

誤り制御符号 (1) : パリティ符号

1 目的

パリティ符号の生成と検査を行うプログラムの作成を通じて、誤り制御符号についての理解を深める。

2 誤り制御の原理

通信回線上や通信機器の回路内には、多かれ少なかれノイズ（雑音）が存在するため、ネットワークを介して交換されるデータはノイズの影響を受ける。また、伝送途中で電気信号が歪むこともある。このため、送信局が送出したものと異なるデータが受信局に届く可能性がある（図 1 参照）。ネットワークを介して正しいデータ交換を行うためには、受信局はデータの誤り（エラー）を検出し、正しいデータに訂正する手段が必要である。このような仕組みを誤り制御（**Error Control**）と呼ぶ。

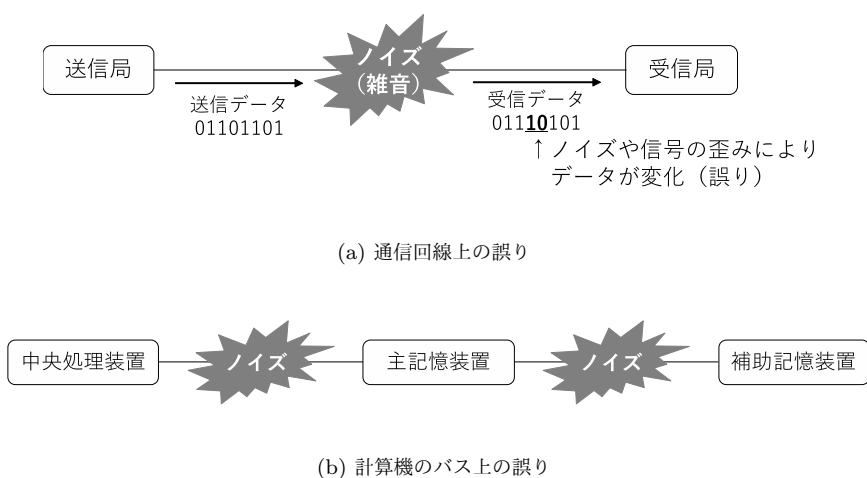


図 1 ノイズの影響とエラーの発生

誤り制御を実現するため、ネットワーク上で交換されるデータ（符号語）には、本来のデータ（情報語）に加えて、検査用の符号（検査語）が含まれている。このような符号語を誤り制御符号（**Error Control Code**）と呼ぶ。

3 誤り制御符号の原理

前述のように、誤り制御符号は、本来のデータ（情報語）に検査語を加えたものである。しかし、一般に計算機内部や通信回線上では、一度に大きな情報語が送受信される。そこで、図 2 に示すように、誤り制御符号には 2 種類の構成方法がある。図 2(a) のような長い情報語を送受信する場合を考える。このとき、図 2(b) のように、情報語を予め定められた単位長さに区切り、各々の情報語に短い検査語を加える方法がある。一方で、図 2(c) のように、情報語を途中で区切ることなく、末尾（あるいは先頭）に比較的長い検査語を加える方法もある。前者は主に、計算機内でのデータ転送や、アプリケーション中でデータを読み書きする際の誤り制御に利用される。後者は主に、通信回線上を介してデータを転送する際の誤り制御に利用される。今回の実験で取り上げるパリティ符号は、前者である。

続いて、誤り制御の手法について解説する。誤り制御の手法は、誤り検出（**Error Detection**）と誤り訂正（**Error Correction**）に大別される。このような分類に応じて、誤り制御符号も二種類に大別される。誤り検出に用いられる符号を誤り検出符号（**Error Detection Code**）と呼び、誤り訂正に用いられる符号を誤り訂正符号（**Error Correction Code**）と呼ぶ（図 3 参照）。誤り検出の手順について解説する。送信局はまず、送信したい情報語から検査語の値を算出し、符号語を構成する。これが送信データとなる。送信データは通信回線を介して受信局に伝送される。データを受け取った受信局は、受信した符号語の内容から検査語の値を算出し、受信データ内の検査語と比較する。両者が一致していれば誤りは生じなかったことを意味する。両者が一致しない場合は伝送途中に誤りが発生したことになるため、受信局は送信局に対し

て誤りの発生を通知する。誤りの通知を受けた送信局は、該当するデータを再び送信する。なお、多くのネットワークでは、受信局は誤りがなかった場合にもその旨を送信局に通知する。そのため、送信局にはつねに何らかの応答が戻される。

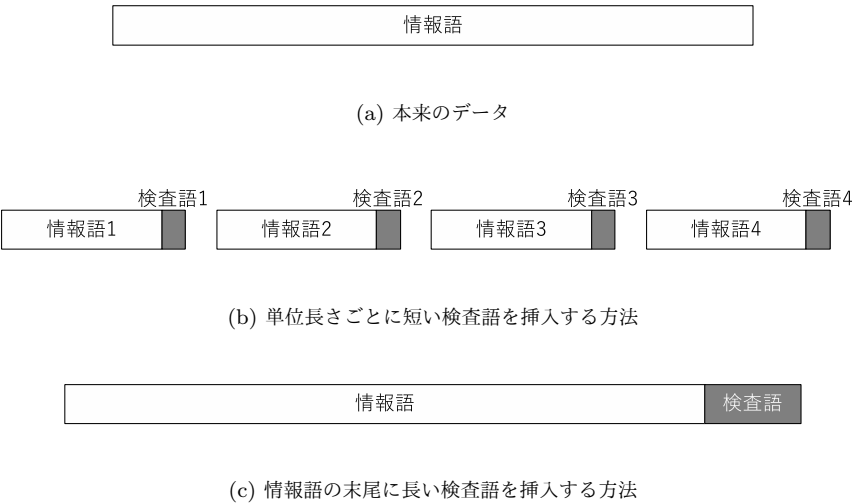


図 2 誤り制御符号の構成方法

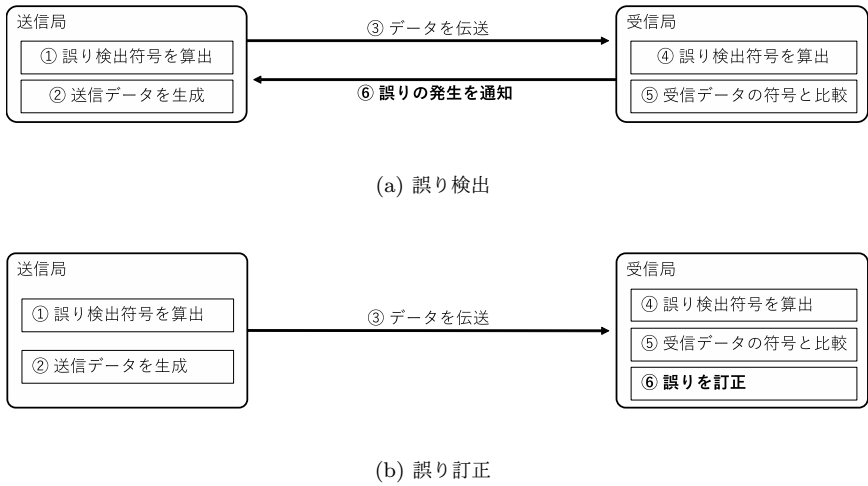


図 3 誤り検出と誤り訂正

4 1 ビット垂直パリティ符号

パリティ符号 (Parity Code) は、最も基本的な誤り検出符号のひとつである。検査語に複数ビットを使用する大きなパリティ符号もあるが、今回は大きさが 1 ビットの 1 ビット垂直パリティ符号を題材として学習を進める。

1 ビット垂直パリティ符号は、送信される情報語 (元データ) の 1 文字あるいは 1 バイトごとに 1 ビットの検査語を付加する方式である。図 4 に電子メールで採用されている 1 ビット垂直パリティの例を示す。送信局から受信局に「TEST!」という文字列を送信するものとする。電子メールでは、各々の文字は 7 ビット JIS コードで表現される。「TEST!」は 5 つ (5 語) の 7 ビット符号で表現される。送信局はこの 5 語のそれぞれに対して、“1” であるビットの数を求める。そして、各々の語に対して、その語の中で “1” が奇数であれば “1” を、偶数であれば “0” を付加して 8 ビットの符号語 (送信データ) を生成する。このため、符号語の中の各語では必ず、“1” の個数は偶数になる。

一方、受信局では、それぞれの符号語に対して “1” であるビットの数を求める。偶数であればその語は正常に伝送されており、奇数であればその語には誤りが含まれていると判定する。すべての符号語で誤りが発見されなかった場合、受信局は受信データの各語から検査語を除去することにより情報語を取り出し、これを受け取る。

なお、符号語の中の “1” の個数を偶数個に保とうとするものを偶数パリティ、奇数個に保とうとするものを奇数パリティと呼ぶ。

情報語（文字表現）

TEST!

情報語
(7ビットJIS表現)

0010101 1010001 1100101 0010101 1000010
“1”は3個 “1”は3個 “1”は4個 “1”は3個 “1”は2個

符号語
(送信データ)

0010101**1** 1010001**1** 1100101**0** 0010101**1** 1000010**0**
検査語（パリティビット）

図4 1ビット垂直パリティ符号

いま、情報語を $c_0 \sim c_{k-1}$ までの k 桁の情報ビット列とする。ここで、1ビット排他的論理和の演算子を \oplus と定義する。このとき、送信局は符号語の中の“1”の個数を偶数に保つため、次のブール式によりパリティビットの値を求める。

$$c_k = c_0 \oplus c_1 \oplus c_2 \oplus \dots \oplus c_{k-1}$$

送信局は、情報ビット列にパリティビット c_k を加えて、 $k+1$ 桁の伝送用の符号ビット列 $c_0 c_1 c_2 \dots c_k$ を作成し、これを伝送する。一方、 $k+1$ 桁の符号ビット列において“1”の個数が偶数のとき、次のブール式が成り立つ。

$$c_0 \oplus c_1 \oplus c_2 \oplus \dots \oplus c_{k-1} \oplus c_k = 0$$

上述のブール式より、1ビット垂直パリティ符号の符号化／復号化回路は、1ビットシフトレジスタの回路および1ビット排他的論理和の回路を各1個で構成できる。パリティ符号は、誤り検出能力は高くないが、回路が簡易かつ安価で、小型の回路で高速に符号化／復号化の処理を行うことが可能である。このため、類似した手法が計算機システム内の主記憶装置の誤り検出にも利用されている。

5 1ビット垂直パリティ符号の実験

5.1 プログラムの作成

次のように入力して、パリティ符号のソースプログラムを実験用ディレクトリにコピーする。

```
$ cd ~/cel-B
$ cp sample-programs/parity.c .
```

正常にコピーができたなら、次のように入力してコンパイルを行う。コンパイルが正常に終了すると、`parity` という実行可能ファイルが生成される。

```
$ gcc -o parity parity.c
```

5.2 プログラムの実行

次のように入力して、パリティ符号のプログラムを実行する。このプログラムは最初に送信局の処理として、乱数により適当なデータを生成し、それに付加すべきパリティビットを算出する。続いて、データを通信路に送出したことを想定し、ランダムにビット誤りを発生させる。最後に、受信局での処理として、受信したデータからパリティビットを算出し、受信データに付与されているものと比較し、エラーの有無を判定する。

```
$ ./parity
```

5.3 パリティ符号の性質の確認

作成したプログラムを何度か実行し、誤りが正しく検出されるとき、誤りがあっても検出されないときの実行例を記録しておく。テキストファイル等にコピーしておくといい。

6 報告内容

レポートには、以下の内容を記載すること。

6.1 実行結果

誤りを検知した場合、検知できなかった場合を含め、次節の考察を行うために必要となる結果を掲載すること。

6.2 実行結果に対する考察

前節の実験結果を用いて、1 ビット垂直パリティ符号の性質を説明すること。

6.3 課題

- (1) 今回の実験で作成したパリティ符号は、偶数パリティと奇数パリティのいずれであるかを答えよ。
- (2) 1 ビット水平パリティ符号について調査せよ。
- (3) 1 ビット水平パリティ符号と 1 ビット垂直パリティ符号を組み合わせることにより、1 ビットの誤りを訂正できることを示せ。

参考文献

- [1] 福永邦雄，泉正夫，荻原昭夫，「コンピュータ通信とネットワーク 第5版」，共立出版，2002 年。
- [2] 荻原春生，中川健治，「情報通信理論 1 —符号理論・待ち行列理論—」，森北出版，1997 年。
- [3] イエルン・ヨステンセン，トム・ホーホルト，「誤り訂正符号入門」，森北出版，2005 年。