初の非零値までのサンプル数を n に加算したものを新たな n とする。

(ii) x n 以降の 2^{k-1} 個の整数値が全て零値である場合には、 x n 以降の 2^{k-1} 個の整数値の部分系列に対応する符号を符号表から得て、 2^{k-1} を n に加算したものを新たな n とする。

3. 2により得た符号の最後に符号 ' 1 ' (終端符号) を加える。

4. 3により得た符号を整数符号として出力する。

【 0 0 7 0 】

なお、第一実施形態と同様に、符号化装置 2 0 0 から復号装置 2 5 0 への整数符号の伝え方によっては、終端符号を加える処理 (符号化手順のうちの 3) は行わないでもよい。

【 0 0 7 1 】

なお、第二実施形態の符号化装置 2 0 0 の整数符号化部 2 1 0 における図 9 (A , B) と図 1 0 (A , B , C , D) を用いた符号化処理により得られる符号は、それぞれ、第一実施形態の符号化装置 1 0 0 の整数符号化部 1 1 0 における図 4 と図 5 の符号木を用いる符号化処理において、最初に用いる符号木を T(1) にした場合に得られる符号に一致する。

また、第二実施形態の符号化装置 2 0 0 の整数符号化部 2 1 0 にて、図 9 (A , B) と図 1 0 (A , B , C , D) のそれぞれを用いて符号化する際に、入力された整数系列 x_1, x_2, \dots, x_N の先頭に (2^{k-1}) 個の零値を追加してから符号化を行い、得られた整数符号の最初にある符号 ' 1 ' を取り除くようにすれば、第一実施形態の符号化装置 1 0 0 の整数符号化部 1 1 0 における図 4 と図 5 のそれぞれの符号木を用いる符号化処理において、最初に用いる符号木を T(0) にした場合に得られる符号に一致する。このように、第二実施形態の符号化装置 2 0 0 の整数符号化部 2 1 0 における対応表を用いた符号化処理は、第一実施形態の符号化装置 1 0 0 の整数符号化部 1 1 0 における符号木を用いた符号化処理と等価なものである。

【 0 0 7 2 】

復号装置

図 7 及び図 8 を参照して、第二実施形態の復号装置 2 5 0 が実行する復号方法の処理手続きを説明する。第二実施形態の復号装置 2 5 0 は、図 7 に示す通り、第一実施形態の復号装置 1 5 0 と同様、整数復号部 2 6 0 を含む。第二実施形態の復号装置 2 5 0 が、第一実施形態の復号装置 1 5 0 と同様、図 8 に示す各ステップの処理を実行することにより、第二実施形態の復号方法が実現される。

【 0 0 7 3 】

第二実施形態の復号装置 2 5 0 には、第一実施形態の復号装置 1 5 0 と同様、第二実施形態の符号化装置 2 0 0 が出力した整数符号が入力される。第二実施形態の復号装置 2 5 0 は、入力された整数符号を、第二実施形態の符号化装置 2 0 0 の整数符号化部 2 1 0 に記憶された対応表と対応表を用いた復号処理で復号して非負の整数値の系列を得ることで、第二実施形態の符号化装置 2 0 0 に入力された非負の整数値の系列そのものを復元するものである。以下では、上述した符号化装置の説明と同様に、非負の整数値を単に「整数値」と呼んで説明する。

【 0 0 7 4 】

[整数復号部 2 6 0]

整数復号部 2 6 0 には、復号装置 2 5 0 に入力された整数符号が入力される。整数復号部 2 6 0 は、予め作成して整数復号部 2 6 0 内に記憶された対応表であり、符号化パラメータ K に対応する予め作成された対応表であり、符号化装置 2 0 0 の整数符号化部 2 1 0

10

20

30

40

50

内に予め記憶され整数符号化部 2 1 0 で用いられたのと同じ対応表に基づいて、入力された整数符号を復号して整数系列 x_1, x_2, \dots, x_N を得て出力する (S 2 6 0)。

【 0 0 7 5 】

より具体的には、整数復号部 2 6 0 は、入力された整数符号のうちのまだ読み込んでいない符号部分を最初から対応表に対応する符号がある部分まで読み込み、読み込んだ符号に対応する整数の部分系列を対応表から得る処理を、入力された整数符号の最初から最後まで繰り返し行うことで得た整数の部分系列を纏めた系列を整数系列 x_1, x_2, \dots, x_N として出力する。この復号処理により、第一実施形態の復号装置 1 5 0 の整数復号部 1 6 0 における図 4 と図 5 のそれぞれの符号木を用いる復号処理において、最初に用いる符号木を T(1)にした場合と同じ整数系列 x_1, x_2, \dots, x_N を得ることができる。

10

【 0 0 7 6 】

なお、符号化装置 2 0 0 の整数符号化部 2 1 0 にて、入力された整数系列 x_1, x_2, \dots, x_N の先頭に $(2^{k_1} - 1)$ 個の零値を追加してから符号化を行い、得られた整数符号の最初にある符号 ' 1 ' を取り除く処理を行った場合は、整数復号部 2 6 0 は、入力された整数符号の先頭に符号 ' 1 ' を追加してから上記対応表に基づいて上述した復号処理で各部分系列を得、はじめに得られた $(2^{k_1} - 1)$ 個の零値を省略した後、部分系列を纏めた整数系列 x_1, x_2, \dots, x_N を出力すればよい。この復号処理により、第一実施形態の復号装置 1 5 0 の整数復号部 1 6 0 における図 4 と図 5 のそれぞれの符号木を用いる復号処理において、最初に用いる符号木を T(0)にした場合と同じ整数系列 x_1, x_2, \dots, x_N を得ることができる。このように、第二実施形態の復号装置 2 5 0 の整数復号部 2 6 0 における対応表を用いた復号処理は、第一実施形態の復号装置 1 5 0 の整数復号部 1 6 0 における符号木を用いた復号処理と等価なものである。

20

【 0 0 7 7 】

以上から分かるように、第一実施形態の符号化装置 1 0 0 によって符号木を用いた符号化処理で得た整数符号を、第二実施形態の復号装置 2 5 0 によって当該符号化処理と対応する対応表を用いた復号処理で復号して整数系列 x_1, x_2, \dots, x_N を得たり、第二実施形態の符号化装置 2 0 0 によって対応表を用いた符号化処理で得た整数符号を、第一実施形態の復号装置 1 5 0 によって当該符号化処理と対応する符号木を用いた復号処理で復号して整数系列 x_1, x_2, \dots, x_N を得たりしてもよい。

【 0 0 7 8 】

< 第三実施形態 >

第一実施形態及び第二実施形態では、予め定めた符号木または予め定めた対応表を用いて符号化復号する手続き的側面から本発明の符号化と復号を実現する形態について説明した。第三実施形態では、これらの手続きにより得られる符号や整数値の系列というデータの側面から本発明の符号化と復号を実現する形態について説明する。

30

【 0 0 7 9 】

符号化装置

図 2 及び図 3 を参照して、第三実施形態の符号化装置 3 0 0 が実行する符号化方法の処理手続きを説明する。第三実施形態の符号化装置 3 0 0 は、図 2 に示す通り、符号化装置 1 0 0 や符号化装置 2 0 0 と同様、整数符号化部 3 1 0 を含む。第三実施形態の符号化装置 3 0 0 が、符号化装置 1 0 0 や符号化装置 2 0 0 と同様、図 3 に示す各ステップの処理を実行することにより、第三実施形態の符号化方法が実現される。

40

【 0 0 8 0 】

第三実施形態の符号化装置 3 0 0 には、符号化装置 1 0 0 や符号化装置 2 0 0 と同様に、非負の整数値の系列 (以下、「整数系列」という) が入力される。以下では、符号化装置 3 0 0 に入力された非負の整数値を単に「整数値」と呼んで説明する。

【 0 0 8 1 】

[整数符号化部 3 1 0]

整数符号化部 3 1 0 は、整数系列に含まれる整数値 0 の 2 個から 2^{k_1} 個のうちの予め定めた何れかの個数の連続に対応する符号として 1 ビットの符号 (以下、「符号 A」という)

50

を得、整数系列に含まれる、0個から $2^{K+1}-1$ 個連続する整数値0と、1個の0以外の整数値 x_n と、による組、に対応する符号として $K \times x_n$ ビットまたは $K \times x_{n+1}$ ビットの符号（以下、「符号B」という）を得る（S310）。ただし、Kは2以上の自然数である符号化パラメータを表す。具体的には、Nサンプルずつの整数系列 x_1, x_2, \dots, x_N を、第一実施形態で説明した符号木や第二実施形態で説明した対応表を用いて符号化して、符号Aや符号Bを得る。ここで、符号Aに対応する整数値0の連続に含まれる0の個数を $L(2^L - 2^{K+1})$ とすると、S310で得られる符号Bは、整数値0の0個から $L-1$ 個、つまり、予め定めた最大で $2^{K+1}-1$ 個の連続とそれに後続する1つの0以外の整数値 x_n に対応する $K \times x_n$ ビットまたは $K \times x_{n+1}$ ビットの符号である。

【0082】

また、ここで、ビット値"x"はビット値"1"でありビット値"y"はビット値"0"であるとして、符号Aは、ビット値が"x"である1ビットの符号であり（図9B、図10D参照）、符号Bは、1ビット目からKビット目までに少なくとも1つのビット値"x"と少なくとも1つのビット値"y"を含み、かつ、最後から $K \times (x_n - 1)$ ビットはビット値が"y"である（図9A及び図9B、図10A～図10D参照）。なお、先述の通り、ビット値"x"はビット値"0"でありビット値"y"はビット値"1"であるとしてもよい。

【0083】

さらに、符号Bについては、より詳しくは、整数系列における整数値0の0個から $2^{K+1}-1$ 個の 2^{K+1} 通りの連続個数と、以下の 2^{K+1} 通りの部分符号の間に一対一の対応関係が存在する。

（符号Bの部分符号）

(1) 符号Bが $K \times x_{n+1}$ ビットであるときの、該符号Bの1ビット目から $K+1$ ビット目までの $K+1$ ビットで構成され、1ビット目のビット値が"x"であり、2ビット目から $K+1$ ビット目までのビット値が"y"である、1通りの部分符号

(2) 符号Bが $K \times x_n$ ビットであるときの、該符号Bの1ビット目からKビット目までのKビットで構成され、1ビット目のビット値が"y"であり、2ビット目からKビット目までのK-1ビット中の少なくとも1ビットのビット値が"x"である、 $2^{K+1}-1$ 通りの部分符号

【0084】

復号装置

図7及び図8を参照して、第三実施形態の復号装置350が実行する復号方法の処理手続きを説明する。第三実施形態の復号装置350は、図7に示す通り、復号装置150や復号装置250と同様、整数復号部360を含む。第三実施形態の復号装置350が、復号装置150や復号装置250と同様、図8に示す各ステップの処理を実行することにより、第三実施形態の復号方法が実現される。

【0085】

第三実施形態の復号装置350には、復号装置150や復号装置250と同様、第三実施形態の符号化装置300が出力した符号列が入力される。以下では、先述した符号化装置の説明と同様に、非負の整数値を単に「整数値」と呼んで説明する。

【0086】

[整数復号部360]

整数復号部360は、入力された符号列に含まれる1ビットの符号（以下「符号A」という）から、整数値0の2個から 2^{K+1} 個のうちの予め定めた何れかの個数の連続を得、入力された符号列に含まれる $K \times x_n$ ビットまたは $K \times x_{n+1}$ ビットの符号（以下、「符号B」という）から、0個から $2^{K+1}-1$ 個連続する整数値0と、1個の0以外の整数値 x_n と、による組を得る（S360）。ただし、Kは2以上の自然数である符号化パラメータを表す。具体的には、入力された符号列に含まれる符号を、第一実施形態で説明した符号木や第二実施形態で説明した対応表を用いて復号して、符号Aや符号Bに対応する上記整数値の系列を得る。ここで、符号Aから得られる整数値0の連続に含まれる0の個数を $L(2^L - 2^{K+1})$ とすると、S360で符号Bから得られる整数値0の連続に含まれる0の個数は0個から $L-1$ 個となる。

10

20

30

40

50

【 0 0 8 7 】

また、ここで、ビット値 "x" はビット値 "1" でありビット値 "y" はビット値 "0" であるとして、符号 A は、ビット値が "x" である 1 ビットの符号であり（図 9 B、図 1 0 D 参照）、符号 B は、1 ビット目から K ビット目までに少なくとも 1 つのビット値 "x" と少なくとも 1 つのビット値 "y" を含み、かつ、最後から $K \times (x \ n \ 1)$ ビットはビット値が "y" である（図 9 A 及び図 9 B、図 1 0 A ~ 図 1 0 D 参照）。なお、符号化装置 3 0 0 と同様、ビット値 "x" はビット値 "0" でありビット値 "y" はビット値 "1" であるとしてもよい。

【 0 0 8 8 】

整数復号部 3 6 0 は、以下の 2 つの処理を行うことにより、入力された符号列に含まれる符号 A と符号 B を特定することができる。具体的には、整数復号部 3 6 0 は、入力された符号列の最初から順に、入力された符号列を、連続する K ビット（以下、「K ビット列」という）ずつ読みこむことと、読み込まれた K ビット列が "y" を含むものであるとき、該 "y" を含む K ビット列の直前のビットが、前記符号 A または前記符号 B の終端のビットであると識別することと、を行う。

【 0 0 8 9 】

さらに、符号 B については、より詳しくは、以下の 2^{K-1} 通りの部分符号と、整数復号部 3 6 0 により得られた整数値による系列における整数値 0 の 0 個から 2^{K-1} 個の 2^{K-1} 通りの連続個数との間に一対一の対応関係が存在する。

（符号 B の部分符号）

(1) 符号 B が $K \times x \ n + 1$ ビットであるときの、該符号 B の 1 ビット目から K+1 ビット目までの K+1 ビットで構成され、1 ビット目のビット値が "x" であり、2 ビット目から K+1 ビット目までのビット値が "y" である、1 通りの部分符号

(2) 符号 B が $K \times x \ n$ ビットであるときの、該符号 B の 1 ビット目から K ビット目までの K ビットで構成され、1 ビット目のビット値が "y" であり、2 ビット目から K ビット目までの K-1 ビット中の少なくとも 1 ビットのビット値が "x" である、 2^{K-1} 通りの部分符号

【 0 0 9 0 】

符号列のデータ構造

第三実施形態の符号化装置及び復号装置が扱う符号列のデータ構造、つまり、第一実施形態及び第二実施形態の符号化装置が整数値の系列から得た符号列であり、第一実施形態及び第二実施形態の復号装置が整数値の系列を得るために用いる符号列のデータ構造について説明する。

【 0 0 9 1 】

[符号列のデータ構造]

非負値である整数値の系列（以下、「整数系列」という）を表す符号列は、整数系列から符号列を得る符号化装置や符号列から整数系列を得る復号装置で用いられるものである。

【 0 0 9 2 】

符号列のデータ構造は、整数系列に含まれる整数値 0 の 2 個から 2^{K-1} 個のうちの予め定められた何れかの個数の連続に対応する符号である 1 ビットの符号（以下、「符号 A」という）と、整数系列に含まれる、0 個から 2^{K-1} 個連続する整数値 0 と、1 個の 0 以外の整数値 $x \ n$ と、による組、に対応する符号である $K \times x \ n$ ビットまたは $K \times x \ n + 1$ ビットの符号（以下、「符号 B」という）と、を含む。ただし、K は 2 以上の自然数である符号化パラメータを表す。ここで、符号 A に対応する整数値 0 の連続に含まれる 0 の個数を $L(2 \leq L \leq 2^{K-1})$ とすると、符号 B は、整数値 0 の 0 個から L-1 個、つまり、予め定められた最大で 2^{K-1} 個の連続とそれに後続する 1 つの 0 以外の整数値 $x \ n$ に対応する $K \times x \ n$ ビットまたは $K \times x \ n + 1$ ビットの符号である。

【 0 0 9 3 】

また、ここで、ビット値 "x" はビット値 "1" でありビット値 "y" はビット値 "0" であるとして、符号 A は、ビット値が "x" である 1 ビットの符号であり（図 9 B、図 1 0

D参照)、符号Bは、1ビット目からKビット目までに少なくとも1つのビット値"x"と少なくとも1つのビット値"y"を含み、かつ、最後からK×(x n 1)ビットはビット値が"y"である(図9A及び図9B、図10A~図10D参照)。なお、先述の通り、ビット値"x"はビット値"0"でありビット値"y"はビット値"1"であるとしてもよい。

【0094】

さらに、符号Bについては、より詳しくは、整数系列における整数値0の0個から $2^{K+1}-1$ 個の 2^{K+1} 通りの連続個数と、以下の 2^{K+1} 通りの部分符号の間の一対一の対応関係が存在する。

(符号Bの部分符号)

(1) 符号BがK×x n+1ビットであるときの、該符号Bの1ビット目からK+1ビット目までのK+1ビットで構成され、1ビット目のビット値が"x"であり、2ビット目からK+1ビット目までのビット値が"y"である、1通りの部分符号

(2) 符号BがK×x nビットであるときの、該符号Bの1ビット目からKビット目までのKビットで構成され、1ビット目のビット値が"y"であり、2ビット目からKビット目までのK-1ビット中の少なくとも1ビットのビット値が"x"である、 $2^{K+1}-1$ 通りの部分符号

【0095】

本実施形態の発明によれば、零値ではない小さな値も含んで小さな値に大きく偏っている整数値の系列や、非特許文献2の技術では対応しきれないくらいに値の偏った整数値の系列であっても、少ない平均ビット数で符号化することができる。また、そのような符号を復号することができる。

【0096】

<第四実施形態>

上述の通り、本発明の符号に用いる符号化パラメータKは、その逆数の対数がGolomb Rice符号におけるRiceパラメータに対応し、本発明の符号は負値のRiceパラメータに対するGolomb Rice符号としても解釈できる。従って、従来の最適なRiceパラメータの推定法を基に最適な符号化パラメータKを入力された整数値による系列の部分系列に対して求め、求めた符号化パラメータKを用いた第一実施形態から第三実施形態の整数符号化部の処理をGolomb Rice符号化と組合せて入力された整数値の系列を符号化してもよく、この符号化に対応する復号を行ってもよい。この形態を第四実施形態として説明する。

【0097】

符号化装置

図11及び図12を参照して、第四実施形態の符号化装置400が実行する符号化方法の処理手続きを説明する。第四実施形態の符号化装置400は、図11に示す通り、パラメータ決定部420と整数符号化部410を含む。第四実施形態の符号化装置400が図12に示す各ステップの処理を実行することにより、第四実施形態の符号化方法が実現される。

【0098】

第四実施形態の符号化装置400には、第一から第三実施形態と同様に、非負の整数値の系列が入力される。第四実施形態の符号化装置400に入力された非負の整数値の系列はNサンプルずつパラメータ決定部420と整数符号化部410に入力される。すなわち、第四実施形態の符号化装置400は、例えばマイクロホンで収録した音声や音楽などをデジタル信号に変換して量子化して得た整数値の絶対値による系列などの整数値の系列を、Nサンプルによるフレームごとに符号化する。

【0099】

以下では、本実施形態の符号化装置に入力された非負の整数値を単に「整数値」と呼んで説明する。

【0100】

[パラメータ決定部420]

パラメータ決定部420には、符号化装置400に入力された整数値の系列のうち、Nサンプルずつの整数系列 x_1, x_2, \dots, x_N が入力される。パラメータ決定部420は、

入力された整数系列 x_1, x_2, \dots, x_N に基づき、その整数系列に対応するRiceパラメータ r とそのパラメータを表す符号であるパラメータ符号と、を得て出力する(S 4 2 0)。パラメータ符号は、復号装置4 5 0が当該パラメータ符号を復号することによりパラメータ決定部4 2 0が決定したRiceパラメータ r を得られるように、Riceパラメータを符号化して得ればよい。

【0 1 0 1】

パラメータ決定部4 2 0は、例えば、入力された整数系列 x_1, x_2, \dots, x_N に含まれる各整数値を用いて式(5)によりRiceパラメータ r を得る。

【0 1 0 2】

【数5】

$$r = \log_2 \left(\frac{\ln 2}{N} \sum_{n=1}^N x_{-n} \right) \dots (5)$$

【0 1 0 3】

式(5)で求まるRiceパラメータ r は、整数系列 x_1, x_2, \dots, x_N について式(2)により推定されるGolomb Rice符号化時の総ビット長の推定値を最小化するものである。

【0 1 0 4】

そして、パラメータ決定部4 2 0は、例えば、式(5)により得たRiceパラメータ r をスカラ量子化して符号を得て、得た符号をパラメータ符号として出力し、当該パラメータ符号に対応するRiceパラメータ r の量子化値をRiceパラメータ r として出力する。

【0 1 0 5】

[整数符号化部4 1 0]

整数符号化部4 1 0には、符号化装置4 0 0に入力された整数値の系列のうちの、 N サンプルずつの整数系列 x_1, x_2, \dots, x_N と、パラメータ決定部4 2 0が出力したRiceパラメータ r が入力される。整数符号化部4 1 0は、Riceパラメータ r が非負値である場合には、Riceパラメータ r の小数第一位を四捨五入するなどによりRiceパラメータ r を丸めて整数値にした後に、整数系列 x_1, x_2, \dots, x_N に対して整数値のRiceパラメータ r によるGolomb Rice符号化を行って、Golomb Rice符号化により得た符号を整数符号として出力し、Riceパラメータ r が負値である場合には、Riceパラメータ r から $K=2$ により K の値を得て、得た K の値の小数第一位を四捨五入するなどにより K の値を丸めて整数値にすることで符号化パラメータ K を得て、得た符号化パラメータ K を用いて第一から第三実施形態の整数符号化部の何れかの処理と同じ処理により整数符号を得て出力する(S 4 1 0)。なお、符号化パラメータ K が1と2の間の値である場合には、例えば、丸め後の整数値を2とすればよい。

【0 1 0 6】

復号装置

図1 3及び図1 4を参照して、第四実施形態の復号装置4 5 0が実行する復号方法の処理手続きを説明する。第四実施形態の復号装置4 5 0は、図1 3に示す通り、パラメータ復号部4 7 0と整数復号部4 6 0を含む。第四実施形態の復号装置4 5 0が図1 4に示す各ステップの処理を実行することにより、第四実施形態の復号方法が実現される。

【0 1 0 7】

第四実施形態の復号装置4 5 0には、第四実施形態の符号化装置4 0 0が出力した整数符号、及びパラメータ符号が入力される。復号装置4 5 0に入力された整数符号及びパラメータ符号は、整数値の系列の N サンプルに対応する符号ごとにパラメータ復号部4 7 0と整数復号部4 6 0に入力される。すなわち、復号装置4 5 0は、符号化装置4 0 0と同じフレームごとに復号処理を行う。

【0 1 0 8】

[パラメータ復号部4 7 0]

パラメータ復号部4 7 0には、復号装置4 5 0に入力されたパラメータ符号が入力される。パラメータ復号部4 7 0は、パラメータ決定部4 2 0がパラメータ符号を得たのと対

10

20

30

40

50

応する復号処理により、パラメータ符号を復号してRiceパラメータ r を得て出力する（S 470）。パラメータ復号部470は、例えば、符号化装置400のパラメータ決定部420が行ったスカラ量子化に対応する復号処理により、パラメータ符号に対応するRiceパラメータ r の量子化値をRiceパラメータ r として得て出力する。

【0109】

[整数復号部460]

整数復号部460には、復号装置450に入力された整数符号と、パラメータ復号部470が出力したRiceパラメータ r が入力される。整数復号部460は、Riceパラメータ r が非負値である場合には、Riceパラメータ r を符号化装置400の整数符号化部410が行ったのと同じ方法により丸めて整数値にした後に整数符号に対してRiceパラメータ r によるGolomb Rice復号を行って整数系列 x_1, x_2, \dots, x_N を得て出力し、Riceパラメータ r が負値である場合には、第四実施形態の整数符号化部410と同じ処理でRiceパラメータ r を符号化パラメータ K に変換して得て、すなわち、Riceパラメータ r から $K=2^{|r|}$ により K の値を得て、符号化装置400の整数符号化部410が行ったのと同じ方法により K の値を丸めて整数値にすることで符号化パラメータ K を得て、得た符号化パラメータ K を用いて第一から第三実施形態の整数復号部の何れかの処理と同じの処理により整数系列 x_1, x_2, \dots, x_N を得て出力する（S 460）。

【0110】

<第四実施形態の変形例1>

第四実施形態では、Golomb Rice符号化・復号と組み合わせた符号化・復号処理を行ったが、Golomb Rice符号化・復号と組み合わせないでもよい。Golomb Rice符号化・復号と組み合わせない形態を第四実施形態の変形例1として説明する。符号化装置401のパラメータ決定部421と復号装置451のパラメータ復号部471の動作は第四実施形態と同じであり、符号化装置401の整数符号化部411と復号装置451の整数復号部461の動作が第四実施形態とは異なる。

【0111】

[整数符号化部411]

整数符号化部411には、符号化装置401に入力された整数値の系列のうち、 N サンプルずつの整数系列 x_1, x_2, \dots, x_N と、パラメータ決定部421が出力したRiceパラメータ r が入力される。整数符号化部411は、Riceパラメータ r が非負値である場合には、2を符号化パラメータ K として用いて、第一から第三実施形態の整数符号化部の何れかの処理と同じ処理により整数符号を得て出力し、Riceパラメータ r が負値である場合には、Riceパラメータ r から $K=2^{|r|}$ により K の値を得て、得た K の値の小数第一位を四捨五入するなどにより K の値を丸めて整数値にすることで符号化パラメータ K を得、得た符号化パラメータ K を用いて第一から第三実施形態の整数符号化部の何れかの処理と同じ処理により整数符号を得て出力する（S 411）。なお、符号化パラメータ K が1と2の間の値である場合には、例えば、丸め後の整数値を2とすればよい。

【0112】

[整数復号部461]

整数復号部461には、復号装置451に入力された整数符号と、パラメータ復号部471が出力したRiceパラメータ r が入力される。整数復号部461は、Riceパラメータ r が非負値である場合には、2を符号化パラメータ K として用いて、第一から第三実施形態の整数復号部の何れかの処理と同じの処理により整数系列 x_1, x_2, \dots, x_N を得て出力し、Riceパラメータ r が負値である場合には、第四実施形態の整数符号化部410と同じ処理でRiceパラメータ r を符号化パラメータ K に変換して得て、すなわち、Riceパラメータ r から $K=2^{|r|}$ により K の値を得て、符号化装置401の整数符号化部411が行ったのと同じ方法により K の値を丸めて整数値にすることで符号化パラメータ K を得て、得た符号化パラメータ K を用いて第一から第三実施形態の整数復号部の何れかの処理と同じの処理により整数系列 x_1, x_2, \dots, x_N を得て出力する（S 461）。

【0113】

10

20

30

40

50

< 第四実施形態の変形例 2 >

第四実施形態では、整数符号化部 4 1 0 においてRiceパラメータrが負値である場合にRiceパラメータrから符号化パラメータKを得る構成としていたが、パラメータ決定部 4 2 0 においてRiceパラメータrが負値である場合に整数系列から符号化パラメータKを直接得る構成としてもよい。この形態を第四実施形態の変形例 2 として、第四実施形態と異なる部分について説明する。

【 0 1 1 4 】

[パラメータ決定部 4 2 2]

第四実施形態のパラメータ決定部 4 2 0 は、Riceパラメータの値に関わらず、Riceパラメータと当該Riceパラメータに対応するパラメータ符号とを必ず出力する構成であったが、第四実施形態の変形例 2 のパラメータ決定部 4 2 2 は、Riceパラメータrが非負値である場合には、Riceパラメータrと当該Riceパラメータrに対応するパラメータ符号とを出力し、Riceパラメータrが負値である場合には、後述する式(6)により整数系列から符号化パラメータKを得て、符号化パラメータKと当該符号化パラメータKに対応するパラメータ符号とを出力する (S 4 2 2)。その際、パラメータ符号は、対応する復号装置 4 5 2 がRiceパラメータrの値と符号化パラメータKの値を一意に特定可能な符号とすればよい。例えば、Riceパラメータrを符号化して得た符号の前に1ビットの符号 ' 1 ' を付したものをパラメータ符号とし、符号化パラメータKを符号化して得た符号の前に1ビットの符号 ' 0 ' を付したものをパラメータ符号とするなどにより、Riceパラメータrに割り当てるパラメータ符号の符号群と、符号化パラメータKに割り当てる符号群とに同じ符号が含まれないようにすればよい。

【 0 1 1 5 】

[整数符号化部 4 1 2]

整数符号化部 4 1 2 には、符号化装置 4 0 2 に入力された整数値の系列のうちの、Nサンプルずつの整数系列 x_1, x_2, \dots, x_N と、パラメータ決定部 4 2 2 が出力したRiceパラメータrまたは符号化パラメータKが入力される。整数符号化部 4 1 2 は、Riceパラメータrが入力された場合には、Riceパラメータrを丸めて整数値にし、整数系列 x_1, x_2, \dots, x_N に対して整数値のRiceパラメータrによるGolomb Rice符号化を行って、Golomb Rice符号化により得た符号を整数符号として出力し、符号化パラメータKが入力された場合には、符号化パラメータKを丸めて整数値にし、整数値の符号化パラメータKを用いて第一から第三実施形態の整数符号化部の何れかの処理と同じ処理により整数符号を得て出力する (S 4 1 2)。なお、符号化パラメータKが1と2の間の値である場合には、例えば、丸め後の整数値を2とすればよい。

【 0 1 1 6 】

[パラメータ復号部 4 7 2]

第四実施形態のパラメータ復号部 4 7 0 はパラメータ符号を復号してRiceパラメータを得るものであったが、第四実施形態の変形例 2 のパラメータ復号部 4 7 2 は、パラメータ符号を復号してRiceパラメータrまたは符号化パラメータKを得て出力する (S 4 7 2)。

【 0 1 1 7 】

[整数復号部 4 6 2]

整数復号部 4 6 2 には、復号装置 4 5 2 に入力された整数符号と、パラメータ復号部 4 7 2 が出力したRiceパラメータrまたは符号化パラメータKが入力される。整数復号部 4 6 2 は、Riceパラメータrが入力された場合には、Riceパラメータrを符号化装置 4 0 2 の整数符号化部 4 1 2 が行ったのと同じ方法により丸めて整数値にした後に、整数符号に対して整数値のRiceパラメータrによるGolomb Rice復号を行って整数系列 x_1, x_2, \dots, x_N を得て出力し、符号化パラメータKが入力された場合には、符号化パラメータKを符号化装置 4 0 2 の整数符号化部 4 1 2 が行ったのと同じ方法により丸めて整数値にした後に、整数値の符号化パラメータKを用いて第一から第三実施形態の整数復号部の何れかの処理と同じの処理により整数系列 x_1, x_2, \dots, x_N を得て出力する (S 4 6 2)。

【 0 1 1 8 】

10

20

30

40

50

< 第五実施形態 >

第四実施形態の符号化装置では、まずRiceパラメータrを求めて、求めたRiceパラメータrから符号化パラメータKを求めていたが、符号化パラメータKを、本発明の符号が最適な分布 $p(x,1/K)$ に対して尤もらしいKを推定することにより、入力された整数値の系列から直接推定してもよい。この場合、Golomb Rice符号は1以下の値をとるKによる本発明の符号と解釈することができる。従って、最適な符号化パラメータKを入力された整数値による系列の部分系列から直接求め、求めた符号化パラメータKを用いた第一実施形態から第三実施形態の整数符号化部の処理をGolomb Rice符号化と組合せて入力された整数値の系列を符号化してもよく、この符号化に対応する復号を行ってもよい。この形態を第五実施形態として説明する。

10

【0119】

符号化装置

図11及び図12を参照して、第五実施形態の符号化装置500が実行する符号化方法の処理手続きを説明する。第五実施形態の符号化装置500は、図11に示す通り、パラメータ決定部520と整数符号化部510を含む。第五実施形態の符号化装置500が図12に示す各ステップの処理を実行することにより、第五実施形態の符号化方法が実現される。

【0120】

第五実施形態の符号化装置500には、第一から第四実施形態と同様に、非負の整数値の系列が入力される。第五実施形態の符号化装置500に入力された非負の整数値の系列はNサンプルずつパラメータ決定部520と整数符号化部510に入力される。すなわち、第五実施形態の符号化装置500は、例えばマイクロホンで収録した音声や音楽などをデジタル信号に変換して量子化して得た整数値の絶対値による系列などの整数値の系列を、Nサンプルによるフレームごとに符号化する。

20

【0121】

以下では、本実施形態の符号化装置に入力された非負の整数値を単に「整数値」と呼んで説明する。

【0122】

[パラメータ決定部520]

パラメータ決定部520には、符号化装置500に入力された整数値の系列のうち、Nサンプルずつの整数系列 x_1, x_2, \dots, x_N が入力される。パラメータ決定部520は、入力された整数系列 x_1, x_2, \dots, x_N に基づき、その整数系列に対応する符号化パラメータKとそのパラメータを表す符号であるパラメータ符号と、を得て出力する(S520)。パラメータ符号は、復号装置550が当該パラメータ符号を復号することによりパラメータ決定部520が決定した符号化パラメータKを得られるように、符号化パラメータKを符号化して得ればよい。

30

【0123】

パラメータ決定部520は、例えば、入力された整数系列 x_1, x_2, \dots, x_N に含まれる各整数値を用いて式(6)により符号化パラメータKを得る。式(4)を基に分布 $p(x,1/K)$ に対する整数系列 x_1, x_2, \dots, x_N の対数尤度を求め、次式のように対数尤度を最大化する符号化パラメータKを得て出力する。

40

【0124】

【数6】

$$K = \log_2 \left(1 + \frac{1}{\frac{1}{N} \sum_{n=1}^N x_n} \right) \dots (6)$$

【0125】

50

式(6)で求まる符号化パラメータKは、整数系列 x_1, x_2, \dots, x_N についての式(4)に基づく分布 $p(x, 1/K)$ に対する対数尤度を最大化するものである。

【0126】

そして、パラメータ決定部520は、例えば、式(6)により得た符号化パラメータKをスカラ量子化して符号を得て、得た符号をパラメータ符号として出力し、当該パラメータ符号に対応する符号化パラメータKの量子化値を符号化パラメータKとして出力する。

【0127】

[整数符号化部510]

整数符号化部510には、符号化装置500に入力された整数値の系列のうち、Nサンプルずつの整数系列 x_1, x_2, \dots, x_N と、パラメータ決定部520が出力した符号化パラメータKが入力される。整数符号化部510は、符号化パラメータKが1以下である場合には、符号化パラメータKから $r = \log_2 K$ によりrの値を得て、得たrの値の小数第一位を四捨五入するなどによりrの値を丸めて整数値にすることでRiceパラメータrを得て、得たRiceパラメータrを用いて整数系列 x_1, x_2, \dots, x_N に対してRiceパラメータrによるGolomb Rice符号化を行って、Golomb Rice符号化により得た符号を整数符号として出力し、符号化パラメータKが1よりも大きい場合には、符号化パラメータKの小数第一位を四捨五入するなどによりKの値を丸めて整数値にした後に、整数値の符号化パラメータKを用いて第一から第三実施形態の整数符号化部の何れかの処理と同じ処理により整数符号を得て出力する(S510)。なお、符号化パラメータKが1と2の間の値である場合には、例えば、丸め後の整数値を2とすればよい。

【0128】

復号装置

図13及び図14を参照して、第五実施形態の復号装置550が実行する復号方法の処理手続きを説明する。第五実施形態の復号装置550は、図13に示す通り、パラメータ復号部570と整数復号部560を含む。第五実施形態の復号装置550が図14に示す各ステップの処理を実行することにより、第五実施形態の復号方法が実現される。

【0129】

第五実施形態の復号装置550には、第五実施形態の符号化装置500が出力した整数符号、及びパラメータ符号が入力される。復号装置550に入力された整数符号及びパラメータ符号は、整数値の系列のNサンプルに対応する符号ごとにパラメータ復号部570と整数復号部560に入力される。すなわち、復号装置550は、符号化装置500と同じフレームごとに復号処理を行う。

【0130】

[パラメータ復号部570]

パラメータ復号部570には、復号装置550に入力されたパラメータ符号が入力される。パラメータ復号部570は、パラメータ決定部520がパラメータ符号を得たのに対応する復号処理により、パラメータ符号を復号して符号化パラメータKを得て出力する(S570)。パラメータ復号部570は、例えば、符号化装置500のパラメータ決定部520が行ったスカラ量子化に対応する復号処理により、パラメータ符号に対応する符号化パラメータKの量子化値を符号化パラメータKとして得て出力する。

【0131】

[整数復号部560]

整数復号部560には、復号装置550に入力された整数符号と、パラメータ復号部570が出力した符号化パラメータKが入力される。整数復号部560は、符号化パラメータKが1以下である場合には、第五実施形態の整数符号化部510と同じ処理で符号化パラメータKをRiceパラメータrに変換して得て、すなわち、符号化パラメータKから $r = \log_2 K$ によりrの値を得て、符号化装置500の整数符号化部510が行ったのと同じ方法によりrの値を丸めて整数値にすることでRiceパラメータrを得て、得たRiceパラメータrを用いて整数符号に対してRiceパラメータrによるGolomb Rice復号を行って整数系列 x_1, x_2, \dots, x_N を得て出力し、符号化パラメータKが1よりも大きい場合には、符号化パラメー

タKを符号化装置500の整数符号化部510が行ったのと同じ方法により丸めて整数値にした後に、整数値の符号化パラメータKを用いて第一から第三実施形態の整数復号部の何れかの処理と同じ処理により整数系列 x_1, x_2, \dots, x_N を得て出力する(S560)。

【0132】

<第五実施形態の変形例1>

第五実施形態では、Golomb Rice符号化・復号と組み合わせた符号化・復号処理を行ったが、Golomb Rice符号化・復号と組み合わせないでもよい。Golomb Rice符号化・復号と組み合わせない形態を第五実施形態の変形例1として説明する。符号化装置501のパラメータ決定部521と復号装置551のパラメータ復号部571の動作は第五実施形態と同じであり、符号化装置501の整数符号化部511と復号装置551の整数復号部561の動作が第五実施形態とは異なる。

【0133】

[整数符号化部511]

整数符号化部511には、符号化装置501に入力された整数値の系列のうちの、Nサンプルずつの整数系列 x_1, x_2, \dots, x_N と、パラメータ決定部521が出力した符号化パラメータKが入力される。整数符号化部511は、入力された符号化パラメータKの小数第一位を四捨五入するなどによりKの値を丸めて整数値にした後に、整数値の符号化パラメータKが1以下である場合には、2を符号化パラメータKとして用いて、第一から第三実施形態の整数符号化部の何れかの処理と同じ処理により整数符号を得て出力し、整数値の符号化パラメータKが1よりも大きい場合には、整数値の符号化パラメータKを用いて第一から第三実施形態の整数符号化部の何れかの処理と同じ処理により整数符号を得て出力する(S511)。

【0134】

[整数復号部561]

整数復号部561には、復号装置551に入力された整数符号と、パラメータ復号部571が出力した符号化パラメータKが入力される。整数復号部561は、入力された符号化パラメータKを符号化装置501の整数符号化部511が行ったのと同じ方法により丸めて整数値にした後に、整数値の符号化パラメータKが1以下である場合には、2を符号化パラメータKとして用いて、第一から第三実施形態の整数復号部の何れかの処理と同じ処理により整数系列 x_1, x_2, \dots, x_N を得て出力し、整数値の符号化パラメータKが1よりも大きい場合には、整数値の符号化パラメータKを用いて第一から第三実施形態の整数復号部の何れかの処理と同じ処理により整数系列 x_1, x_2, \dots, x_N を得て出力する(S561)。

【0135】

<第五実施形態の変形例2>

第五実施形態では、整数符号化部510において符号化パラメータKが1以下である場合に符号化パラメータKからRiceパラメータrを得る構成としていたが、パラメータ決定部520において符号化パラメータKが1以下である場合に整数系列からRiceパラメータrを直接得る構成としてもよい。この形態を第五実施形態の変形例2として、第五実施形態と異なる部分について説明する。

【0136】

[パラメータ決定部522]

第五実施形態のパラメータ決定部520は、符号化パラメータKの値に関わらず、符号化パラメータKと当該符号化パラメータKに対応するパラメータ符号とを必ず出力する構成であったが、第五実施形態の変形例2のパラメータ決定部522は、符号化パラメータKが1よりも大きい場合には、符号化パラメータKと当該符号化パラメータKに対応するパラメータ符号とを出力し、符号化パラメータKが1以下である場合には、式(5)により整数系列からRiceパラメータrを得て、Riceパラメータrと当該Riceパラメータrに対応するパラメータ符号とを出力する(S522)。その際、第四実施形態の変形例2と同様に、パラ

メータ符号は、対応する復号装置 5 5 2 が Rice パラメータ r の値と符号化パラメータ K の値を一意に特定可能な符号とすればよい。

【 0 1 3 7 】

[整数符号化部 5 1 2]

整数符号化部 5 1 2 には、符号化装置 5 0 2 に入力された整数値の系列のうちの、 N サンプルずつの整数系列 x_1, x_2, \dots, x_N と、パラメータ決定部 5 2 2 が出力した符号化パラメータ K または Rice パラメータ r が入力される。整数符号化部 5 1 2 は、Rice パラメータ r が入力された場合には、Rice パラメータ r の値を丸めて整数値にした後に、整数系列 x_1, x_2, \dots, x_N に対して整数値の Rice パラメータ r による Golomb Rice 符号化を行って、Golomb Rice 符号化により得た符号を整数符号として出力し、符号化パラメータ K が入力された場合には、符号化パラメータ K を丸めて整数値にした後に、整数値の符号化パラメータ K を用いて第一から第三実施形態の整数符号化部の何れかの処理と同じ処理により整数符号を得て出力する (S 5 1 2)。なお、符号化パラメータ K が 1 と 2 の間の値である場合には、例えば、丸め後の整数値を 2 とすればよい。

10

【 0 1 3 8 】

[パラメータ復号部 5 7 2]

第五実施形態のパラメータ復号部 5 7 0 はパラメータ符号を復号して符号化パラメータ K を得るものであったが、第五実施形態の変形例 2 のパラメータ復号部 5 7 2 は、パラメータ符号を復号して符号化パラメータ K または Rice パラメータ r を得て出力する (S 5 7 2)。

20

【 0 1 3 9 】

[整数復号部 5 6 2]

整数復号部 5 6 2 には、復号装置 5 5 2 に入力された整数符号と、パラメータ復号部 5 7 2 が出力した符号化パラメータ K または Rice パラメータ r が入力される。整数復号部 5 6 2 は、Rice パラメータ r が入力された場合には、符号化装置 5 0 2 の整数符号化部 5 1 2 が行ったのと同じ方法により Rice パラメータ r の値を丸めて整数値とし、整数符号に対して整数値の Rice パラメータ r による Golomb Rice 復号を行って整数系列 x_1, x_2, \dots, x_N を得て出力し、符号化パラメータ K が入力された場合には、符号化パラメータ K を符号化装置 5 0 2 の整数符号化部 5 1 2 が行ったのと同じ方法により丸めて整数値にした後に、整数値の符号化パラメータ K を用いて第一から第三実施形態の整数復号部の何れかの処理と同じ処理により整数系列 x_1, x_2, \dots, x_N を得て出力する (S 5 6 2)。

30

【 0 1 4 0 】

< 第四実施形態及び第五実施形態のその他の変形 >

なお、Rice パラメータ r と符号化パラメータ K とは、 $K=2^r$ および $r=\log_2 K$ により相互に変換可能なものであるので、符号化装置の整数符号化部と復号装置の整数復号部で同じパラメータの同じ値を用いるようにできるのであれば、何れのパラメータを符号化・復号の対象とするかと上記の変換とをどのように組み合わせて用いてもよい。また、符号化装置の整数符号化部と復号装置の整数復号部で同じパラメータの同じ値を用いるようにできるのであれば、Rice パラメータ r と符号化パラメータ K のどちらの値を丸め処理の対象としてもよいし、パラメータの符号化前後 (すなわち量子化前後) のどちらの値を丸めの処理の対象としてもよいし、符号化における値の量子化と丸めの処理を同時に行うようにしてもよい。要するに、第四実施形態及び第五実施形態及びこれらの変形例 1 と変形例 2 で説明したのはあくまでも例であって、 N サンプルによるフレームごとに、符号化装置の整数符号化部と復号装置の整数復号部で同じパラメータの同じ値を用いた符号化処理と復号処理を行えるようにすればよい。

40

【 0 1 4 1 】

< 第六実施形態 >

第四実施形態及び第五実施形態では、入力された整数値による系列の部分系列に対して求めた Rice パラメータ r または符号化パラメータ K を用いて第一実施形態から第三実施形態の符号化・復号をすることにより、本発明の符号化と復号を実現する形態について説明し

50

た。第六実施形態では、Riceパラメータ r と符号化パラメータ K の少なくとも何れかを用いて、本発明の符号化と復号を実現する形態について説明する。

【0142】

符号化装置

図11及び図12を参照して、第六実施形態の符号化装置600が実行する符号化方法の処理手続きを説明する。第六実施形態の符号化装置600は、図11に示す通り、符号化装置400や符号化装置500と同様、パラメータ決定部620と整数符号化部610を含む。第六実施形態の符号化装置600が、符号化装置400や符号化装置500と同様、図12に示す各ステップの処理を実行することにより、第六実施形態の符号化方法が実現される。

10

【0143】

第六実施形態の符号化装置600には、符号化装置400や符号化装置500と同様に、非負の整数値の系列が入力される。第六実施形態の符号化装置600に入力された非負の整数値の系列は N サンプルずつパラメータ決定部620と整数符号化部610に入力される。

【0144】

以下では、本実施形態の符号化装置に入力された非負の整数値を単に「整数値」と呼んで説明する。

【0145】

[パラメータ決定部620]

パラメータ決定部620には、符号化装置600に入力された整数値の系列のうち、 N サンプルずつの整数系列 x_1, x_2, \dots, x_N が入力される。パラメータ決定部620は、入力された整数系列 x_1, x_2, \dots, x_N に基づき、その整数系列に対応するRiceパラメータ r と符号化パラメータ K の少なくとも何れかと、そのRiceパラメータ r と符号化パラメータ K の少なくとも何れかを表すパラメータ符号と、を得て出力する(5620)。つまり、パラメータ決定部620は、入力された非負の整数値による系列に含まれる所定サンプル毎の整数値による系列である整数系列ごとに、整数系列に含まれる整数値に基づいて式(5)によりライスパラメータ r を求める処理と、整数系列に含まれる整数値に基づいて式(6)により符号化パラメータ K を求める処理と、の少なくとも何れかと、ライスパラメータ r と符号化パラメータ K の少なくとも何れかを符号化してパラメータ符号を得る処理と、を

20

30

【0146】

[整数符号化部610]

整数符号化部610には、符号化装置600に入力された整数値の系列のうち、 N サンプルずつの整数系列 x_1, x_2, \dots, x_N と、パラメータ決定部620が出力したRiceパラメータ r と符号化パラメータ K の少なくとも何れかが入力される。整数符号化部610は、符号化パラメータ K が入力された場合には、符号化パラメータ K が1以下であれば、第五実施形態と同様、Golomb Rice符号化を行って、Golomb Rice符号化により得た符号を整数符号として出力し、符号化パラメータ K が1よりも大きければ、第五実施形態と同様、第一から第三実施形態の整数符号化部の何れかの処理と同じ処理により整数符号を得て出力し、Riceパラメータ r が入力された場合には、Riceパラメータ r が非負値であれば、第四実施形態と同様、Golomb Rice符号化を行って、Golomb Rice符号化により得た符号を整数符号として出力し、Riceパラメータ r が負値であれば、第四実施形態と同様、第一から第三実施形態の整数符号化部の何れかの処理と同じ処理により整数符号を得て出力する(5610)。つまり、整数符号化部610は、所定サンプル毎の整数値による系列である整数系列ごとに、パラメータ決定部620が得た符号化パラメータ K 、または、パラメータ決定部620が得たRiceパラメータ r から $K=2^r$ により得た符号化パラメータ K 、を符号化パラメータ K として用いて符号Aや符号Bの符号を得る。特に、整数系列に対応する符号化パラメータ K が1以下である場合、または、整数系列に対応するRiceパラメータ r が非負である場合、には、整数系列をGolomb Rice符号化して符号列を得るようにしてもよい。

40

50

【 0 1 4 7 】

復号装置

図 1 3 及び図 1 4 を参照して、第六実施形態の復号装置 6 5 0 が実行する復号方法の処理手続きを説明する。第六実施形態の復号装置 6 5 0 は、図 1 3 に示す通り、復号装置 4 5 0 や復号装置 5 5 0 と同様、パラメータ復号部 6 7 0 と整数復号部 6 6 0 を含む。第六実施形態の復号装置 6 5 0 が、復号装置 4 5 0 や復号装置 5 5 0 と同様、図 1 4 に示す各ステップの処理を実行することにより、第六実施形態の復号方法が実現される。

【 0 1 4 8 】

第六実施形態の復号装置 6 5 0 には、第六実施形態の符号化装置 6 0 0 が出力した整数符号、及びパラメータ符号が入力される。復号装置 6 5 0 に入力された整数符号及びパラメータ符号は、整数値の系列の N サンプルに対応する符号ごとにパラメータ復号部 6 7 0 と整数復号部 6 6 0 に入力される。つまり、パラメータ復号部 6 7 0 と整数復号部 6 6 0 には、所定単位ごとのパラメータ符号と符号列とが入力される。

【 0 1 4 9 】

[パラメータ復号部 6 7 0]

パラメータ復号部 6 7 0 は、パラメータ決定部 6 2 0 がパラメータ符号を得たのに対応する復号処理により、パラメータ符号を復号して Rice パラメータ r と符号化パラメータ K の少なくとも何れかを得て出力する (S 6 7 0)。つまり、パラメータ復号部 6 7 0 は、所定単位ごとに、パラメータ符号を復号して Rice パラメータ r と符号化パラメータ K の少なくとも何れかを得る。

【 0 1 5 0 】

[整数復号部 6 6 0]

整数復号部 6 6 0 には、復号装置 6 5 0 に入力された整数符号と、パラメータ復号部 6 7 0 が出力した Rice パラメータ r と符号化パラメータ K の少なくとも何れかが入力される。整数復号部 6 6 0 は、符号化パラメータ K が入力された場合には、符号化パラメータ K が 1 以下であれば、第五実施形態と同様、Golomb Rice 復号を行って整数系列 x_1, x_2, \dots, x_N を得て出力し、符号化パラメータ K が 1 よりも大きければ、第五実施形態と同様、第一から第三実施形態の整数復号部の何れかの処理と同じ処理により整数系列 x_1, x_2, \dots, x_N を得て出力し、Rice パラメータ r が入力された場合には、Rice パラメータ r が非負値であれば、第四実施形態と同様、Golomb Rice 復号を行って整数系列 x_1, x_2, \dots, x_N を得て出力し、Rice パラメータ r が負値であれば、第四実施形態と同様、第一から第三実施形態の整数復号部の何れかの処理と同じの処理により整数系列 x_1, x_2, \dots, x_N を得て出力する (S 6 6 0)。つまり、整数復号部 6 6 0 は、所定単位ごとに、パラメータ復号部 6 7 0 が得た符号化パラメータ K 、または、パラメータ復号部 6 7 0 が得た Rice パラメータ r から $K=2^{\lfloor r \rfloor}$ により得た符号化パラメータ K 、を符号化パラメータ K として用いて整数値 0 の $2^{\lfloor r \rfloor}$ 個から $2^{\lfloor r \rfloor + 1}$ 個の連続や、0 個から $2^{\lfloor r \rfloor + 1}$ 個連続する整数値 0 と 1 個の 0 以外の整数値 x_n とによる組からなる、整数値による系列を得る。特に、符号化パラメータ K が 1 以下である場合、または、Rice パラメータ r が非負である場合、には、符号列を Golomb Rice 復号して整数値による系列を得るようにしてもよい。

【 0 1 5 1 】

本実施形態の発明によれば、整数値の系列や部分系列に対応する Rice パラメータ r に相当する値が分かれば、その系列に最適な符号化パラメータ K を用いることで、少ない平均ビット数で符号化することができる。また、そのような符号を復号することができる。

【 0 1 5 2 】

< 補記 >

本発明の装置は、例えば単一のハードウェアエンティティとして、キーボードなどが接続可能な入力部、液晶ディスプレイなどが接続可能な出力部、ハードウェアエンティティの外部に通信可能な通信装置 (例えば通信ケーブル) が接続可能な通信部、CPU (Central Processing Unit、キャッシュメモリやレジスタなどを備えていてもよい)、メモリである RAM や ROM、ハードディスクである外部記憶装置並びにこれらの入力部、出力

部、通信部、CPU、RAM、ROM、外部記憶装置の間のデータのやり取りが可能ないように接続するバスを有している。また必要に応じて、ハードウェアエンティティに、CD-ROMなどの記録媒体を読み書きできる装置（ドライブ）などを設けることとしてもよい。このようなハードウェア資源を備えた物理的実体としては、汎用コンピュータなどがある。

【0153】

ハードウェアエンティティの外部記憶装置には、上述の機能を実現するために必要となるプログラムおよびこのプログラムの処理において必要となるデータなどが記憶されている（外部記憶装置に限らず、例えばプログラムを読み出し専用記憶装置であるROMに記憶させておくこととしてもよい）。また、これらのプログラムの処理によって得られるデータなどは、RAMや外部記憶装置などに適宜に記憶される。

【0154】

ハードウェアエンティティでは、外部記憶装置（あるいはROMなど）に記憶された各プログラムとこの各プログラムの処理に必要なデータが必要に応じてメモリに読み込まれて、適宜にCPUで解釈実行・処理される。その結果、CPUが所定の機能（上記、部、手段などと表した各構成要件）を実現する。

【0155】

本発明は上述の実施形態に限定されるものではなく、本発明の趣旨を逸脱しない範囲で適宜変更が可能である。また、上記実施形態において説明した処理は、記載の順に従って時系列に実行されるのみならず、処理を実行する装置の処理能力あるいは必要に応じて並列的あるいは個別に実行されるとしてもよい。

【0156】

既述のように、上記実施形態において説明したハードウェアエンティティ（本発明の装置）における処理機能をコンピュータによって実現する場合、ハードウェアエンティティが有すべき機能の処理内容はプログラムによって記述される。そして、このプログラムをコンピュータで実行することにより、上記ハードウェアエンティティにおける処理機能がコンピュータ上で実現される。

【0157】

この処理内容を記述したプログラムは、コンピュータで読み取り可能な記録媒体に記録しておくことができる。コンピュータで読み取り可能な記録媒体としては、例えば、磁気記録装置、光ディスク、光磁気記録媒体、半導体メモリ等のようなものでもよい。具体的には、例えば、磁気記録装置として、ハードディスク装置、フレキシブルディスク、磁気テープ等を、光ディスクとして、DVD（Digital Versatile Disc）、DVD-RAM（Random Access Memory）、CD-ROM（Compact Disc Read Only Memory）、CD-R（Recordable）/RW（ReWritable）等を、光磁気記録媒体として、MO（Magneto Optical disc）等を、半導体メモリとしてEEPROM（Electrically Erasable and Programmable Read Only Memory）等を用いることができる。

【0158】

また、このプログラムの流通は、例えば、そのプログラムを記録したDVD、CD-ROM等の可搬型記録媒体を販売、譲渡、貸与等することによって行う。さらに、このプログラムをサーバコンピュータの記憶装置に格納しておき、ネットワークを介して、サーバコンピュータから他のコンピュータにそのプログラムを転送することにより、このプログラムを流通させる構成としてもよい。

【0159】

このようなプログラムを実行するコンピュータは、例えば、まず、可搬型記録媒体に記録されたプログラムもしくはサーバコンピュータから転送されたプログラムを、一旦、自己の記憶装置に格納する。そして、処理の実行時、このコンピュータは、自己の記憶媒体に格納されたプログラムを読み取り、読み取ったプログラムに従った処理を実行する。また、このプログラムの別の実行形態として、コンピュータが可搬型記録媒体から直接プログラムを読み取り、そのプログラムに従った処理を実行することとしてもよく、さらに、

このコンピュータにサーバコンピュータからプログラムが転送されるたびに、逐次、受け取ったプログラムに従った処理を実行することとしてもよい。また、サーバコンピュータから、このコンピュータへのプログラムの転送は行わず、その実行指示と結果取得のみによって処理機能を実現する、いわゆるASP (Application Service Provider) 型のサービスによって、上述の処理を実行する構成としてもよい。なお、本形態におけるプログラムには、電子計算機による処理の用に供する情報であってプログラムに準ずるもの(コンピュータに対する直接の指令ではないがコンピュータの処理を規定する性質を有するデータ等)を含むものとする。

【0160】

また、この形態では、コンピュータ上で所定のプログラムを実行させることにより、ハードウェアエンティティを構成することとしたが、これらの処理内容の少なくとも一部をハードウェア的に実現することとしてもよい。

【0161】

上述の本発明の実施形態の記載は、例証と記載の目的で提示されたものである。網羅的であるという意思はなく、開示された厳密な形式に発明を限定する意思もない。変形やバリエーションは上述の教示から可能である。実施形態は、本発明の原理の最も良い例証を提供するために、そして、この分野の当業者が、熟考された実際の使用に適するように本発明を色々な実施形態で、また、色々な変形を付加して利用できるようにするために、選ばれて表現されたものである。すべてのそのような変形やバリエーションは、公正に合法的に公平に与えられる幅にしたがって解釈された添付の請求項によって定められた本発明のスコープ内である。

【図1】

入力 x	二進数表現	符号 (r = 0)	符号 (r = 1)	符号 (r = 2)
0	0000	1	10	100
1	0001	01	11	101
2	0010	001	010	110
3	0011	0001	011	111
4	0100	00001	0010	0100
5	0101	000001	0011	0101
6	0110	0000001	00010	0110
7	0111	00000001	00011	0111
8	1000	000000001	000010	00100
9	1001	0000000001	000011	00101
:	:	:	:	:

図1

【図2】

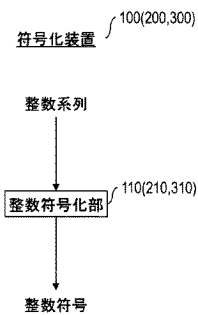


図2

【図3】

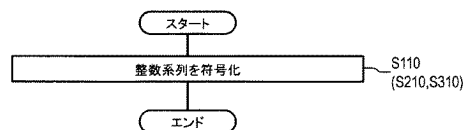


図3

【 図 4 】

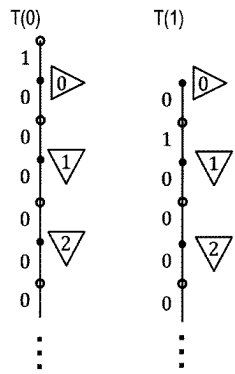


図4

【 図 5 】

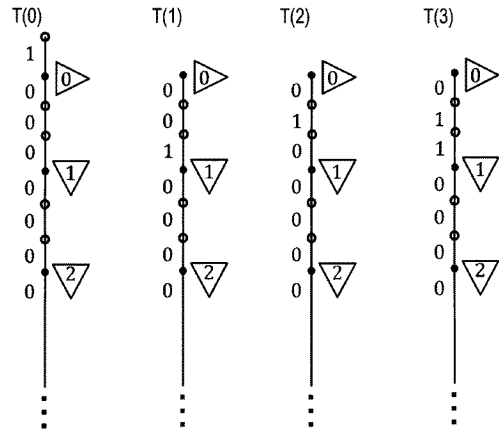


図5

【 図 6 】

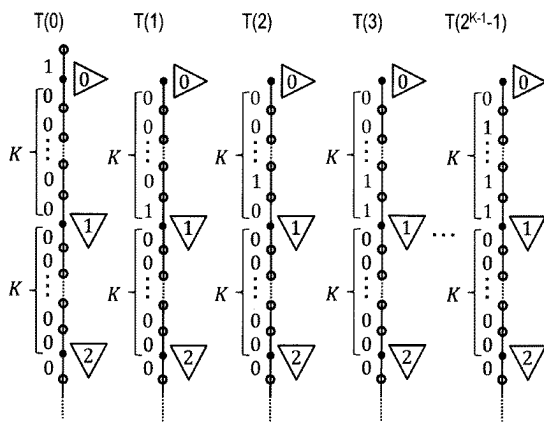


図6

【 図 7 】

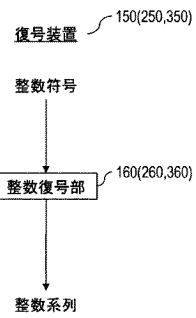


図7

【 図 8 】

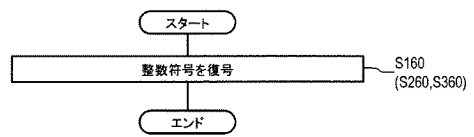


図8

【図9A】

整数値の部分系列	符号
1	01
2	01 00
3	01 00 00
4	01 00 00 00
:	:

図9A

【図10A】

整数値の部分系列	符号
1	001
2	001 000
3	001 000 000
4	001 000 000 000
:	:

図10A

【図9B】

整数値の部分系列	符号
00	1
01	1 00
02	1 00 00
03	1 00 00 00
04	1 00 00 00 00
:	:

図9B

【図10B】

整数値の部分系列	符号
01	010
02	010 000
03	010 000 000
04	010 000 000 000
:	:

図10B

【図10C】

整数値の部分系列	符号
001	011
002	011 000
003	011 000 000
004	011 000 000 000
:	:

図10C

【図11】

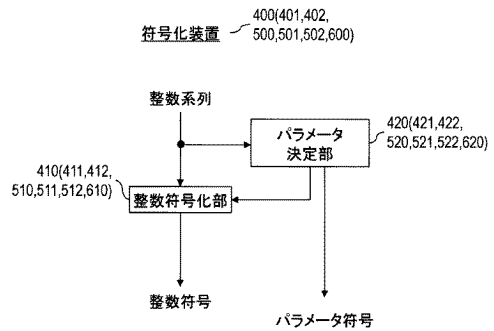


図11

【図10D】

整数値の部分系列	符号
0000	1
0001	1 000
0002	1 000 000
0003	1 000 000 000
0004	1 000 000 000 000
:	:

図10D

【図12】

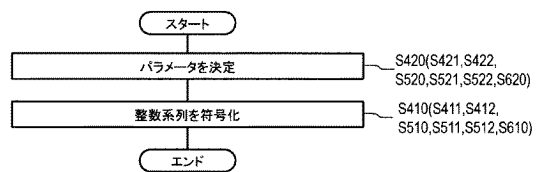


図12

【 図 1 3 】

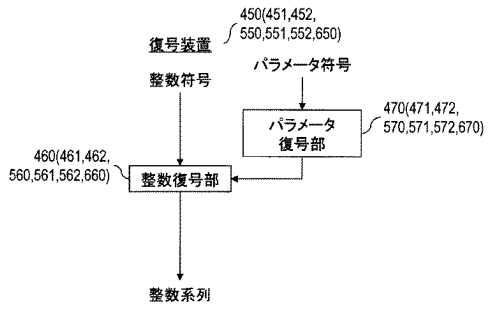


図 1 3

【 図 1 4 】

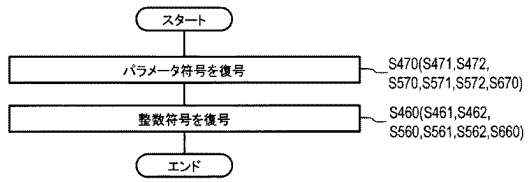


図 1 4

フロントページの続き

(72)発明者 守谷 健弘

東京都千代田区大手町一丁目5番1号 日本電信電話株式会社内

(72)発明者 原田 登

東京都千代田区大手町一丁目5番1号 日本電信電話株式会社内

審査官 北村 智彦

(56)参考文献 特開2013-110763(JP,A)

国際公開第2008/059752(WO,A1)

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, 2007年 9月 1日, Vol.61, No.9, pp.95-100

Ryosuke Sugiura et al., Optimal Golomb Rice Code Extension for Lossless Coding of Low
Entropy Exponentially Distributed Sources, IEEE TRANSACTIONS ON INFORMATION THEORY, 2
018年 4月, Vol.64, No.4, pp.3153-3161

(58)調査した分野(Int.Cl., DB名)

H03M 3/00 - 9/00

G10L 19/00

G10L 19/002

IEEE Explore