信号の誤り訂正技術

井上 徹^{*}

1. ま え が き

誤り訂正技術は昨今では、デジタルオーディオ、移動体通信、コンピュータ、マルチメディア周辺機器など広範囲に使われており、データの伝送、媒体の記録再生あるいは蓄積が行われるところでは、もはや必須の技術となってきた。

放送関係ではこれまで衛星音声放送ほか、FM多重放送など低ビットレートのデータにデジタル変調を施し、その後、誤り訂正符号が用いられてきた。最近、地上デジタルテレビ放送さらにはHDTVデジタル放送の検討が進められており、また、報道番組の制作に民間の衛星通信を用いるSatellite News Gathering (以下、SNGと略記)のサービスも開始されている。

デジタル放送の特長として、ある一定のレベル以上の受信状態が確保されれば、誤り訂正技術によって送信側と同一レベルの画質、音質が得られることが挙げられる。一方、衛星通信の分野では符号化率 $R=1/2$ 、拘束長7の畳込み符号が使われており、インテルサット (INTELSAT, International Telecommunication Satellite Organization; 国際電気通信衛星機構)によって仕様化されたIBS (INTELSAT Business Service)、システムとしてはVSAT (Very Small Aperture Terminal)がある。このインテルサット仕様IESS-308には、畳込み符号化、ビタビ復号とリードソロモン符号を組合せた接続符号を備えることが、オプションで仕様として追加されている。

接続符号は主に深宇宙探査などの用途を中心に開発されてきたが、最近ではインテルサットの小型衛星通信システムやニュース取材、番組伝送のためのディジ

図1 接続符号化

タルSNGシステムに接続符号化方式が用いられるようになった。

2. 誤り訂正方式

2.1 接続符号

接続符号は1966年、G. D. Forney, Jr. によって提案された¹⁾。図1にそのモデルを示す。接続符号には外部符号、内部符号に、各々畳込み符号、ブロック符号の組合せがあるので4通りの組合せがあるが、今回紹介するタイプは、外部符号にブロック符号であるRS符号を、内部符号に畳込み符号、ビタビ復号を用いるものである。なお、外部符号、内部符号は通信路より見て内側の符号を内部符号、外側の符号を外部符号と呼ぶ。

接続符号は当初から、もっぱら理論的研究が先行してきたが、その優れた特性が実用上も注目されるようになり、近年、広く衛星通信等の分野で用いられるようになった。なお、G. D. Forneyは接続符号の復号に複数回消失・誤り訂正を使用する一般化最小距離復号なるアルゴリズムを提案している。それによると、ある条件のもとで2元符号をガウシアン通信路に適用した場合、外部符号を、①硬判定（イレージャを用いない）復号した場合、②軟判定（イレージャと誤りを訂正）復号した場合、③一般化最小距離復号をした場合に復号後の誤り率

P_r (ndc: not decoding correctly) が

$$P_r \text{ (ndc ①)} \leq e^{-0.5d \cdot s}$$

$$P_r \text{ (ndc ②)} \leq e^{-0.686d \cdot s}$$

$$P_r \text{ (ndc ③)} \leq e^{-1.0d \cdot s}$$

[†] 三菱電機株式会社 映像システム研究所

“Fundamental Technologies for Digital Television Broadcasting; (5) Error Correction Techniques for Digital Signals” by Tohru Inoue (Imaging Systems Laboratory, Mitsubishi Electric Corp., Kyoto)

図2 HDTV 衛星伝送システム

で与えられることを示している (ただし d は最小距離, s は S/N). すなわち, 軟判定復号, 一般化最小距離復号は, 硬判定復号より S/N で各々 1.4 dB, 3.0 dB 利得が得られることが示されている. なお, 畳込み符号は元来最尤復号を基本とするアルゴリズムがあり, 有効な復号法が検討されている²⁾.

2.2 接続符号の実用例

(1) デジタル SNG

平成元年の民間衛星通信の開始以来, 放送局等で使用されている SNG システムは, ニュースおよび番組素材の伝送において, その有効性が認識されつつある. 特に離島や遠隔地からの伝送には必須の技術になってきている. しかしながら, アナログの SNG システムでは放送品質の十分な確保が難しい. また, デジタル化は時代の流れともなっており, デジタル SNG もすでに実用化に入る段階にある. アナログ SNG からデジタル SNG への移行期間は, アナログ, デジタルの両方の SNG が混在するため, デジタルの共存条件での伝送実験も行われており, アナログ・デジタル間の相互干渉, 妨害の問題も検討されつつある. また, 誤り訂正方式は接続符号化方式を採用し, 内部符号に低 C/N で復号可能な畳込み符号化/ビタビ復号を用い, 外部符号にはビタビ復号によるバースト誤りを訂正する RS (リードソロモン) 符号を採用している^{3)~5)}.

復号側では, ビタビ (Viterbi) 復号の出力側で訂正しきれない誤りが誤り伝搬を起こし, それがバースト誤りとなるので, インタリーブを施して外部符号 (あるいは外符号) である RS 符号が訂正できる程度のランダム誤りに変換する必要がある. 長いインタリーブ

変調 : QPSK
RS 符号 : (255, 239) 8 シンボル訂正 RS 符号
畳込み符号 : 3/4 畳込み符号化/軟判定ビタビ復号
($K=7$, バンクチュー方式, パスメモリー長 35)

図3 インタリーブ段数によるシステムのパフォーマンスの変化

はバースト誤りのランダム化には効果があるが, 遅延時間とハードウェア量の点からは不利となる. 理想的なインタリーブの深さはビタビ復号器のバースト誤り訂正の特性に依存する. インタリーブの段数 l が 4 のとき最適値が得られることが, 理論とシミュレーションから報告されている⁴⁾⁶⁾.

(2) HDTV 信号の衛星伝送

衛星を介して, HDTV のスタジオ規格信号を符号化しデジタル伝送を行うものとして, 14/12 GHz の Ku バンドのトランスポンダで 1 チャンネル 36 MHz の帯域幅を使い, QPSK 変調にて伝送するシステムが開発されている⁷⁾⁸⁾. 誤り訂正としては 3/4 畳込み符号と (255, 239) RS (Reed-Solomon) 符号による接続符号化を用いる. データは 8 バイト訂正の RS 符号に符号化され, インタリーブを施された後,

3/4 畳込み符号化されて伝送される．図 2 にシステム全体図を示す．インタリーブ段数は (1) 項で述べた SNG と同じく 4 段が最適であった．そのパフォーマンスを図 3 に示す．

3. 接続符号のハードウェア

以下，SNG など で実用化されている接続符号のハードウェア構成を述べる．したがって，個々の符号を構成している RS 符号器/復号器と畳込み符号器/ビット復号器の構成を解説する．

RS 符号は，演算を定義する体とシンボルを表現する体が同一のガロア体で定義されている符号である．すなわち，GF (2⁸) の体で定義されていたらシンボルも GF (2⁸) で表現され，したがって誤りシンボルもガロア体の元 GF (2⁸) として表現される．符号長 n ，情報記号数 k とするとき，最小距離 d は $d=n-k+1$ で与えられる．符号の最小距離は $d \leq n-k+1$ であるので，検査シンボル数に対して最も訂正能力が高いという特徴がある．符号パラメータは

符号長 $n=2^m-1$

情報記号数 $k=n-(d-1)$

検査記号数 $n-k=d-1$

と与えられる．

ガロア体および RS 符号の数学的構造および復号アルゴリズムなどは符号理論の成書⁹⁾を参照されたい．また，RS 符号，畳込み符号ともに線形符号であるので，符号化は行列演算でできる．また，多項式除算による方法も比較的容易に実現できるので省略する．

3.1 RS 復号器¹⁰⁾

RS 復号器をどう実現するかは，RS 符号の訂正能力，許容復号遅延時間，ハードウェアコストなどにより様々な回路構成が考えられている．RS 復号アルゴリズムの手順は回路構成に共通に以下の手順を踏む．

- ① シンドロームを計算する．
- ② シンドロームから誤り位置多項式，誤り数値多項式を計算する．
- ③ 誤り位置多項式を解いて誤り位置を計算する．
- ④ 誤り位置多項式，誤り数値多項式，誤り位置の情報から誤り数値を計算する．
- ⑤ 誤りを訂正する．

RS 符号ではバイト単位の訂正を行うため，誤り位置とともに誤り数値を計算しなければならない．最小距離の大きい RS 符号の復号は②の演算で繰り返し演算を用いる復号アルゴリズムを適用することが多い．ここではユークリッド復号法⁹⁾¹⁰⁾を想定して解説を進める．また，RS 復号器はオンザフライ (on-the-fly)

図 4 RS 復号器の構成¹⁰⁾

復号 (データを連続的に訂正し，流れを止めずに出力する) を要求されることが多く，セミカスタム LSI による専用 LSI を開発することが多い．

(1) RS 復号器の構成

ユークリッド復号を行う RS 復号器の構成は，図 4 に示すように 7 つの回路部分に分かれる．

- ① データ遅延メモリ
- ② シンドローム計算回路
- ③ ガロア体上の多項式演算回路
- ④ チェンサーチ回路
- ⑤ 誤り数値計算回路
- ⑥ 誤り訂正回路
- ⑦ 制御回路

これらは明確に分かれるものではなく，例えばハードウェア削減のため回路の共用化が検討されている．例えば，

I. ガロア体多項式演算回路，誤り数値演算回路，誤り訂正回路の共用化

II. シンドローム計算回路，チェンサーチ回路の共用化

が検討されている．特にガロア体演算回路 (GLU : Galois Logic Unit) を用いる回路構成では I の共用化を用いることが多い．

(2) シンドローム計算回路

RS 符号の復号器では受信語から，まずシンドロームの計算を行う．ガロア体上の RS 符号では図 5 に示すように， m ビットパラレルでデータを情報シンボルの先頭からチェックシンボルまで連続的に入力する．シンドローム S_j ($j=0, 1, \dots, d-2$; d は最小距離) は w を生成多項式の初期値とすると

$$S_j = \sum_{i=0}^{n-1} r_i \cdot \alpha^{i(j+w)} \\ = (((\dots (r_{n-1} \cdot \alpha^{j+w} + r_{n-2}) \alpha^{j+w} \\ + \dots + r_2) \alpha^{j+w} + r_1) \alpha^{j+w} + r_0) \quad (1)$$

と表現され，ガロア体上の加算器とガロア体上の乗算

器で構成できる。回路例を図6に示す。

(3) ガロア体多項式演算回路

RS復号に多く用いられるユークリッド復号法を想定した回路構成例を述べる。

ユークリッドアルゴリズムは、演算がすべてガロア体上の多項式演算に帰着されており、判定条件も多項式の次数で行うので理解しやすい特徴があり、様々な回路構成が提案されている。

ユークリッドアルゴリズムは、以下のガロア体上の多項式演算を実行すればよい。

$$Q(z)=[M_1(z)/M_2(z)]$$

$$R(z)=M_1(z)-Q(z)M_2(z)$$

$$U(z)=Q(z)U_1(z)+U_2(z)$$

[ ] は商多項式を表す。

図7にガロア体上の多項式除・乗算回路例を示す。図7におけるガロア体の乗算回路はANDゲートとEXORの組合せで実現できる。また、除算回路は逆元ROMと乗算回路の組合せで構成できる。この回路および次数判定の回路を用いれば、ユークリッドアル

ゴリズムが実現できる。さらにハードウェアを小さくするためRAMを遅延メモリーに用い、伝送速度より高速のクロックを用いて、演算ステップ数を多く使う代わりにガロア体上の演算回路 (GLU) を1つだけ持ち、繰り返し用いることで、誤り位置多項式、誤り数値多項式、誤り数値計算、誤りの訂正を実現する。

GLUはガロア体上の演算を専用回路で実行することによって演算を高速で実行する。図8にGLUを用いた回路の構成を示す。図8のGLUはガロア体上の多項式演算を考慮して構成され、ガロア体演算部、算術演算部、制御部の3つの部分に分かれる。外部インタフェースは、ガロア体演算部、算術演算部が共通の入出力インタフェースを持つ。

ガロア体演算部

GLUと1次記憶用RAMからなる。GLUはガロア体上多項式除算および乗算が、ガロア体の積和演算から構成されていることから、この2つの機能を有している。X、Y、Zのレジスタに対し

$$X \times Y + Z$$

$$X/Y + Z$$

の演算を使い分ける。誤り数値計算では以下の演算を実行する。

$$X^2 \times Y + Z$$

$$X^w + Z$$

除算は逆元回路と乗算回路に分けて構成することで乗算回路を積和演算と共用化できる。除算でZの加算が不要であるが、0をZに入力する。数値計算で用いる $X^w + Z$ のwは、生成多項式によって決まる。記憶用のRAM A、B、Cは、シンドロームから誤り位置多項式と誤り数値を求めるまでの中間的な数値を

図5 RS受信語系列¹⁰⁾

図6 シンドローム計算回路¹⁰⁾

(1) ガロア体上の多項式除・剰余演算回路例

(2) ガロア体上の多項式乗算回路例

図7 多項式演算回路¹⁰⁾

図8 RS 復号 LSI (多項式演算部)¹⁰⁾

記憶するために用いる。また、多項式演算をパイプライン処理で演算速度を上げるため、3個のRAMが独立にリード/ライトできる構成になっている。

(4) チェンサーチ回路

多項式演算回路により得られた位置多項式は

$$\sigma(z) = \sigma_t \prod_{i \in E} (Z - \alpha_i) \quad (2)$$

である。ここで σ_t は誤り位置多項式の最大次数の係数、 E は誤り位置の集合、 i は誤り位置、 α_i は位置 i に対応するガロア体の元である。誤り位置は(2)式の根を求め、そのガロア体の元に対応する位置を誤り位置と推定する。

誤り位置を求める方法は、一般的にはチェンサーチアルゴリズムを用いる。チェンサーチを行う目的は、誤り位置とその誤り位置に対応する(2)式のガロア体の元を求めることである。この操作は基本的には多項式に値を代入するだけなので、チェンサーチ回路ではGF(2^m)の元が次々に代入され、その結果、この元を誤り位置多項式に代入した値が出力され、したがって、このデータが0かどうかを調べ、根であるかどうかを判定する。チェンサーチ回路はシンドローム計算回路とよく似た回路構成であるので、両者を兼用した回路構成も提案されている。

(5) 誤り数値計算回路

2元のBCH符号と異なりバイト誤りを訂正するRS符号であるため必要な回路である。誤り位置 i の

数値 e_i は(1)式のシンドローム多項式から求めると

$$e_i = \frac{\eta(\alpha^{-i})}{(\alpha^{-i})^{-A+1} \cdot \sigma'(\alpha^{-i})} \quad (3)$$

与えられる。ここで $\sigma'(Z)$ は $\sigma(Z)$ の形式微分、 $\eta(Z)$ は誤り数値多項式で

$$\eta(Z) = \sigma_t \sum_{i \in E} e_i \prod_{\substack{j \in E \\ j \neq i}} (Z - \alpha_j) \quad (4)$$

与えられる。(3)式の A は生成多項式より決まり、例えば α^{120} から始まる生成多項式では $A=120$ である。

GLUによる誤り数値計算

GLUを用いて誤り数値計算を行った場合、 $\sigma'(\alpha^{-i})$ 、 $(\alpha^{-i})^{-A+1}$ 、 $\eta(\alpha^{-i})$ をそれぞれガロア体演算回路を使って計算した後、(3)式全体の演算を行う。また、チェンサーチにおいて検出した誤り位置 i に対して α^{-i} を用意しておく。形式微分 $\sigma'(z)$ は

$$\sigma'(z) = \sum_{i=0}^{(t-1)/2} \sigma_{2i+1} Z^{2i} \quad (5)$$

となり、偶数次の係数のみをもつ。 σ_{2i+1} は、 $\sigma(z)$ の $2i+1$ 次の係数である。(5)式を変形すると $\sigma'(z) = \sigma_1 + Z^2(\sigma_3 + Z^2(\sigma_5 + Z^2(\dots)))$ となる。これはGLUの $X^2 \times Y + Z$ の機能を使い計算する。すなわち、初期値として X に α^{-i} を、 Y に0を、 Z に誤り位置多項式の最大奇数次の係数を入力する。次に $X^2 \times Y + Z$ の計算をして、その結果を Y に入力し、次の奇数次の係数を Z に入力する。これを次々繰り返す。最後に Z に σ_1 が入力され $X^2 \times Y + Z$ が計算されて $\sigma'(\alpha^{-i})$

図9 ポーゼンクラフト型畳込み符号器

が得られる。

3.2 畳込み符号と Viterbi 復号

(1) 畳込み符号

図9はポーゼンクラフト (Wozencraft) 型といわれる畳込み符号の符号器である⁹⁾。情報系列を長さ k_0 ビットのブロックに分割し、各ブロックの k_0 ビットを並列化する。各ビットを m 段に縦続接続されたレジスタおよび線形組合せ回路によって並列直列変換器に入力し、直列化された符号系列を通信路へ出力する。

畳込み符号を規定するパラメータとしては

$$\text{符号化率} \quad r = k_0/n_0$$

$$\text{拘束長 (段数)} \quad K = m + 1$$

である。 r は情報系列のビット数 k_0 と符号系列のビ

図10 畳込み符号器の例

図11 状態遷移図

ット数 n_0 の比率を表し、ブロック符号の情報ビット k 、符号長 n に対応する。畳込み符号はブロックを単位に符号化復号化が行われるが、ブロック符号と違い、畳込み符号では過去のブロックが現在のブロックにいろいろ影響を及ぼす。あるブロックを符号化または復号するのに影響の及ぶ範囲を拘束長 (ビット) といい

$$n_A = n_0(m+1)$$

で表す。拘束長 n_A と符号長 n が同程度の畳込み符号とブロック符号では、訂正能力、ハードウェア規模は、ほぼ同程度になると考えられている。

畳込み符号の誤り訂正能力を高めるには、後述する格子線図上で規定される最小自由距離を大きくすればよい。一般的には r が一定であれば拘束長 K (段数) を大きくすればよい。

(2) ビタビ復号

(a) 硬判定ビタビアルゴリズム 図10の符号器から生成される畳込み符号を考える。状態遷移図は図11のようになる。格子線図は図12のようになる。符号の最小自由距離は格子線図ではすべて0の状態 (S_{00}) から出発し、他の状態を経て再び0の状態に戻るパスのハミング重みの最小値で与えられる。この例では $S_{00} \rightarrow S_{01} \rightarrow S_{10} \rightarrow S_{00}$ のパスのハミング重みが5であるので、最小自由距離は5であることがわかる。

初期状態として0が2個入力され4ビットの情報 (0101) が入力され、その後、2つの0で終結したとする。前後の0は受信側の格子線図を既知の点から出発させ既知の点に終結させるためのものである。いま、図12に示すように情報系列を (010100) とすると符号系列は (001101000111) となり、通信路において誤り

図12 硬判定ビタビ復号

系列 (000100100000) が加算されて (001001100111) なる系列が受信されたとする。送信側と受信側の同期は完全にとれているものとする。ビタビ復号は、このような出発点と終結点を与えられた格子線図上で受信系列に最も近いパスを捜し出す。

硬判定復号とは、受信系列を1個の閾値だけを設け1, 0のパターンとして符号語との“ちかさ”を評価する測度としてハミング距離を用いる。t=3のS₀₀に合流する2つの符号系列(000000)と(110111)と、受信系列(001001)とのハミング距離を調べると、前者で2、後者で5であるので、後者を除去し前者を生き残りパスとする。以下、同様の処理を続け、例えばt=5でS₀₀に合流する2つの符号系列(0011011100)および(1110011011)のように、受信系列に対するハミング距離が等しいときは片方を選択する。最後に終結点に合流する2つの符号系列(001101110000)および(001101000111)と受信系列(001001100111)とのハミング距離を調べると、前者で5、後者で2であるので、後者を最尤の符号系列であると判断する。したがって、この最終的なパスに対応する情報系列(010100)がビタビ復号により推定された。これは通信路の2ビットの誤りを訂正したことになる。

(b) 軟判定ビタビアルゴリズム 受信側の復号器の前にA/D変換器を設けて複数個の閾値を設け、受信系列をアナログの値で復号器に入力する。このアナログ値をメトリックという。図13に3ビット(8値)軟判定のためのメトリック表を示す。

2ビット(4値)、4ビット(16値)軟判定も同様である。硬判定復号と同じように(001101000111)なる符号系列が通信路において雑音の影響を受けて受信されたとする。このとき3ビット軟判定ビタビ復号法を適用した例を図14に示す。

この符号の最小自由距離は5であるから、2ビットのランダム誤りまでは硬判定復号器でも訂正できる。しかし、この例では0レベルを閾値とすると3ビットの誤りとなり、硬判定復号器では誤りが訂正できない。これに対し、3ビット軟判定復号を行えば完全な復号が可能であることがわかった。実際のシステムでは、3ビット軟判定復号器は硬判定の復号器に比べて約2.0 dB 符号化利得がある。メトリックの改良が提案されている¹¹⁾。

(3) ビタビ復号器の構成

ビタビ復号器をハード化するには以下の問

題を解決しなければならない。

- ① パスメモリーの打ち切り
- ② メトリックの量子化
- ③ パスメトリックの正規化

などである。

① パスメモリーの打ち切り これまで復号器は無限の大きさのメモリーを持ち、生き残りパスとそのパスメトリックをデータ系列が伝送し終わるまで記憶していたとした。また、すべてのデータの入力が終わらなければ最尤のパスの推定が不可能なので、復号出力が復号器より出力されるのはこれより後になる。さらに、復号器の格子線図の出発点と終結点を規定するため、本来伝送に必要なダミービットをデータビットの前後に付加しなければならない。実際、復号器の格子線図の状態を一意に決定するには、 $k_0(K-1)$ 個の情報ビットが必要となる。これらのことは物理的に不可能であったり、伝送の効率を低下させるので、原理的なビタビアルゴリズムを修正する必要がある。

復号器の格子線図の出発点は無限の過去に、終結点は無限の未来にあると仮定しよう。現在の復号時刻では $2^{k_0(K-1)}$ 本の生き残りパスが状態数 $2^{k_0(K-1)}$ 個あることになる。この生き残りのパスを過去に遡って行くと、次第に1本のパスに終結していくことになる。無

受信信号レベル	← Low 0 High →							
	000	001	010	011	100	101	110	111
量子化レベル	0	1	2	3	4	5	6	7
符号の0シンボルに対するメトリック	0	1	2	3	4	5	6	7
符号の1シンボルに対するメトリック	7	6	5	4	3	2	1	0

図13 メトリック

図14 軟判定ビタビ復号

限の過去まで遡れば、ある状態に収束するが、実際は有限の過去までしか遡れないから、現在最小のパスメトリックを持つパスの最も古い状態の情報ビットを現在の復号出力とする。このように、メモリーを有限長で打ち切るため生じる誤りはトランケーションエラーと呼ばれる。拘束長の5倍すなわち $5K$ 本程度のブランチを遡れば、トランケーションの影響はチャンネルで生じた誤りと同等であるとされており、パスメモリーとしては1本の生き残りパスが過去の $5K$ 程度の状態を持つようにし、1個当たりの状態が k_0 ビットの情報系列を持てばよいから

$$2^{k_0(K-1)} \cdot 5K \cdot k_0$$

だけの容量のRAMを用意すればよい。実際には k_0 ビット単位で $2^{k_0(K-1)}$ 並列に読出し書込みを行う。このパスメモリートランケーションは畳込み符号に特有な誤り伝搬を打ち切るためにも有効である。

② メトリックの量子化　メトリックとして、硬判定のときはハミング距離を、軟判定のときは正整数を用いてきた。しかし、ビタビアルゴリズムを最尤復号法とするためには、通信路の誤り率を考慮した対数尤度で表現された尤度関数をメトリックとして用いなければならない。しかしながら、様々なシミュレーションによれば、厳密に尤度を計算して求めたメトリックと正整数とを用いた場合の差は殆どないとされている。正整数をメトリックとして用いてよいなら、ハードウェア設計上非常に有利になる。

③ パスメトリックの正規化　復号器の格子線図で $2^{k_0(K-1)}$ 個の状態にそれぞれ 2^{k_0} 本の枝が合流し、 2^{k_0} 本の枝が次の時刻の状態に向かって出力する。正整数のメトリックを用いたとして、先の例に明らかにように、生き残りパスのメトリックは時間に対し単調に増加するので、一定値に抑えるにはパスメトリックの値の正規化が必要である。

- I. 現在のある状態に合流すべき 2^{k_0} 本の枝のすべてについて、そのブランチメトリックを計算する。
- II. このブランチメトリックを1時刻過去のパスメ

トリックにそれぞれ加算し、新たなパスメトリックとする。

- III. この中で最も小さなパスメトリックを持つものをこの状態の生き残りパスとして、そのパスメトリックをこの状態の状態メトリックとする。
- IV. 以上を $2^{k_0(K-1)}$ 個の状態すべてについて行う。
- V. $2^{k_0(K-1)}$ 本の生き残りのうち、最小のパスメトリックを持つものの最も古い情報ビットを現在の復号出力とする。
- VI. $2^{k_0(K-1)}$ 本の生き残りパスのパスメトリックのそれぞれから、上の最小のパスメトリックを減じる。これが新しいパスメトリック、すなわち状態メトリックとなる。

この正規化手法によれば、生き残りのパスメトリックは $(K-1)\xi$ 以下であることが知られている。ここで ξ はブランチメトリックの最大値である。図14の例では $\xi=14$ となるから、生き残りパスメトリックは28以下となる。

(4) ビタビ復号器のハードウェア規模¹¹⁾¹²⁾

図15にハードウェアブロック図を示す。ビタビ復号器は大きく分けて

- ① 枝メトリック計算回路
- ② 加算比較選択回路
- ③ パスメモリー

の3つに分けて考えると便利である。

① 枝メトリック計算回路　 $2^{k_0(K-1)}$ 個の状態それぞれから 2^{k_0} 本のブランチが出力され、次の $2^{k_0(K-1)}$ 個の状態それぞれに入力される。ブランチは 2^{n_0} 種類あるから、計算すべきブランチメトリックは 2^{n_0} 個である。例えば3ビット軟判定の場合、3ビット量子化された1ビットの符号シンボルについて、図16の回路によってシンボル0とシンボル1に対するメトリックを計算する。このシンボルメトリックより図17の加算器でブランチメトリックを計算する。 2^{n_0} 個のブランチメトリックのすべてについて、ブランチに対するメトリックを求めるから、この加算器が 2^{n_0} 個必要となる。

図15 ビタビ復号器の構成¹⁰⁾

図16 シンボルメトリックの計算

図17 ブランチメトリックの計算

② 加算比較回路 1時刻過去の $2^{k_0(K-1)}$ 個の状態メトリックと 2^{n_0} 個のブランチメトリックからすべてのパス(2^{k_0} 本)のメトリックを計算し、最小のメトリックを持つものをこの状態への生き残りパスとして、そのメトリックをこの状態のメトリックとする回路である。加算比較選択回路からはトレリス接続信号が出力され、トレリスの各状態に合流するパスのうち、どちらを選択したかがパスメモリーへ伝送される。

$2^{k_0(K-1)}$ 個の状態のすべてについてこのような演算が必要であるから、この回路が $2^{k_0(K-1)}$ 組必要である。

③ パスメモリー 1本の生き残りパスが過去の $5K$ 個の状態を持つようにし、1個の状態あたり k_0 ビットの情報系列を記憶しておけばよいから

$$2^{k_0(K-1)} \cdot 5K \cdot k_0$$

だけのメモリーを用意すればよい。シフトレジスタは k_0 ビット単位で $2^{k_0(K-1)}$ 段並列に読出し、書込みの行えるものでなければならない。

3. む す び

デジタルテレビ放送の伝送系に用いられる誤り訂正技術について、中心となりつつある接続符号および、その周辺技術としてRS符号、ビタビ復号とその実用例を解説した。誤り訂正技術はガロア体をベースにしているので、それを専門とする技術者以外には馴染みがなく、取っつきにくいのが現状である。本稿で

はかなり直観的な説明で理解が可能のように工夫したつもりであるが、紙面の都合で意をつくせなかったところもあり、不明の点は多数出ている符号理論の成書をひもといてみられることをお勧めする。

ビタビ復号もRS符号も充分枯れた技術ではあるが、放送に適用するにあたり、例えば復号誤りや誤り伝搬の影響、さらにはグレースフルデグラデーションの導入など、蓄積しなければならないノウハウは多いと思われる。

〔参 考 文 献〕

- 1) G. D. Forney, Jr.: "Concatenated Codes", MIT Press (1966)
- 2) A. J. Viterbi and J. K. Omura: "Principles of Digital Communication and Coding", McGraw-Hill, New York (1979)
- 3) 田村信一, 坂口裕直, 加藤正光, 高山亨, 坪池光芳, 坂戸美朝, 磯部清治, 杉山淳一: "ディジタルSNGの衛星伝送実験", テレビ学技報, ITE Technical Report, 17, 62, pp. 1-6 (Oct. 1993)
- 4) 坂戸美朝, 佐々木源, 浅野研一, 浅井光太郎, 海老沢秀明, 吉田英夫: "ディジタルSNGシステム用ビデオコーデック", テレビ学技報, ITE Technical Report, 17, 13, ROFT 93-20 (Feb. 1993)
- 5) 服部伸一, 浅井光太郎, 田中浩一, 坂戸美朝, 浅野研一: "衛星利用放送品質コーデック", 三菱電機技報, 67, 7, pp. 33-38 (1993)
- 6) 三宅 真, 藤野 忠: "インターリーピングの深さを低減した接続符号化方式の理論ビット誤り率特性", 信学技報, IT 89-53, pp. 13-18
- 7) T. Kasezawa, T. Shinohara, T. Nakai, M. Nishida and T. Murakami: "HDTV Digital Transmission through Satellite Channel", International Workshop on HDTV'92, Kawasaki City (Nov. 18-20, 1992)
- 8) T. Nakai, T. Kasezawa, T. Shinohara, M. Nishida and T. Murakami: "Digital HDTV Transmission System via Satellite", SPIE, 1976, pp. 31-41, High-Definition Video (1993)
- 9) 宮川洋, 原島博, 今井秀樹: "情報と符号の理論", 岩波講座情報科学 4, 第8章 (1982)
- 10) 井上徹監修: "実戦誤り訂正技術", 第4章, 第6章, トリケップス (1993)
- 11) 三宅真, 藤野忠, 藤原謙一: "簡単化されたディジタルメトリック計算を行うビタビ復号器の特性", 信学論B, J 70-B, 6, pp. 673-681 (July 1987)
- 12) 三宅真, 村上圭司, 山岸篤弘, 石田弘, 石津文雄, 小島年春, 藤野忠: "低C/N比伝送用接続符号化モデムの設計と開発", 信学技報, SAT 88-24, pp. 19-24

いのうえ とおる
井上 徹 昭和44年、京都大学工学部電気工学科卒業。同年、三菱電機(株)中央研究所に入社。昭和50～52年、米国COMSAT研究所へ国際職員として出向。誤り訂正符号関連の研究開発に従事。現在、同社映像システム開発研究所開発第一部主席研究員。工学博士。