

H.223 多重化コード誤り訂正方式

阿南 泰三[†] 上戸 貴文[‡] 伊藤 隆[†] 松田 喜一[†]

[†] 株式会社富士通研究所 〒211-8588 川崎市中原区上小田中 4-1-1

[‡] 富士通九州デジタル・テクノロジー株式会社 〒812-0011 福岡市博多区博多駅前 3-22-8

E-mail: [†] {anan, itoh.t, matsuda.kiichi}@jp.fujitsu.com, [‡] kamito@qdt.fujitsu.com

あらまし 既に第三世代移動通信サービスが始まり、高速な回線を利用した ITU-T H.324 をベースにしたモバイルビデオフォンがサービスの目玉となっている。このような無線モバイルシステムおよび端末では、フェージングなどによる干渉を受け、エラーを生じたデータが相手端末で受信されてしまう問題がある。本論文では、3G324M 内の多重化プロトコルで用いられる ITU-T H.223 の多重化コードに生じるエラーに焦点をあてる。状態遷移モデルと統計的な手法によって送信端末の多重化方法を推定し、多重化コードに生じたエラーを訂正する新しい方法について述べる。

キーワード ビデオフォン, H.324, 3G324M, H.223, エラー訂正, エラー耐性

A New Error Correction Method of H.223 Multiplexing Codes

Taizo Anan[†] Takafumi Kamito[‡] Takashi Itoh[†], and Kiichi Matsuda[†]

[†] Fujitsu Laboratories Ltd. 1-1 Kamikodanaka 4-chome, Nakahara-ku, Kawasaki 211-8588, Japan

[‡] Fujitsu Kyushu Digital Technology Ltd. 22-8 Hakata-ekimae 3-chome, Hakata-ku, Fukuoka 812-0011, Japan

E-mail: [†] {anan, itoh.t, matsuda.kiichi}@jp.fujitsu.com, [‡] kamito@qdt.fujitsu.com

Abstract This paper proposes a new error correction method of ITU-T H.223 Multiplexing Codes. The H.223 is used to realize audiovisual communication on mobile phones called H.324 Terminal. In mobile channels, it is known that the transmitted data are damaged by the channel errors caused by the phasing problem. This paper focuses on the error correction of the H.223 multiplexing codes, which is very important to the service quality of audiovisual phones. The proposed method models how the sending terminal selects the multiplexing codes. We predict the terminal's behavior by the model and correct the received multiplexing codes.

Keyword Video Phone, H.324, 3G324M, H.223, Error correction, Error resilience

1. まえがき

2001 年から第三世代移動通信サービスが始まり、高速な回線を生かしたビデオフォン端末が新しいサービスの目玉として商品化されている。また、ビデオフォンからのアクセスを前提にした様々なサービスの実証実験が行われており、ビデオフォン端末の広がりを見込める。しかしこのような携帯端末は複数の基地局から送信される電波を移動しながら受信するため、受信電波の相互干渉が発生し、受信電力が時間的に変動するフェージングとよばれる現象が起こることが知られている。無線通信路のエラー発生率は有線通信路のそれと比較して 1000~100 万倍にもなる。画像データに生じるエラーは、映像が崩れたり、周囲と色の異なるブロックが発生したりと非常に目立つので、ビデオフォン端末では通信路エラーに対する何らかの対策が

必須となる。

ビデオフォンの通信方式は ITU-T の H.324[1] をベースにしたものが用いられている。図 1 に示すように H.324 では H.223[2] と呼ばれる多重化部が画像や音声コーデックおよび H.245[7] とよばれる端末制御部、H.245 データの再送制御を行う CCSRL/NSRP 部[1] と通信路を結んでいる。エラー訂正技術は図 1 の各部分で行われるが、H.223 多重化部は通信データの入り口となっているので、ここでのエラー訂正技術はその後続くエラー訂正の成否と画像・音声品質を大きく左右する。

H.223 のエラー訂正技術の適用箇所は大きく分けて三つある。一つは通信路上で H.223 パケットの切れ目を示す同期フラグの検出部分、二つ目は多重化コードなど重要な情報が格納されるヘッダ部分、三つ目はパケット中の AV データなどの多重化されたデータその

ものである。また、H.223 は誤り耐性の強度に応じた Annex を選択することが可能となっている。Annex は Annex.A~Annex.D がある ([2]~[6])。

それぞれの特徴をまとめると、以下のようになる。

- [1] ベースライン (Annex なし)
 - ・ 多重化データ (ヘッダ以外の AV データ) の誤り検出
 - ・ 多重化コードの誤り検出
- [2] Annex.A
 - ベースラインのエラー耐性に加えて
 - ・ 同期フラグの検出機能向上
- [3] Annex.B
 - Annex.A のエラー耐性に加えて
 - ・ パケット長を示す情報をヘッダに追加
 - ・ 多重化ヘッダ (多重化コードを含む) の誤り訂正機能
- [4] Annex.C
 - Annex.B のエラー耐性に加えて
 - ・ 多重化データ (ヘッダ以外の AV データ) の誤り訂正機能 (RCPC 符号)
- [5] Annex.D
 - Annex.B のエラー耐性に加えて
 - ・ 多重化データ (ヘッダ以外の AV データ) の誤り訂正機能 (リードソロモン符号)

エラー耐性の最も強い Annex.C や D は、多重化パケット内の AV データそのものに誤り訂正機能を盛り込むため、非常に計算量が多い。このため、携帯端末のようにプラットフォームの処理能力が低く、消費電力が重視される場合には採用することは難しい。現実的な計算コストを考えると適用可能なのは Annex.B までである。Annex.B は AV データの誤り訂正機能がサポートされないため、H.223 プロトコル部以外での誤り訂正や隠蔽と併用して使用するのが望ましい。

以上の理由から、携帯端末で最も現実的である Annex.B について焦点をあて、多重化パケットヘッダに対する誤り訂正について検討を行う。

H.223 のヘッダ情報で破壊されると最も AV 品質に影響を及ぼすのが多重化コードである。多重化コードはオーディオやビデオデータのように複数種類のデータを一つのパケットにまとめる方法を示した 0~15 までの番号を付与されたものである。多重化コードと多重化方法を記述したテーブルは通信開始時のネゴシエーション中に端末間で交換され、相手端末の多重化方法を知ることができる (図 2)。多重化コード 0 番はネゴシエーション用に予め多重化方法が決まっている。このようなしくみのため、受信したデータの多重化コードが通信路エラーによって破壊されると、受信側で

はデータの分離が事実上不可能となる。この結果受信映像が乱れたり、音声途切れたりする現象が生じる。Annex.B は誤り訂正方式を採用しているのですがエラーからの回復が可能な場合もあるが、Annex.B のヘッダは 24bit 長であるのに対して、エラー訂正が可能なのは 3bit でしかない。またエラー訂正方式の特性から、4bit 以上の誤りがヘッダに生じると訂正は不可能で、この場合に訂正を行うと誤ったデータに誤訂正されてしまう。更に、ヘッダ 24bit のうち、多重化コードは 4bit であるので、多重化コードに誤りが生じていなくとも他のヘッダ情報に誤りが 4bit 以上生じていれば、訂正処理後の多重化コードは誤ったものになる。このため、誤り訂正後の多重化コードが常に正しいとは限らず、逆に誤り訂正をしたがために多重化コードを誤ってしまう可能性がある。このような問題を解決するため、本論文では多重化コードを全く別の手段を用いて推定する方法について考察する。

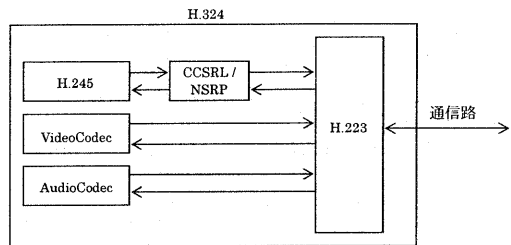


図 1 H.324 の構成図

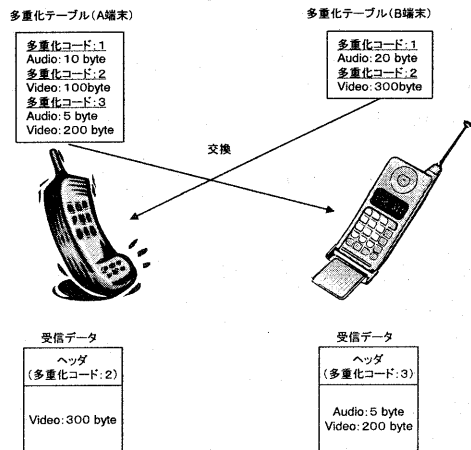


図 2 多重化コードとテーブルの交換

2. 多重化コード推定の概要

2.1. 多重化コードの推定の方針

実際に H.324 規格に適合する擬似 H.324 通信ソフトウェアを作成し、通信の過程で相手端末に送られる多重化コードを時系列で並べてみると、以下ようになった。

0, 0, 2, 3, 2, 1, 2, 2, 3, 2, 3, 3, ...

ここでは隣り合う多重化コードには相関が見られず、多重化コードの時間的推移をモデル化するのは困難である。そこで、H.223 や H.324 のプロトコル上の性質を利用して送信端末の多重化コード決定プロセスをモデル化し、これに基づいて多重化コードを推定する方針をとることにした。

2.2. 多重化コード決定プロセスのモデル化

多重化コード決定に関わる H.324 の状態を以下の三つとして定義する (図 3)。

I. 初期状態 (S_0)

II. オーディオ送信状態 (S_A)

III. その他のデータ送信状態 (S_V)

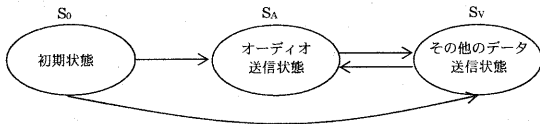


図 3 多重化コード決定プロセス

ここで、“初期状態”とは多重化テーブルを通信相手と交換する前の状態で、この時点で用いられる多重化コードは 0 のみである。また、“オーディオ送信状態”とは、オーディオデータを含む多重化パケットを生成するために使用する多重化コードを選択する状態である。また、“その他のデータ送信状態”とはオーディオを含まない多重化パケットの送信タイミングをあらわしている。ビデオフォンにおいてはビデオや端末制御データの送信タイミングがそれに該当する。初期状態から他状態への遷移は、端末ネゴシエーションが完了した時に行われる。オーディオ送信状態への遷移は、オーディオを送信するタイミングが来たときであり、その他のデータ送信状態への遷移は、オーディオ送信状態でオーディオを実際に送信した後で、次のオーディオ送信までの時間である。

多重化コード決定プロセスを三つの状態にわけると、各状態で選択される多重化コードには相関性が見えてくる。先ほどの例を挙げれば、以下のように分類が可能となる。

多重化コード (受信順)

0, 0, 2, 3, 2, 1, 2, 2, 3, 2, 3, 3, ...

初期状態:

0, 0, ...

オーディオ送信状態

3, 1, 3, 3, 3, ...

その他のデータ送信状態

2, 2, 2, 2, 2, ...

以上のように、多重化コード決定プロセスの各状態において、多重化コードは直前の多重化コードと非常に強い相関があることがわかる。たとえばオーディオ送信状態においては、無音期間と有音期間はそれぞれある程度の時間だけ継続するので、多重化コードもある一定期間同じパターンが繰り返される傾向が強い。その他のデータ送信状態ではオーディオデータが多重化されることはなく、ビデオデータのみが送信されるため同じパターンが繰り返される。つまり受信データを観察して送信端末が多重化コード決定プロセスの中でどの状態にあるかを推定すれば、非常に高確率で多重化コードを推定することが可能となる。

本論文で提案する多重化コード推定のプロセスは以下の手順で行われる。

- ① 多重化コード決定プロセスの状態を推定するためのパラメータ推定
- ② ①で推定したパラメータを用いた状態の推定
- ③ 多重化コードの推定

次節で各々について議論していく。

3. 状態の推定

3.1. 定義

まず最初にいくつかの定義を行う。

[1] 多重化コード

MC: 多重化コードをあらわす。ただし、 $0 \leq MC \leq 15$ とする。

[2] 状態

初期状態: S_0 とする。

オーディオ送信状態: S_A とする。

その他のデータ送信状態: S_V とする。

[3] オーディオ挿入タイミング

たとえば、音声コーデックが AMR であれば、オーディオデータの挿入は 20msec 毎になる。これを、

$$\{t_A(0), t_A(1), t_A(2), \dots, t_A(n), \dots\} \quad (1)$$

とおく。

以上の定義の下、受信データから送信端末の多重化決定プロセスの状態の推定を行う。

3.2. 状態推定の基本概念

図 3中の初期状態 S_0 については、通信開始から端末間のネゴシエーションが終了するまでの期間であるので容易に推定することが可能である。ここではオーディオ送信状態 S_A とその他のデータ送信状態 S_V のいずれであるかを推定する手段について述べる。

受信データの総受信サイズから時刻を計算するのは容易である。送信側で最初にオーディオを送信した時点から 20msec 毎にオーディオデータが送信される。今、回線速度が 64 kbps ならば 20msec は受信データ上で 160 バイトの期間となる。つまり、受信側では最初にオーディオデータを受信してから 160 バイト毎にオーディオデータが挿入されていることになる。オーディオジッタを厳密に 0 に守って送信する端末においては、受信側で 160 バイト毎にオーディオデータを受け取ることになるので、最初のオーディオパケットを受け取ってから $160 \times n$ (n は正整数) バイトの受信データを受け取ったときには、状態はオーディオ送信状態 S_A であり、それ以外では、その他のデータ送信状態 S_V である。しかし、H.324 ではオーディオ送信タイミングのジッタが許されており、実際にオーディオが挿入されるタイミングは $160 \times n$ を中心に揺らぐことになる。そこで、状態の推定はジッタが 0 の場合にオーディオが挿入されるタイミングを中心に揺らぎを許容して、どれだけの期間はオーディオ挿入タイミングであるかを受信データから学習し、その期間内ではオーディオ送信状態である確率が高く、それ以外の期間ではその他のデータ送信状態である確率が高いと判定する。

3.3. 状態推定の方法

あるオーディオ挿入タイミング $t_A(n)$ を中心にして受信したデータから状態のヒストグラムを作ると図 4 のようになると予想される。このヒストグラムからオーディオ挿入タイミング $t_A(n)$ を中心として、オーディオ送信状態である期間 $IA(n)$ を求めれば、その他のデータ送信状態である期間は $IA(n)$ の外側にあると想定してよい。 $IA(n)$ を閾値 k を導入して以下のように定義する。

$$IA(n) = [t_A(n) - k, t_A(n) + k] \quad (2)$$

また、ある時刻 t に対して、時刻 t とオーディオ挿入

タイミングとの距離を次式で定義する。

$$d(t) = \min |t_A(n) - t, n = 0, 1, 2, \dots| \quad (3)$$

このとき、状態の推定は以下で判定される。

$$\text{状態} = \begin{cases} d(t) \leq k \Rightarrow t \text{ はオーディオ送信状態 } (S_A) \\ d(t) > k \Rightarrow t \text{ はその他のデータ送信状態 } (S_V) \end{cases} \quad (4)$$

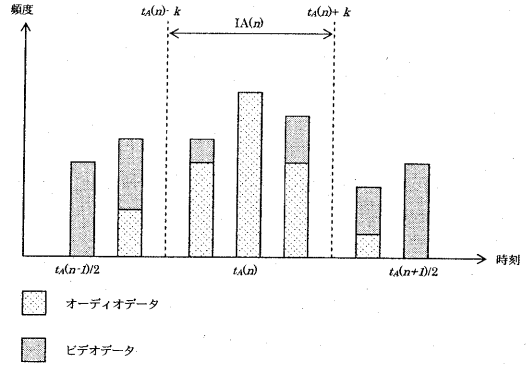


図 4 多重化コードのサンプリング

3.4. オーディオ送信状態期間の推定

式(4)から状態の推定方法が明らかになった。次の問題は閾値 k の決定方法である。閾値は学習データ（通信中に多重化コードの誤り検出を行い、誤りのないデータのみをサンプルしたもの）を用いる。学習データに対して推定値の正解率が最も高くなるように閾値 k を決定する。

定義 1 :

Audio 挿入タイミング t_A からの距離が d でオーディオデータが挿入された回数を $n_A(d)$ とする。

定義 2 :

定義 1 と同様に Audio 挿入タイミング t_A からの距離が d でその他のデータが挿入された回数を $n_V(d)$ とする。

定義 1 から、オーディオ挿入タイミング t_A からの距離が d で、オーディオデータが挿入される確率 $p_A(d)$ は、次式で与えられる。

$$p_A(d) = \frac{n_A(d)}{\sum_{d=0}^T n_A(d)} \quad (5)$$

T はジッタ 0 である場合のオーディオデータ挿入タイミング間隔を 1/2 したものである。例えば、オーディオのパケットの送信間隔が 20msec であれば、 $T=20/2$

msec となる。

同様に、オーディオ挿入タイミング t_A からの距離が d で、その他のデータが挿入される確率 $p_V(d)$ は次式で与えられる。

$$p_V(d) = \frac{n_V(d)}{\sum_{d=0}^T n_V(d)} \quad (6)$$

以上から、距離（閾値）が k 以下のとき、その時刻に挿入されたパケットがオーディオデータを含む確率は、次式で定義される。

$$P_A(k) = \sum_{d=0}^k p_A(d) \quad (7)$$

同様に、距離（閾値）が k 以上のとき、その時刻に挿入されたパケットがその他のデータである確率は、次式で定義される。

$$P_V(k) = \sum_{d=k+1}^T p_V(d) \quad (8)$$

最適な閾値 k は距離が k 以下に含まれるオーディオデータと k 以上に含まれるその他のデータがそれぞれ最も多く含まれるように決定する。例えば、 k を増やすと式(7)の確率は増大するが、逆に式(8)の確率は小さくなる。これとは反対に k の値を小さくすれば、式(8)の確率は大きくなるが、式(7)の確率は小さくなる。最適な閾値 k は式(7)の確率と式(8)の確率を両方最大にするものである。閾値 k は次式で求めることができる。

$$k = \arg \max_l P_A(l) \times P_V(l) \quad (9)$$

閾値 k の実際の学習は通信中に行うことができる。通常 1 秒間にオーディオデータを 50 個、その他のデータを 50 個～150 個ほど受信するので、非常に短時間で学習を行うことができる。

4. 多重化コードの推定

各状態で受信される多重化コードは非常に相関が高い。たとえば、オーディオデータに関する多重化コードは、①オーディオのみ多重化、② {オーディオ無音、ビデオデータ} の多重化、③ {オーディオ最大長、ビデオデータ} の多重化；のほぼ三種類しか用いられない。また、その他のデータ送信状態において用いられる多重化コードは①ビデオデータのみ、②端末制御データのみ；のほとんど二つしか存在しない。また端末制御データは数秒に一度しか送信されないため、その他のデータ送信タイミングにおいては、ほぼ 100% 近くビデオデータのみを多重化する多重化コードが使

表 1 多重化コード統計表

状態	1 時刻前の多重化コード	現在の多重化コード
オーディオ送信状態	1	3
	3	1
	4	4
	5	5
その他のデータ送信状態	2	2

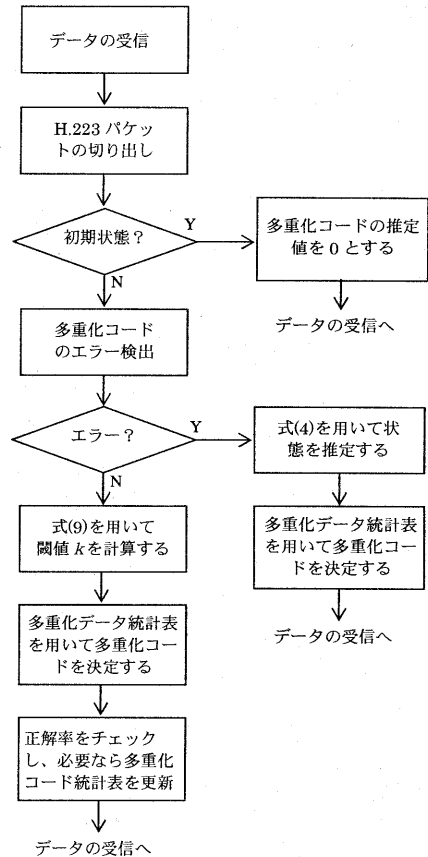


図 5 提案手法のフローチャート

用される。そこで多重化コードの推定は、最初に状態の推定を行い、次に同一状態の 1 時刻前の多重化コードに最も相関の高い多重化コードを推定値として出力することによって行う。これは 1 時刻前の多重化コードと次に送信される多重化コードの頻度統計を計算しておくことで達成される。こうして作成されたのが表 1 である。また、誤りのない多重化コードを受信したら、多重化コードの推定を行い、推定前と推定後の多重化コードを比較することによって正解率をチェック