

1030MHz/1090MHz 信号の復号方式の改善

電子航法開発部 古賀 禎

1. はじめに

SSR,SSRモードS、TCASおよびASASは、同一の周波数(1030MHz/1090MHz)を利用している。これらのシステムは、ランダムに信号をアクセスするため、航空交通の増大に伴い、混信による性能の劣化が指摘されている。特に、拡張スキッタによる長距離の空対空監視システムは、非常に多くの干渉信号(40,000個以上/秒のモードA/Cフルーツ)の下で利用されるため、メッセージの復号率が著しく低下することが予想される。これらの問題を解決するため、拡張スキッタの改良受信方式がMITリンカーン研究所で開発された。米国航空無線技術委員会(RTCA)は、Do-260[1]にて、改良受信方式を推奨している。改良受信方式により復号率が改善されたものの、より優れた受信方式の開発が求められている。

本報告では、改良受信方式の1つ、Brute Force方式を改善する方法を提案する。第2節にて受信処理およびBruteForce方式の受信技術について述べた後、第3節にてBrute Force方式の改善法の詳細について述べる。

2. 受信技術

2.1 受信処理

受信機は、3つの受信処理(プリアンブル検出処理、ビット判定処理、誤り検出訂正処理)を行い、復号メッセージを得る。

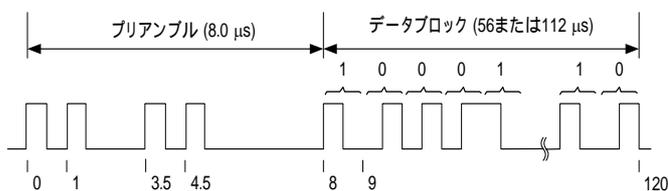


図2 1090MHz 信号形式

図1に受信処理の構成を示す。プリアンブル検出処理では、1090MHz信号(図2)のプリアンブルを検出し信号の有無を判定する。ビット判定処理では、データブロックに含まれるビット値を判定し、2つのビット列(決定ビット列と信頼性ビット列)を生成する。決定ビット列はデータのビット値を示し、信頼性ビット列は対応する決定ビットの信頼性を示す。低信頼性ビットが1の場合、対応する決定ビットの信頼性は低く、0の場合、決定ビットの信頼性が高いことを意味する。

誤り訂正処理では、決定ビット列と信頼性ビット列を用いて訂正を行う。まず、Conservative方式と呼ばれる方式を実施する。この方式は、連続する24ビット内に集中して生じた低信頼性ビットの誤りを訂正する。Conservative方式で訂正できた場合、誤り訂正処理は終了する。訂正できない場合、BruteForce方式を実施する。BruteForce方式は、受信メッセージに生じた5ビット以下(誤りの存在範囲に制限はない)の低信頼性ビット誤りを訂正する。

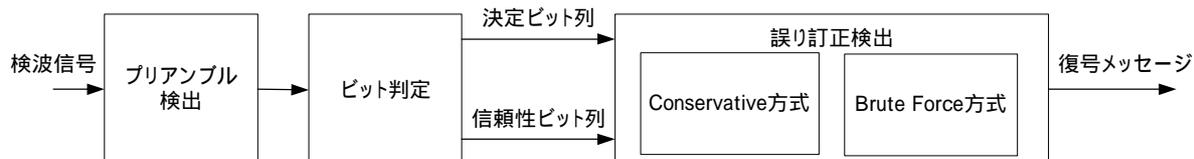


図1 受信処理の構成

送信 送信ビット列 S 0000000
 受信 決定ビット列 D 0000011
 信頼性ビット列 C 0100011

メッセージ長 7 ビット
 生成多項式 $g(x)=x^3+x+1$

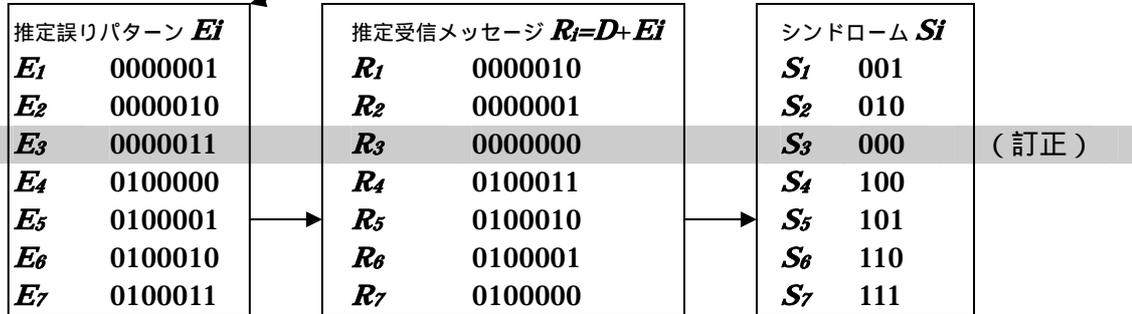


図 3 Brute Force 方式による誤り訂正

2.2 Brute Force 方式

本節では、Brute Force 方式による誤り訂正の手順、見逃し誤りについて述べる

2.2.1 訂正手順

Brute Force 方式では、誤りは低信頼性と判定された決定ビットに生じると仮定し、訂正を行う。以下に手順を示す。

低信頼性ビット列 C の 1 であるビットのいずれかを反転させ、推定誤りパターン列 E_i を生成する。

E_i と決定ビット列 D を排他的論理和 (Ex-OR) し、推定受信メッセージ R_i を作成する。

R_i からシンδροーム S_i を計算する。

$S_i = 0$ の場合、 E_i がメッセージに重畳した誤りパターンと判定し、訂正が完了する。

$S_i \neq 0$ の場合、 C の 1 であるビットを反転させ、次の推定誤りパターン列 E_2 を生成する。～ のパターンを繰り返す。

$2^t - 1$ 個の推定誤りパターン (t は低信頼性ビットの数) に対して ～ を実施して、 $S_i = 0$ とする推定誤りパターンがない場合、訂正不能とし終了する。

例として、送信ビット列 $S=(0000000)$ を送信し、決定ビット列 $D=(0000011)$ 、信頼性ビット列 $C=(0100011)$ を受信した場合を考える。図 3 に、訂正の過程を示す。第 2,6,7 ビット目の信

頼性が低いので、これらビットを反転させて、推定誤りパターン E_i ($i=1,2,\dots,7$) を作成する。次に、推定送信メッセージ $R_i=E_i+D$ を作成し、シンδροーム S_i を計算する。例では、 R_3 の時シンδροームが 0 となることから、 E_3 が発生誤りパターン、 R_3 が送信メッセージであると判定し、誤り訂正を完了する。

2.2.2 見逃し誤り

見逃し誤りとは、送信ビット列が復号後にそれとは異なる送信ビット列となる誤りである。復号メッセージが送信系列の一つとなるため、誤り訂正処理は、誤りであるにもかかわらず正しいと判定する。例えば、送信ビット列 $S=(0011101)$ を送信し、決定ビット列 $D=(0000001)$ 、信頼性ビット列 $C=(0011101)$ を受信した場合を考える。低信頼性のビットが 4 個であるから、15 個の推定誤りパターンが得られる。このうち 2 つの推定誤りパターン (0000001),(0011100) がシンδροームを 0 とする。ここで、前者の (0000001) を重畳誤りパターンと判定すると復号メッセージは $R=(0000000)$ となり、 S とは異なる送信ビット列に誤って復号される。

符号の最小距離は 6 であるから、低信頼ビットの数 (t) が 5 以下の場合、見逃し誤りを発生する誤りパターンは現れない [2]。一方、 t が 6 以

$$g(x) = x^{24} + x^{23} + x^{22} + x^{21} + x^{20} + x^{19} + x^{18} + x^{17} + x^{16} + x^{15} + x^{14} + x^{13} + x^{12} + x^{10} + x^3 + 1$$

$$= (x+1)(x^6 + x^5 + x^4 + x^2 + 1)(x^{17} + x^{16} + x^{15} + x^{13} + x^{10} + x^8 + x^7 + x^6 + x^5 + x^4 + x^3 + x + 1)$$

図 4 符号の生成多項式

上の場合、見逃し誤りが発生する場合がある。

[1]では、見逃し誤りの発生や探索数の増加などの理由から、探索する低信頼性ビットの数を5としている。

3. Brute Force 方式の改善

低信頼性ビット列のうち、多くの低信頼性ビット列は、見逃し誤りを発生しない。重みが6なる低信頼性ビット列の総数は約26億個($= {}_{112}C_6$)であるが、このうち見逃し誤りとなるパターンは、数百個である。従って、訂正を試みる低信頼性ビットの数(t)を増加すれば、復号率を改善できる。そこで本稿では、BruteForce方式の改善策として、tを増やすことにより復号率を改善する方法を提案する。

一方、tが増加すると見逃し誤りも増加するという欠点がある。そこで、見逃し誤りの原因となる低信頼性ビット列を判定し、このような低信頼性ビット列の場合には誤り訂正を行わないことで、見逃し誤りの発生を抑制する。3.1に見逃し誤りの判定法を示す。

また、tが増加すると、推定誤りパターンの探索数が指数的に増加するため、探索が困難となる。そこで、3.2では、探索数を削減する方法について述べる。3.3では、提案法の全体を示す。3.4では、提案法の高信頼性ビット誤りに対する特性について述べる。

3.1 見逃し誤りパターンの判定

低信頼性ビット列Cから生成される推定誤りパターン E_i ($i=1, \dots, 2^t-1$)の中に、符号語Wがある場合、見逃し誤りが発生する可能性がある。例えば、2.2.2節の低信頼性ビット列0011101から生成される誤り推定パターン E_i

中に、0011101という符号語が含まれる。このため、見逃し誤りが生じている。符号語は、シンドロームを計算により、簡単に判別できる。つまり、 E_i を順次生成し、シンドローム計算を行い、シンドロームを0とする E_i があった場合、見逃し誤りを発生する低信頼性ビット列であると判定できる。

4.2 探索数の削減

生成多項式 $g(x)$ は、1ビットのパリティを生成する多項式 $g_1(x) = x+1$ が含むため(図4に $g(x)$ の既約多項式分解を示す。)送信ビット列は偶パリティとなっている。これより、決定ビット列のパリティを計算することで、決定ビット列中に奇数個の誤り、または、偶数個の誤りのいずれかが生じていることがわかる。

従って、全ての誤りパターンを探索する必要はなく、パリティが1の場合、重みが奇数の推定誤りパターンを、0の場合、重みが偶数の推定誤りパターンを探索すればよい。これにより、探索する推定誤りパターンの数は1/2に削減できる。表2の第2列に、探索する推定誤りパターンの数を示す。

表2 探索する推定誤りパターンの数

低信頼性ビット数	探索1	探索2	合計探索数(A)	BF探索数(B)	A/B
1,3,5	1,4,16	-	1,4,16	1,7,31	
2,4	1,7,	-	1,7	3,15	
6	31		31	63	0.49
7	64	7	71	127	0.56
8	127	-	127	255	0.50
9	256	93	340	511	0.68
10	511	-	511	1023	0.50
11	1024	638	1662	2047	0.81
12	2047	-	2047	4095	0.50
13	4096	3302	7398	8191	0.90
14	8191	-	8191	16383	0.50

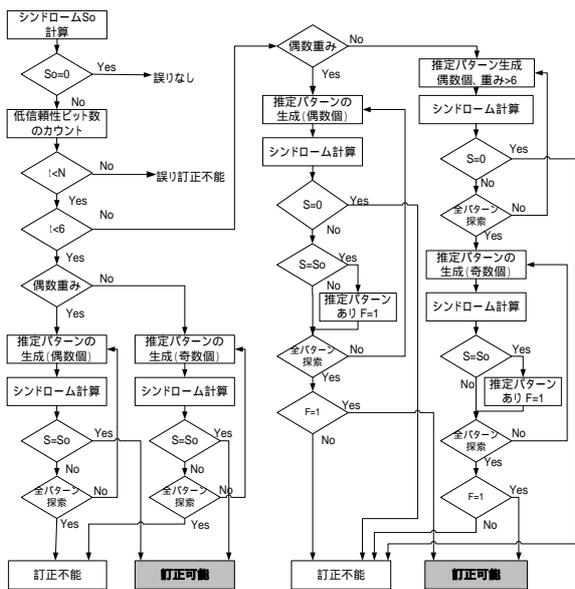


図 5 提案法のフローチャート

ただし、信頼性ビット数が7以上かつ奇数重みのパターンを探索する場合、見逃し誤りの判定のため、6以上の偶数重みの推定誤りパターンを生成し、検査する必要がある。表2の第3列に見逃し誤り判定のために必要となる推定誤りパターン列の数を、第4列に提案法による全探索数を示す。提案法による探索数は、BruteForce方式と比べ、50~90%となる。

3.3 提案法の全体

ここで、3.1 および 3.2 をまとめ、提案法による処理の全体を考える。図5に提案法の全体をフローチャートで示す。まず、決定ビット列のシンドローム S_0 の値を得る。次に、低信頼性ビットの数(t)をカウントし N 以上である場合、訂正不能と判定する。 $t < N$ 以下の場合、訂正を試みる。 $t < 6$ 場合、図5の左下の処理に移る。パリティを調べ、誤りパターンの重みの偶奇を決める。その後、推定誤りパターン生成、シンドローム計算を繰り返し、一致する誤りパターンを探索する。 $t \geq 6$ の場合、図5の右側の処理に移る。パリティを調べ、誤りパターンの重みの偶奇を決める。偶数の場合、偶重みパターンを生成し、シンドローム計算を行った後、見逃し

し誤りの判定と誤りパターン探索を行う。奇数の場合、まず偶重みパターンを生成し見逃し誤りの可能性を判定する。次に奇数重みの誤りパターン生成し、シンドローム計算を行い、誤りパターンを探索する。

3.3 高信頼性ビットの誤り

通信中、高信頼性と判定されたビットに誤り生じる場合がある。この場合にも、見逃し誤りが発生する。提案法では、1ビットの高信頼性ビットの誤りが起きた場合、誤り訂正不能となるため、見逃し誤りを回避できる。

例として、 $S=(0000000)$ を送信し、第5ビットに高信頼性ビット誤り、第6,7ビットに低信頼性ビット誤りが発生した場合を考える。決定ビット列 $D=(0000111)$, 低信頼性ビット $C=(000011)$ となる。決定ビット列のパリティは1であるから、奇数重みの推定誤りパターン(000001),(0000010)を生成する。ところが、両者とも $S=0$ となるため、最終的に誤り訂正不能となる。提案法では、1ビットの高信頼性誤りが起きた場合でも、見逃し誤りを生じない。

4. まとめ

本稿では、Brute Force方式において訂正を試みる低信頼性ビットの数を増加させることにより、復号率を向上する方法を提案した。提案法は、見逃し誤りを発生する低信頼性ビット列を排除することで、見逃し誤りを抑制する。また、生成多項式の性質を利用することにより、推定誤りパターンの探索数を削減すると共に、高信頼性誤りによる誤りを抑制できる。今後は、実環境のデータに元に、本方式の特性を評価する。また、誤り訂正方式の他、ビット判定方式の改良などについても検討する予定である。

参考文献

[1]RTCA," Minimum Operational Performance Standards for 1090MHz Automatic Dependent Surveillance Broadcast (ADS-B)", Do-260, 2000
 [2]今井,"符号理論",電子情報通信学会,1990