

目次

第 1 章	はじめに	5
第 2 章	情報理論の基礎 (Shanon の情報量)	7
2.1	エントロピー	7
2.2	情報量	9
2.3	複合事象のエントロピー	10
2.4	条件付きエントロピー	13
2.5	相互情報量	15
第 3 章	情報の表現 (数系)	19
3.1	位取り表現法	19
3.2	基数変換 (radix transformation)	21
3.3	負の数の表現法	22
3.4	r 進小数	24
3.5	固定小数点表現と浮動小数点表現	25
第 4 章	情報の伝達 (符号理論)	29
4.1	通信系のモデル	29
4.2	Hamming 距離	31
4.3	誤り検出・訂正の原理	32
4.4	線形符号	33
4.5	線形符号の復号	34
4.6	単一パリティ検査記号	35
4.7	Hamming 符号	36
4.8	補遺	37
第 5 章	集合と論理	41
5.1	集合	41
5.1.1	集合の定義	41
5.1.2	集合の演算	41
5.1.3	有限集合と無限集合	43
5.1.4	数学的帰納法	45
第 6 章	ブール代数と基本論理回路	47
6.1	ブール代数	47
6.2	MIL 記号	49

第 7 章	状態機械	53
7.1	有限状態機械	53
7.2	有限オートマトン	55
第 8 章	読書案内	57
第 9 章	練習問題略解	61

第1章 はじめに

この講義では、情報理論 (Information Theory) や符号理論 (Coding Theory) の基礎概念を織り交ぜながら、離散数学 (Discrete Mathematics) と呼ばれる学問の基礎を学ぶ。

計算機内部では、計算機がどのような状態にあるかは離散的かつ有限な量で表されており、状態の変化もある時間間隔で離散的に発生する。このような離散的対象を表現・解析・設計するために、離散数学と呼ばれる数学が用いられる。離散数学自身は非常に広範な内容を持っているが、本講義で扱うのはその中の基礎的項目のいくつかである。また、通常、離散数学には含まれないが通信システムに関連した重要な基礎理論である情報理論や符号理論についても触れる。具体的には、講義内容は以下になる予定である。

- 計算機の歴史
- 情報理論の基礎 (Shannon の情報量)
- 情報の表現 (数系)
- 情報の伝達 (符号理論)
- 集合と論理
- ブール代数と基本論理回路
- 状態遷移

この中で、計算機の歴史については、本格的な講義をする素養が筆者にはないので、計算機の歴史に関するトピックスを随時取り上げることで、講義に代えることにしたい。なお、この講義では特に教科書は指定しないが、このテキストの巻末に参考書をいくつか挙げておく。

電気電子システム工学科 S クラスでは新入生用に推奨ノートパソコンを指定しているが、その中に対話形科学技術計算言語 Octave がインストールされている。Octave を用いると、複雑な数値計算がきわめて簡単に計算できるので、このテキストの中にも Octave の使用例をいくつか提示しておいた。Octave に興味のある人は、<http://www.sp.es.yamanashi.ac.jp/~ohki/matlab/> を参照されたい。

第2章 情報理論の基礎 (Shanonの情報量)

一口に「情報」(Information)と言っても、その具体的な形態は様々であり、情報を受け取る人によって情報の価値もまちまちである。したがって、「情報」とは何か、ということを一般的に定義することは難しい。そこで、C.E.Shannon に始まる現代の情報理論においては、情報の形態や価値といったものを一旦無視して、「情報は、われわれの不確実な知識を確実にしてくれるものである」という点だけに注目して情報の量を定義する。

2.1 エントロピー

まず最初に、状況の不確実さを測る量を定義しよう。今、起こりうる事柄(「事象」と呼ぶ)が $A_1, A_2, \dots, A_N$ の N 個あるものとし、それぞれの起こる確率を $p_1, p_2, \dots, p_N$ であるとしよう。ここで、確率の定義からすべての事象の確率の和は 1 に等しい ($\sum_{i=1}^N p_i = 1$)。例えば、コイン投げという状況を考えると、事象の数 $N = 2$ 、事象 $A_1 =$ 「表が出る」、事象 $A_2 =$ 「裏が出る」と置くことができ、またコインが特に歪んでいなければ $p_1 = p_2 = \frac{1}{2}$ と考えるのが普通であろう。さて、これからコインを投げようとしている状況の不確実さはどのようにあらわされるだろうか？もし、コインが極端に歪んでいて、どう考えても表しか出そうにない ($p_1 \simeq 1$) 場合には状況には不確実な部分はほとんど無いから、不確実さ $\simeq 0$ と考えるのが自然である。逆に、裏しか出そうに無い場合にも不確実さは 0 と考えられる。結局、どちらの事象も同じ確率で起こる普通のコインの場合がもっとも不確実さが大きいと考えられる。このように、不確実さを表す量は特定の事象が特に起きやすい(確率が大きい)場合には 0 に近く、すべての事象が同じように起きやすい時に最大となるような性質を持っている必要がある。このような性質を満たす不確実さの量として、以下のような「エントロピー」(entropy)を定義することができる¹。

定義 1 (エントロピー)

N 個の事象 $A_1, A_2, \dots, A_N$ の起こる確率が $p_1, p_2, \dots, p_N$ である状況を考えてとき、以下の式で示す量 H を「エントロピー」と呼ぶ。

$$H(p_1, p_2, \dots, p_N) = - \sum_{i=1}^N p_i \log_2(p_i) \quad (2.1)$$

(注) ここでは、2 の対数を使用しているが、常用対数や自然対数を用いることもできる。 ■

例 1 歪んでいない普通のコインを投げる場合を考えると、表の出る確率 p_1 と裏の出る確率 p_2 は

¹エントロピーの定義でなぜ対数関数を用いるのか疑問に思うかもしれないが、実は情報量の満たすべき性質をいくつか定めると、それに対応するエントロピーの定義に用いる関数としては対数関数以外ありえないことが証明できる。興味のある読者は「甘利 俊一：情報理論，ダイヤモンド社」などを参照されたい。

共に $\frac{1}{2}$ と考えられるので, エントロピー H は以下のように計算できる.

$$\begin{aligned} H &= -\sum_{i=1}^2 p_i \log_2(p_i) \\ &= -\frac{1}{2} \log_2\left(\frac{1}{2}\right) - \frac{1}{2} \log_2\left(\frac{1}{2}\right) \\ &= -\log_2\left(\frac{1}{2}\right) \end{aligned} \tag{2.2}$$

$$\begin{aligned} &= -\log_2(2^{-1}) \\ &= \log_2(2) \\ &= 1 \end{aligned} \tag{2.3}$$

例2 少々歪んだコインを投げる場合を考え, 表の出る確率 $p_1 = \frac{3}{4}$, 裏の出る確率 $p_2 = \frac{1}{4}$ と仮定してみよう. エントロピー H は以下のように計算できる.

$$\begin{aligned} H &= -\sum_{i=1}^2 p_i \log_2(p_i) \\ &= -\frac{3}{4} \log_2\left(\frac{3}{4}\right) - \frac{1}{4} \log_2\left(\frac{1}{4}\right) \\ &= \frac{3}{4} \log_2\left(\frac{4}{3}\right) + \frac{1}{4} \log_2(4) \\ &= \frac{3}{4}(\log_2(4) - \log_2(3)) + \frac{1}{4} \log_2(4) \\ &= \frac{3}{4}(2 - \log_2(3)) + \frac{1}{4} \cdot 2 \\ &= \frac{3}{2} - \frac{3}{4} \log_2(3) + \frac{1}{2} \\ &= 2 - \frac{3}{4} \log_2(3) \\ &\simeq 2 - 0.75 \times 1.59 \\ &= 0.8075 \\ &\simeq 0.808 \end{aligned} \tag{2.4}$$

このように, コインが公正なものでないときにはコインが公正な場合よりもエントロピーが減少する.

例3 4月15日の天気について過去の統計から次のような確率が得られたとする; 晴れの確率 $p_1 = 0.25$, 曇りの確率 $p_2 = 0.5$, 雨の確率 $p_3 = 0.25$. 今年の4月15日の天気に関するエントロピー

(不確実さ) H は以下のように計算できる .

$$\begin{aligned}
 H &= - \sum_{i=1}^3 p_i \log_2(p_i) \\
 &= -0.25 \log_2(0.25) - 0.5 \log_2(0.5) - 0.25 \log_2(0.25) \\
 &= 0.25 \log_2(4) + 0.5 \log_2(2) + 0.25 \log_2(4) \\
 &= 0.25 \times 2 + 0.5 \times 1 + 0.25 \times 2 \\
 &= 0.5 + 0.5 + 0.5 \\
 &= 1.5
 \end{aligned} \tag{2.5}$$

例 4 上記の例の計算を Octave を用いて計算してみよう .

```
[me@linuxbox me] octave
octave:1> p = [0.25 0.5 0.25]
p =
    0.25000    0.50000    0.25000

octave:2> H = - sum(p .* log2(p))
H = 1.5000
octave:3> exit
[me@linuxbox me]
```

このように , Octave を使う場合 , 確率を表すベクトル p さえ入力してしまえばエントロピーの計算はたった 1 行の式で計算できるので極めて簡単である .

2.2 情報量

情報量は不確実さの減少の度合いと考えることができるので , 前節で定義したエントロピーを用いて情報量を以下のように定義することができる .

定義 2 (情報量)

ある事象が起こる前の状況のエントロピーを H_0 , 事象が起こった後の状況のエントロピーを H_1 とすると , この事象が起こることを知ることによって得られる情報量 I は次の式で定義される .

$$I = H_0 - H_1 \tag{2.6}$$

■

例えば , もう一度コイン投げを考えてみると , 歪んでいないコインを投げる前の状況のエントロピー H_0 は , 先に求めたように $H_0 = 1$ であり , 一方 , コインを投げた後ではコインの裏表は決定してしまっていて不確実さは無いので $H_1 = 0$ である . したがって , コインの裏表のどちらが出るかを知ることによって得られる情報量は ,

$$I = H_0 - H_1 = 1 - 0 = 1 \tag{2.7}$$

と計算できる．このコイン投げの場合のような二者択一の状況は情報量を考える場合のもっとも単純な状況と考えられるので，この時の情報量を情報量の単位として用いることが多く，これを「ビット」(bit)と呼ぶ．よって，コイン投げの情報量は 1bit と表現されることになる．

2.3 複合事象のエントロピー

事象の中には，「天気は晴れで，気温は 20 度以上」といったように，「天気」という事象系 A ($A_1 = \text{「晴れ」}, A_2 = \text{「曇り」}, A_3 = \text{「雨」}$) と「気温」という事象系 B ($B_1 = \text{「気温は 20 度未満」}, B_2 = \text{「気温は 20 度以上」}$) という複数の事象系の組み合わせによって表されるものが少なくない．これらを「複合事象」と呼ぶ．本節では，複合事象のエントロピーについて考えてみよう．

まず， M 個の事象から成る事象系 A において事象 A_i が生じ， N 個の事象から成る事象系 B において事象 B_j が生じる確率を $p(A_i, B_j)$ と書くことにしよう．これを「同時確率分布」と呼ぶ．すると，先に定義したエントロピーの式を同時確率分布に当てはめることによって，この複合事象のエントロピーは次のように計算することができる．

定義 3 (複合事象のエントロピー)

$$H(A, B) = - \sum_{i=1}^M \sum_{j=1}^N p(A_i, B_j) \log_2(p(A_i, B_j)) \quad (2.8)$$

■

次に， B に関する事象には興味が無く，どの A_i が起こるかだけに関心のある場合を考えてみよう．このとき， B でどのような事象が起こるかに関わりなく， A で事象 A_i が起こる確率 $p(A_i)$ を周辺確率分布と呼ぶ． $p(A_i)$ は同時確率分布から次のように計算できる．

$$p(A_i) = \sum_{j=1}^N p(A_i, B_j) \quad (2.9)$$

したがって， A だけに着目したときのエントロピーは，

$$H(A) = - \sum_{i=1}^M p(A_i) \log_2(p(A_i)) \quad (2.10)$$

同様に， A に関わりなく B で B_j が起こる確率 (これも周辺確率分布と呼ぶ) と B のみに着目したエントロピーは，

$$p(B_j) = \sum_{i=1}^M p(A_i, B_j) \quad (2.11)$$

$$H(B) = - \sum_{j=1}^N p(B_j) \log_2(p(B_j)) \quad (2.12)$$

である．

例 5 「天気」という事象系 A ($A_1 = \text{「晴れ」}$, $A_2 = \text{「晴れ以外」}$) と「気温」という事象系 B ($B_1 = \text{「気温は 20 度未満」}$, $B_2 = \text{「気温は 20 度以上」}$) の複合事象系 (A, B) を考えよう。この事象系の確率が以下のように求まったとする。

1. 晴れ (A_1) で 20 度未満 (B_1) の確率 $p(A_1, B_1) = \frac{1}{5}$
2. 晴れ (A_1) で 20 度以上 (B_2) の確率 $p(A_1, B_2) = \frac{2}{5}$
3. 晴れ以外 (A_2) で 20 度未満 (B_1) の確率 $p(A_2, B_1) = \frac{3}{10}$
4. 晴れ以外 (A_2) で 20 度以上 (B_2) の確率 $p(A_2, B_2) = \frac{1}{10}$

まず周辺確率分布を求めると、以下のようになる。

$$p(A_1) = \sum_{j=1}^2 p(A_1, B_j) = \frac{1}{5} + \frac{2}{5} = \frac{3}{5} \quad (2.13)$$

$$p(A_2) = \sum_{j=1}^2 p(A_2, B_j) = \frac{3}{10} + \frac{1}{10} = \frac{2}{5} \quad (2.14)$$

$$p(B_1) = \sum_{i=1}^2 p(A_i, B_1) = \frac{1}{5} + \frac{3}{10} = \frac{1}{2} \quad (2.15)$$

$$p(B_2) = \sum_{i=1}^2 p(A_i, B_2) = \frac{2}{5} + \frac{1}{10} = \frac{1}{2} \quad (2.16)$$

これから、周辺確率分布のエントロピーは次のように計算される。

$$\begin{aligned} H(A) &= -\sum_{i=1}^2 p(A_i) \log_2(p(A_i)) = -\frac{3}{5} \log_2\left(\frac{3}{5}\right) - \frac{2}{5} \log_2\left(\frac{2}{5}\right) \\ &= \frac{3}{5}(\log_2(5) - \log_2(3)) + \frac{2}{5}(\log_2(5) - \log_2(2)) \\ &= \log_2(5) - \frac{3}{5} \log_2(3) - \frac{2}{5} \\ &\simeq 2.32 - 0.6 \times 1.59 - 0.4 = 0.966 \end{aligned} \quad (2.17)$$

$$\begin{aligned} H(B) &= -\sum_{i=1}^2 p(B_i) \log_2(p(B_i)) = -\frac{1}{2} \log_2\left(\frac{1}{2}\right) - \frac{1}{2} \log_2\left(\frac{1}{2}\right) \\ &= \log_2(2) = 1 \end{aligned} \quad (2.18)$$

一方，同時確率分布のエントロピーは以下の通りである．

$$\begin{aligned}
 H(A, B) &= - \sum_{i=1}^2 \sum_{j=1}^2 p(A_i, B_j) \log_2(p(A_i, B_j)) \\
 &= -\frac{1}{5} \log_2 \left(\frac{1}{5} \right) - \frac{2}{5} \log_2 \left(\frac{2}{5} \right) - \frac{3}{10} \log_2 \left(\frac{3}{10} \right) - \frac{1}{10} \log_2 \left(\frac{1}{10} \right) \\
 &= \frac{1}{5} \log_2(5) + \frac{2}{5} (\log_2(5) - \log_2(2)) + \frac{3}{10} (\log_2(10) - \log_2(3)) + \frac{1}{10} \log_2(10) \\
 &= \frac{1}{5} \log_2(5) + \frac{2}{5} (\log_2(5) - \log_2(2)) + \frac{3}{10} (\log_2(2) + \log_2(5) - \log_2(3)) \\
 &\quad + \frac{1}{10} (\log_2(2) + \log_2(5)) \\
 &= \frac{1}{5} \log_2(5) + \frac{2}{5} (\log_2(5) - 1) + \frac{3}{10} (1 + \log_2(5) - \log_2(3)) + \frac{1}{10} (1 + \log_2(5)) \\
 &= \log_2(5) - \frac{3}{10} \log_2(3) \\
 &\simeq 2.32 - 0.3 \times 1.59 = 1.843 \simeq 1.84
 \end{aligned} \tag{2.19}$$

例 6 上記の例題を Octave で検算してみよう．

```
octave:1> p = [2/5 1/5 ; 1/10 3/10]
```

```
p =
```

```
0.40000 0.20000
0.10000 0.30000
```

```
octave:2> pA = sum(p')
```

```
pA =
```

```
0.60000 0.40000
```

```
octave:3> pB = sum(p)
```

```
pB =
```

```
0.50000 0.50000
```

```
octave:4> HA = - sum(pA .* log2(pA))
```

```
HA = 0.97095
```

```
octave:5> HB = - sum(pB .* log2(pB))
```

```
HB = 1
```

```
octave:6> H = - sum(sum(p .* log2(p)))
```

```
H = 1.8464
```

octave:7>

なお，数値が手計算といくらか異なっているが，これは手計算と Octave では有効数字の桁数が違うためである．

2.4 条件付きエントロピー

複合事象を考えると各事象系は互いに無関係ではなく，事象系 A で何が起きたかを知ってしまえば事象系 B に関する不確かさが減少することが多いと考えられる．このような状況を扱うために，本節では「条件付き確率」と「条件付きエントロピー」というものを導入する．

前節と同様に，事象系 A と B を考えよう．事象系 A で事象 A_i が起こったという条件のもとで，事象系 B において事象 B_j が起こる確率を「条件付き確率」と呼び， $p_{A_i}(B_j)$ と表す．同様に， B_j が起こったという条件の下で A_i が起こる条件付き確率は $p_{B_j}(A_i)$ である．前節で定義した同時確率分布，周辺確率分布と条件付き確率分布の間には以下のような関係が成立する．

1.

$$\begin{aligned} \sum_{i=1}^M \sum_{j=1}^N p(A_i, B_j) &= 1 \\ \sum_{i=1}^M p(A_i) &= \sum_{j=1}^N p(B_j) = 1 \\ \sum_{j=1}^N p_{A_i}(B_j) &= \sum_{i=1}^M p_{B_j}(A_i) = 1 \end{aligned} \quad (2.20)$$

2.

$$\begin{aligned} p(A_i) &= \sum_{j=1}^N p(A_i, B_j) \\ p(B_j) &= \sum_{i=1}^M p(A_i, B_j) \end{aligned} \quad (2.21)$$

3.

$$p(A_i, B_j) = p(A_i)p_{A_i}(B_j) = p(B_j)p_{B_j}(A_i) \quad (2.22)$$

4.

$$\begin{aligned} p_{A_i}(B_j) &= \frac{p(B_j)p_{B_j}(A_i)}{p(A_i)} = \frac{p(A_i, B_j)}{p(A_i)} \\ p_{B_j}(A_i) &= \frac{p(A_i)p_{A_i}(B_j)}{p(B_j)} = \frac{p(A_i, B_j)}{p(B_j)} \end{aligned} \quad (2.23)$$

ここで，1. は考えている確率の総和が 1 にならなければならないという確率の満たすべき条件であり，2. は周辺確率分布の定義そのものである．3. は「 A_i と B_j が同時に起こる確率」というものが「 A_i が起きる確率」と「 A_i が起こる条件の下で B_j が起こる確率」との積，あるいは「 B_j が起きる確率」と「 B_j が起こる条件の下で A_i が起こる確率」との積に他ならない，という当たり前の事実を表している．4. は「Bayes の公式」と呼ばれる有名な公式であり，3. から直ちに導くことができる．

事象系 A と B が無関係な場合「 A と B は確率的に独立」と言い、以下の式が成り立つ。

1.

$$p(A_i, B_j) = p(A_i)p(B_j) \quad (2.24)$$

2.

$$p_{A_i}(B_j) = p(B_j), \quad p_{B_j}(A_i) = p(A_i) \quad (2.25)$$

さて、このような準備のもとで条件付きエントロピーを定義してみよう。事象系 A で事象 A_i が起こったことが分かった時に、事象系 B の各事象 B_j が起こる確率は条件付き確率 $p_{A_i}(B_j)$ となる。したがって、この場合のエントロピーは、

$$H_{A_i}(B) = - \sum_{j=1}^N p_{A_i}(B_j) \log_2(p_{A_i}(B_j)) \quad (2.26)$$

と書くことができる。一方、事象 A_i が起こる確率は $p(A_i)$ であるから、事象系 A で任意のある事象が起こったことが分かったときの条件付きエントロピー $H_A(B)$ は、上記の式 $H_{A_i}(B)$ を A_i について平均したものとなる。

定義 4 (条件付きエントロピー)

$$\begin{aligned} H_A(B) &= \sum_{i=1}^M p(A_i) H_{A_i}(B) \\ &= - \sum_{i=1}^M \sum_{j=1}^N p(A_i) p_{A_i}(B_j) \log_2(p_{A_i}(B_j)) \end{aligned} \quad (2.27)$$

■

条件付きエントロピーは以下のような性質を持っている。

性質 1

$$\begin{aligned} H(A, B) &= H(A) + H_A(B) \\ &= H(B) + H_B(A) \end{aligned} \quad (2.28)$$

性質 2

$$H_A(B) \geq 0 \quad (2.29)$$

性質 3

$$H(A) + H(B) \geq H(A, B) \quad (2.30)$$

性質 4

$$H(A) \geq H_B(A), \quad H(B) \geq H_A(B) \quad (2.31)$$

例 7 前節の天気と気温の複合事象系の例題を条件付確率の観点から見直してみよう．Bayes の公式を用いると，同時確率分布から以下のように条件付確率分布を求めることができる．

$$p_{A_1}(B_1) = \frac{p(A_1, B_1)}{p(A_1)} = \frac{1/5}{3/5} = \frac{1}{3} \quad (2.32)$$

$$p_{A_1}(B_2) = \frac{p(A_1, B_2)}{p(A_1)} = \frac{2/5}{3/5} = \frac{2}{3} \quad (2.33)$$

$$p_{A_2}(B_1) = \frac{p(A_2, B_1)}{p(A_2)} = \frac{3/10}{2/5} = \frac{3}{4} \quad (2.34)$$

$$p_{A_2}(B_2) = \frac{p(A_2, B_2)}{p(A_2)} = \frac{1/10}{2/5} = \frac{1}{4} \quad (2.35)$$

これから，条件付エントロピーは

$$\begin{aligned} H_A(B) &= - \sum_{i=1}^2 \sum_{j=1}^2 p(A_i) p_{A_i}(B_j) \log_2(p_{A_i}(B_j)) \\ &= - \frac{3}{5} \frac{1}{3} \log_2\left(\frac{1}{3}\right) - \frac{3}{5} \frac{2}{3} \log_2\left(\frac{2}{3}\right) - \frac{2}{5} \frac{3}{4} \log_2\left(\frac{3}{4}\right) - \frac{2}{5} \frac{1}{4} \log_2\left(\frac{1}{4}\right) \\ &= \frac{1}{5} \log_2(3) + \frac{2}{5} (\log_2(3) - \log_2(2)) + \frac{3}{10} (\log_2(4) - \log_2(3)) + \frac{1}{10} \log_2(4) \\ &= \frac{1}{5} \log_2(3) + \frac{2}{5} (\log_2(3) - 1) + \frac{3}{10} (2 - \log_2(3)) + \frac{1}{10} 2 \\ &= \frac{3}{10} \log_2(3) + \frac{2}{5} \\ &= 0.3 \times 1.59 + 0.4 = 0.877 \end{aligned} \quad (2.36)$$

のように計算される．天気を無視して気温についてだけ計算したエントロピーは $H(B) = 1$ であったので，天気についての確率を考慮したエントロピー $H_A(B)$ はそれよりも小さくなっていることが見て取れる．

2.5 相互情報量

事象系 A と B になんらかの関係があれば， A が何であるかを知ることによって， B が何であるかについても若干の情報が得られるはずである．先程の天気の例で言うと，今日の天気が分かれば，直接気温を測定しなくともおおよそ気温の予想がつくということである．この情報量は， A を知ることによって得られる B のエントロピーの減少分によって定義することができ，「相互情報量」 $I(A, B)$ と呼ばれる．

定義 5 (相互情報量)

$$I(A, B) = H(B) - H_A(B) \quad (2.37)$$

■

一般に何かを測定する場合には，測定される複合事象のすべてを把握することは難しい．そのような場合に，直接知ることのできる事象から残りの事象について推定することが重要となる．相互情報量は，そのような場合に得られる情報量を計算するために役立つ．

相互情報量の性質を以下に示す．

性質 1

$$I(A, B) = I(B, A) \quad (2.38)$$

性質 2

$$I(A, B) \geq H(A), H(B) \quad (2.39)$$

性質 3

$$I(A, B) \geq 0 \quad (2.40)$$

なお, 3 つ以上の事象系についても相互情報量を定義することができるが, 煩雑になるのでここでは省略する.

例 8 前節の天気と気温の例では, 天気を知ることによって得られる気温に関する情報量 (相互情報量) は以下のように計算される.

$$I(A, B) = H(B) - H_A(B) = 1 - 0.877 = 0.133 \text{ [bit]} \quad (2.41)$$

演習問題

問 1 ある講義について過去 10 年間の学生の出席率と天気の関係について調べたところ, 以下のような結果であった.

- 講義の延べ回数 $N = 100$
- 天気が晴れの時出席率が 75%以上だった回数 $N_{11} = 30$
- 天気が晴れの時出席率が 75%未満だった回数 $N_{12} = 10$
- 天気が曇りの時出席率が 75%以上だった回数 $N_{21} = 20$
- 天気が曇りの時出席率が 75%未満だった回数 $N_{22} = 15$
- 天気が雨の時出席率が 75%以上だった回数 $N_{31} = 10$
- 天気が雨の時出席率が 75%未満だった回数 $N_{32} = 15$

以下の問に答えよ.

1. 天気に関する事象 A ,

$$A = \{A_1, A_2, A_3\}, \quad A_1 = \text{晴れ}, \quad A_2 = \text{曇り}, \quad A_3 = \text{雨}$$

学生の出席率に関する事象 B ,

$$B = \{B_1, B_2\}, \quad B_1 = \text{出席率 75\%以上} \quad B_2 = \text{出席率 75\%未満}$$

と定める. 過去 10 年間の経験から, 複合事象系 (A, B) の結合確率分布 $p(A_1, B_1), p(A_1, B_2), p(A_2, B_1), p(A_2, B_2), p(A_3, B_1), p(A_3, B_2)$ を求めよ.

2. 上で求めた結合確率から周辺確率 $p(A_1)$, $p(A_2)$, $p(A_3)$, $p(B_1)$, $p(B_2)$ を求めよ.
3. 複合事象のエントロピー $H(A, B)$ を計算せよ.
4. 事象 A のみのエントロピー $H(A)$ を計算せよ.
5. 事象 B のみのエントロピー $H(B)$ を計算せよ.
6. ある日の天気 (事象 A) が分かったという条件の下で, 出席率 (事象 B) に関する条件付き確率 $p_{A1}(B_1)$, $p_{A1}(B_2)$, $p_{A2}(B_1)$, $p_{A2}(B_2)$, $p_{A3}(B_1)$, $p_{A3}(B_2)$ を計算せよ.
7. ある日の天気が分かったという条件の下での, この講義の出席率に関する不確定さ, つまり条件付きエントロピー $H_A(B)$ を求めよ.
8. ある日の天気について知ることによって, この講義の出席率について得ることができる情報量, つまり相互情報量 $I(A, B)$ はいくらになるか?

第3章 情報の表現（数系）

計算機内部では、情報は最終的には離散的な量つまり数によって表されている．そこで、数をどのように表現するか？ということが問題となる．通常、私たちが使用している数の表現法（「数系」と呼ぶ）は10進数と呼ばれるものであるが、これは「位取り表現法」(Positional representation system)とよばれるものの一種であり、位取り表現法には10進数以外にも多くの数系が存在する．既に述べたように、情報量は最終的には2つの事象のうち1つが実現される場合の情報量に分解することができ、この時の情報量を情報量の単位としてbit（ビット）と呼んでいる．このため、計算機内部では2や2のべき乗に基づく位取り表現法がよく利用される．以下では、これらの数系について説明する．

3.1 位取り表現法

通常、自然数を表すために私たちが用いる数はどのように作られているか、を考えてみると、次のような構造になっていることが分かる．

$$\begin{aligned} 10 \text{ 進数 } 123 &\leftrightarrow 1 \times 100 + 2 \times 10 + 1 \\ &= 1 \times 10^2 + 2 \times 10^1 + 3 \times 10^0 \end{aligned}$$

つまり、

- 数を10の冪（べき）に分解する
- 各冪の係数だけを取り出して、順序を変えずに並べる

という操作を行っていると考えることができる．このとき、係数1, 2, 3の記されている場所を「位取り」(position)または「桁」(place)と言い、各桁には、それぞれ 10^2 , 10^1 , 10^0 という重み(significant)が加わっていると考えられる．このような数の表現法を「10進記数法」(10進法, decimal numeration system), この記数法で表された数字を「10進数」(decimal number)と呼ぶ．また、10をこの記数法の「基数」(radix)という．10進法では、各桁の数字(10進数字, decimal digit)は、0, 1, 2, $\dots$, 9のみである．

10進数を一般化して、任意の自然数 r ($r \geq 2$)による r 進法を考えることができる．

定義 6 (r 進法)

任意の自然数 a は、自然数 r ($r \geq 2$)を用いて、一意的に、

$$\begin{aligned} a = \sum_{k=0}^n a_k r^k &= a_n r^n + a_{n-1} r^{n-1} + \dots + a_0 r^0 \\ &\quad (a_i \in 0, 1, \dots, r-1) \end{aligned} \tag{3.1}$$

と表現することができる．このとき，係数 a_i だけを取り出して並べた

$$a_n a_{n-1} \cdots a_0$$

が， a の r 進数表現である．ただし，先頭の係数に 0 が連続する ($a_n = a_{n-1} = \cdots = a_{m+1} = 0$) 場合は，0 は省略して

$$a_m a_{m-1} \cdots a_0$$

と書き， $(m+1)$ 桁の r 進数と呼ぶ．他の記数法と区別したいときには，

$$a_m a_{m-1} \cdots a_0(r)$$

のように末尾に基数を明示する．

係数 a_i と a の関係は形式的には以下のように書くことができる．

$$a_i = \left\lceil \frac{a}{r^i} \right\rceil \bmod r \quad (3.2)$$

$\lceil x \rceil$: ガウスの記号， x を越えない最大の整数

$x \bmod y$: 剰余， x を y で割った余り

例えば， $a = 123$ ， $r = 10$ のときは次のようになる．

$$a_2 = \left\lceil \frac{123}{10^2} \right\rceil \bmod 10 = 1 \bmod 10 = 1$$

$$a_1 = \left\lceil \frac{123}{10^1} \right\rceil \bmod 10 = 12 \bmod 10 = 2$$

$$a_0 = \left\lceil \frac{123}{10^0} \right\rceil \bmod 10 = 123 \bmod 10 = 3$$

例 9 (記数法)

1. 2 進記数法 (binary numeration system)

$$a = \sum_{k=0}^n a_k 2^k, \quad (a_i \in \{0, 1\}) \quad (3.3)$$

2. 8 進記数法 (octal numeration system)

$$a = \sum_{k=0}^n a_k 8^k, \quad (a_i \in \{0, 1, \dots, 7\}) \quad (3.4)$$

3. 16 進記数法 (hexadecimal numeration system)

$$a = \sum_{k=0}^n a_k 16^k, \quad (a_i \in \{0, 1, \dots, 9, A, B, \dots, F\}) \quad (3.5)$$

(注) 数学で通常使用しているアラビア数字は元々 10 進法であるから，基底が 11 以上の記数法ではアラビア数字で表せない数字が出てくる．それらには適当な記号を割り当てることになっており，16 進法では伝統的にアルファベットの $A \sim F$ を使用している．

r 進数の四則演算は，基本的には 10 進数と変わらない．

1. 加算

$$\begin{array}{r} 1111_{(2)} \\ + 111_{(2)} \\ \hline 10110_{(2)} \end{array}$$

2. 減算

$$\begin{array}{r} 1010_{(2)} \\ - 111_{(2)} \\ \hline 11_{(2)} \end{array}$$

3. 乗算

$$\begin{array}{r} 1011_{(2)} \\ \times 101_{(2)} \\ \hline 1011_{(2)} \\ 1011_{(2)} \\ \hline 110111_{(2)} \end{array}$$

4. 除算

$$\begin{array}{r} 10_{(2)} \\ \hline 1010_{(2)} \overline{) 10111_{(2)}} \\ \underline{1010_{(2)}} \\ 11_{(2)} \end{array}$$

3.2 基数変換 (radix transformation)

ある記数法で表現された数を他の記数法に変換することを「基数変換」と呼ぶ．基数変換の具体的な計算法には様々なものがあるが，ここでは代表的なもののみを紹介する．

1. 10 進数から r 進数への変換

基本的には，10 進数を r 進数の冪乗の塊に分解してやれば良い．具体的には，下記に示すように 10 進数を r で繰り返し除算して余りと商を書き，最下段から順に商，余りを取って

並べる．

$$\begin{array}{r}
 2 \overline{) 123_{(10)}} \\
 2 \overline{) 61} \quad \dots \quad 1 \\
 2 \overline{) 30} \quad \dots \quad 1 \\
 2 \overline{) 15} \quad \dots \quad 0 \\
 2 \overline{) 7} \quad \dots \quad 1 \\
 2 \overline{) 3} \quad \dots \quad 1 \\
 1 \quad \dots \quad 1
 \end{array}$$

したがって， $123_{(10)} = 1111011_{(2)}$ となる．

2. r 進数から 10 進数への変換

r 進数の各桁に対して，対応する重みを 10 進数で表したものを掛けて総和を取ればよい．

$$\begin{aligned}
 1111011_{(2)} &= 1 \times 2^6 + 1 \times 2^5 + 1 \times 2^4 + 1 \times 2^3 + 0 \times 2^2 + 1 \times 2^1 + 1 \times 2^0 \\
 &= 64 + 32 + 16 + 8 + 2 + 1 \\
 &= 123_{(10)}
 \end{aligned} \tag{3.6}$$

3.3 負の数の表現法

前節までは自然数の r 進表現を扱ったが，負の数はどのように表すべきであろうか？ 1 つの方法は，通常私たちが 10 進数を手計算するときに行うように， $\pm$ の符号と数の絶対値を並べるやり方である．形式的に書けば以下のような形になる．

$$\begin{aligned}
 |a| &= \sum_{k=0}^n a_k r^k \quad (a_k \in \{0, 1, \dots, r-1\}) \\
 &\quad \downarrow \\
 a &= \pm \sum_{k=0}^n a_k r^k \\
 &= \pm a_n a_{n-1} \dots a_0_{(r)}
 \end{aligned}$$

しかし，この方法では通常の数他に新たに正負の符号を導入し，常に符号を考慮しながら計算を行わなければならない．計算機内部では数の正負に関わらず同じ手順で計算ができることが望ましいので，「符号 + 絶対値」の方法は少々都合が悪い．そこで，以下に示すような「補数」という考え方が導入されている．

定義 7 (補数表示 (complement representation))

1. r の補数

r 進法 n 桁の整数 a ($a > 0$) に対して,

$$a^c = r^n - a \quad (3.7)$$

を a に対する「 r の補数」と呼ぶ.

2. $r - 1$ の補数

r 進法 n 桁の整数 a ($a > 0$) に対して,

$$a^p = (r^n - 1) - a \quad (3.8)$$

を a に対する「 $r - 1$ の補数」と呼ぶ.

■

補数を用いると, 負の数 $a_{(r)}$ ($a_{(r)} < 0$) を以下のような形で表現することができる.

$$\begin{aligned} a_{(r)} &= -|a_{(r)}| \\ &= r^n - |a_{(r)}| - r^n \\ &= (r^n - |a_{(r)}|) - r^n \\ &= |a_{(r)}|^c - r^n \end{aligned} \quad (3.9)$$

あるいは

$$\begin{aligned} a_{(r)} &= -|a_{(r)}| \\ &= (r^n - 1) - |a_{(r)}| - (r^n - 1) \\ &= ((r^n - 1) - |a_{(r)}|) - (r^n - 1) \\ &= |a_{(r)}|^p - (r^n - 1) \end{aligned} \quad (3.10)$$

この式は, 負の数 $a_{(r)}$ を直接 r 進数で表して演算する代わりに, $a_{(r)}$ の絶対値の補数を用いて演算をしておいて, 最後に r^n または $r^n - 1$ を引くという補正演算を行っても同じ結果が得られることを示している.

また, 補数を用いると減算を加算に置き換えることができる.

$$\begin{aligned} a_{(r)} - b_{(r)} &= a_{(r)} + r^n - b_{(r)} - r^n \\ &= a_{(r)} + (r^n - b_{(r)}) - r^n \\ &= (a_{(r)} + b_{(r)}^c) - r^n \end{aligned} \quad (3.11)$$

あるいは

$$\begin{aligned} a_{(r)} - b_{(r)} &= a_{(r)} + (r^n - 1) - b_{(r)} - (r^n - 1) \\ &= a_{(r)} + ((r^n - 1) - b_{(r)}) - (r^n - 1) \\ &= (a_{(r)} + b_{(r)}^p) - (r^n - 1) \end{aligned} \quad (3.12)$$

計算機内部では, 減算器よりも加算器を構成する方が容易な場合が多いので, これも望ましい性質である.

3.4 r 進小数

前節までに整数の r 進表現を学んだので，本節では小数の表現を考えよう．整数の r 進表現が r の正の冪に対する係数のみから成るものであったのに対して，小数の r 進表現は r の負の冪に対する係数をも含むことになる．

定義 8 (r 進小数)

$$\begin{aligned} a &= \pm \left(\sum_{k=0}^n a_k r^k + \sum_{l=0}^m a_{-l} r^{-l} \right) \\ &= \pm (a_n r^n + a_{n-1} r^{n-1} + \cdots + a_0 r^0 + a_{-1} r^{-1} + \cdots + a_{-m} r^{-m}) \end{aligned} \quad (3.13)$$

から係数のみを取り出し，小数点を付け加えて並べた，

$$a_n a_{n-1} \cdots a_0 . a_{-1} \cdots a_{-m} \quad (3.14)$$

を a の r 進小数表現と呼び，小数点の左側を整数部，右側を小数部と呼ぶ． ■

例 10

$$\begin{aligned} 111.11_{(2)} &= 1 \times 2^2 + 1 \times 2^1 + 1 \times 2^0 + 1 \times 2^{-1} + 1 \times 2^{-2} \\ &= 4_{(10)} + 2_{(10)} + 1_{(10)} + \frac{1}{2}_{(10)} + \frac{1}{4}_{(10)} \\ &= 7.75_{(10)} \end{aligned}$$

r 進小数の基数変換を考える場合，整数部は整数の場合と同様であるが，小数部については負の冪の扱いに注意しなければならない．具体的には，以下の通りである．

1. 10 進小数から r 進小数への変換

「小数部を r 倍したとき，1 の位に a が立つならば，その小数には r^{-1} が a 個含まれる」という事実を利用する．具体的には，10 進数の小数部を r 倍してその小数部を取り出し再度 r 倍する，という操作を小数部がなくなるまで繰り返し，各段階ごとの商の 1 の位の数を並べ直すことによって r 進小数が得られる．

2. r 進小数から 10 進小数への変換

小数部の第 j 桁の数に重み r^{-j} を掛けて足しあわせれば良い．

例 11

- 10 進 $\rightarrow$ 2 進変換

$$\begin{array}{r}
 0.6875_{(10)} \\
 \times 2 \\
 \hline
 1.3750 \\
 0.3750 \\
 \times 2 \\
 \hline
 0.7500 \\
 0.7500 \\
 \times 2 \\
 \hline
 1.5000 \\
 0.5000 \\
 \times 2 \\
 \hline
 1.0000
 \end{array}
 \implies 0.6875_{(10)} = 0.1011_{(2)}$$

- 2 進 $\rightarrow$ 10 進変換

$$10111.1011_{(2)} = 16 + 4 + 2 + 1 + 0.5 + 0.125 + 0.0625 = 23.6875_{(10)}$$

3.5 固定小数点表現と浮動小数点表現

前節までは、位取り記数法の桁数については特に注意を払わなかった。しかし、実際のコンピュータ上では数は必ず有限の決まった桁数で表されなければならない。例えば、コンピュータのハードウェアが 2 進 8 桁しか扱えないとしよう。このとき、実数（当然、小数を含む）を表現する最も単純な方法は、適当な位置（例えば、4 桁と 5 桁の間）に小数点を固定し、小数点以上 4 桁、小数点以下 4 桁で実数を表現する方法である。これを、「固定小数点表示」と呼ぶ。

例 12 $3.1_{(10)}$ の 2 進 8 桁固定小数点表現

$$\begin{aligned}
 3.1_{(10)} &= 0011.0001100110011\dots_{(2)} \\
 &\approx 11.0010
 \end{aligned}$$

四捨五入するか、切り捨てるか、切り上げるか、は処理系による

固定小数点表現には、以下のような特徴がある。

- 処理が容易
- 表現できる数は、数直線上に等間隔で並ぶ
- 表現できる数の範囲（ダイナミックレンジ）が狭い

固定小数点表現の最大の問題点はダイナミックレンジが狭いことである．先程の例の2進小数点以上4桁，小数点以下4桁の場合，表現できる数の範囲は0から $1111.1111_{(2)} = 16.9375_{(10)}$ までとなる．この点を改良するために用いられるのが「浮動小数点表現」であり，実数 a を固定小数点数 f と r 進整数 r^e の積を用いて，

$$a = f \times r^e = (0.f_{-1}f_{-2}\cdots f_{-m})_{(r)} \times r^{(e_ne_{n-1}\cdots e_0)_{(r)}} \quad (3.15)$$

と考え， $f_{-1}\cdots f_{-m}$ と $e_ne_{n-1}\cdots e_0$ を並べて数表現する．このとき， f を仮数部， e を指数部と呼ぶ．

例 13 $3.1_{(10)}$ を2進8桁，仮数部5桁，指数部3桁の浮動小数点表現で表した場合

$$\begin{aligned} 3.1_{(10)} &= 11.0001100110011\cdots_{(2)} \\ &= 0.110001100110011\cdots \times 2^{010_{(2)}} \\ &\sim 0.11001_{(2)} \times 2^{010_{(2)}} \\ &\quad \downarrow \\ 3.1_{(10)} &= 11001010_{(2)} \end{aligned}$$

浮動小数点数には，

- 処理は多少複雑
- 表現できる数は，数直線上で非一様
- 表現できる数の範囲が広い（ただし，精度が良いわけではない）

という特徴がある．上の例で言えば，表現できる数の範囲は0から $0.1111_{(2)} \times 2^{111_{(2)}} = 61440_{(10)}$ となるので，同じ8ビットの固定小数点表現と比較して表現できる数の範囲ははるかに広いことが分かる．ただし，精度という点で言えば，固定小数点表現においては8ビットがすべて数の表現に用いられるのに対して，上の例の浮動小数点表現では数の表現に直接用いられるのは5ビットのみであるから，精度の点では浮動小数点の方が若干劣る．

演習問題

問2 以下の10進数を2進数に直せ．負の数については符号＋絶対値の方法で表現せよ．

1. $1998_{(10)}$
2. $27.725_{(10)}$
3. $-666_{(10)}$

問3 以下の2進数を10進数に直せ．負の数については符号＋絶対値の方法で表現せよ．

1. $100010011_{(2)}$
2. $10110.101_{(2)}$

3. $-11011_{(2)}$

問 4 以下の 2 進数の演算を実行せよ .

1. $1001011_{(2)} + 1100100_{(2)}$

2. $1011000_{(2)} - 100011_{(2)}$

3. $101111_{(2)} \times 1011_{(2)}$

4. $10011010_{(2)} \div 110_{(2)}$

問 5 補数について以下の問に答えよ .

1. $1998_{(10)}$ の 10 の補数を求めよ .

2. $1998_{(10)}$ の 9 の補数を求めよ .

3. $1010111_{(2)}$ の 2 の補数を求めよ .

4. $1010111_{(2)}$ の 1 の補数を求めよ .

問 6 $-4.75_{(10)}$ を仮数部 4 桁 (符号 1 桁 + 3 桁) , 指数部 4 桁 (符号 1 桁 + 3 桁) の 2 進浮動小数点数で表せ . ただし , 符号部分は + に 0 , - に 1 を割り当てるものとする .

第4章 情報の伝達（符号理論）

人間にとって意味のある情報というものは死蔵しておいては役に立たないので、何らかの意味で情報を伝達することが重要となる。これが「通信」である。しかし、情報が伝達されてゆく経路（「通信路」）には雑音や外乱が存在するのが普通であり、そのままでは、通信路に入力された情報と通信路から出力される情報には違いが生じてしまう。そこで、目的の情報をそのまま送信するのではなく、符号器というものを通して冗長（余分）な情報を付け加えてから送信することによって、雑音による誤りを除去しようという技術が開発されてきている。これが「誤り訂正符号」と呼ばれるものである¹。CDやMDと言ったデジタル・オーディオ機器の実現には、この誤り訂正符号が重要な役割を果たしている。CDの表面が少々汚れていても何の問題もなく音楽が再生されるのは、CDへの情報の記録に誤り訂正符号が用いられているからである。

本章では、誤り訂正符号としては最も簡単な「線形符号」の基礎を説明する（CDなどに用いられている符号はもう少し複雑なものである）。

4.1 通信系のモデル

ここで考える通信系はC.E.Shannonが提案したモデルに基づくものであり、図4.1に示すように情報源、符号器、通信路、復号器、宛先で構成されている。通信路は物理的には電線・無線・光ファイバ・CD-ROMなどさまざまな方法で実現されうるが、いずれにせよ必ず雑音を伴っており、通信路からの出力には誤りが含まれる。

情報源からは、情報源アルファベットと呼ばれる有限の記号を用いて表現された情報が出力される。例えば、デジタルコンピュータの場合、情報源アルファベットは0と1という2つの記号からなり、情報は001101010000111010...といった系列が出力される。これを「情報系列」あるいは「通報」と呼ぶ。ここでは、情報系列を、

$$i_1 i_2 i_3 i_4 \cdots, \quad i_j \in \{0, 1\} \quad (4.1)$$

という記号で一般的に表すことにしよう。

符号器は情報系列に冗長な情報を付加して新たな系列を生成するが、時間的に次々と連続して発生する情報系列を一度にまとめて処理することはできないので、まず情報系列を適当な長さ K 個ごとのブロックに分割する。

$$i_1 = [i_1 i_2 \cdots i_K], \quad i_2 = [i_{K+1} i_{K+2} \cdots i_{2K}], \quad \cdots \quad (4.2)$$

符号器は、各ブロック i_j ごとに冗長な情報を付加して新たな系列（「符号系列」） x を生成する。

$$x_1 = [x_1 x_2 \cdots x_N], \quad x_2 = [x_{N+1} x_{N+2} \cdots x_{2N}], \quad \cdots \quad (4.3)$$

¹ 誤り訂正符号の参考文献としては、例えば、今井秀樹：「符号理論」、コロナ社などがある。

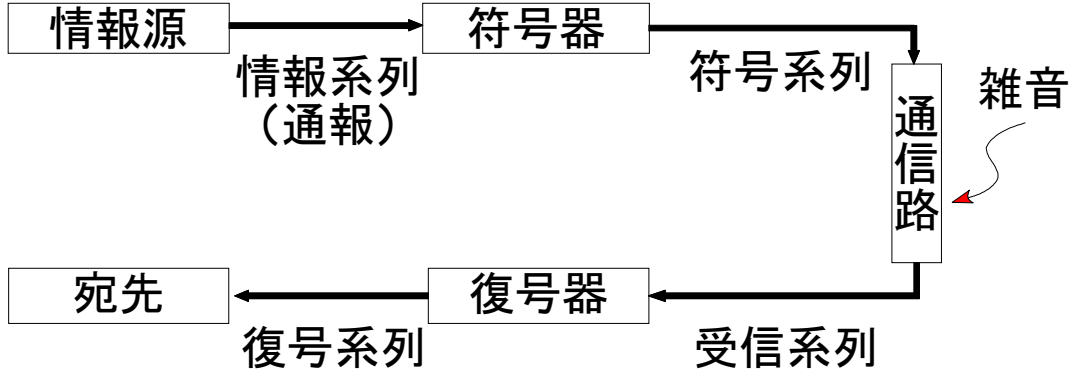


図 4.1: 通信系のモデル

ここで、 N を「符号長」と呼ぶ。なお、符号系列 x のアルファベットは情報系列 i のアルファベットと違っていても構わないが、ここでは簡単のために同じアルファベット (具体的には 0 と 1) を用いることにする。

通信路では雑音によって誤りが発生し、符号系列とは一部異なった受信系列

$$y_1 = [y_1 y_2 \cdots y_N], \quad y_2 = [y_{N+1} y_{N+2} \cdots y_{2N}], \quad \cdots \quad (4.4)$$

が出力される。復号器では、これに適当な演算を施して元の情報系列 i を復元する。本節では、通信路における誤りは「加法的」とであると仮定する。つまり、

$$\begin{aligned} \mathbf{y} &= [y_1 y_2 \cdots y_N] \\ &= [x_1 + e_1 \ x_2 + e_2 \ \cdots \ x_N + e_N] \\ &= \mathbf{x} + \mathbf{e} \end{aligned} \quad (4.5)$$

のように受信系列 $\mathbf{y}$ は符号系列 $\mathbf{x}$ と誤り $\mathbf{e}$ の和であると仮定する。ここで、 $e_i \in \{0, 1\}$ であり、 x_j と e_j の加算を、

$$\begin{aligned} 0 + 0 &= 0, & 0 + 1 &= 1 \\ 1 + 0 &= 1, & 1 + 1 &= 0 \end{aligned} \quad (4.6)$$

によって定義する。したがって、 $e_j = 0$ なら x_j は正しく受信され、 $e_j = 1$ なら x_j は誤って受信される。

1 ブロック (K ビット) の情報系列には、 $M = 2^K$ 通りの種類が存在する。このそれぞれに対して符号語を対応させると、その数は同じく $M = 2^K$ である。これらの符号語の集合 C を「ブロック符号」と呼ぶ。長さ N の符号語を送信してこれが正しく受信されれば、最大 K ビットの情報が伝送される。したがって、 $(N - K)$ ビットは冗長な情報であり、符号語 C は、

$$\frac{N - K}{N} \quad (4.7)$$

の「冗長度」を持つと言う。

4.2 Hamming 距離

符号の誤りを訂正するためには、各符号語を区別するための基準が必要である。そのために、符号語の間にある種の距離²を定義する。

定義 9 (Hamming 距離)

長さ N の 2 つのベクトル $\mathbf{u} = [u_1 u_2 \cdots u_N]$, $\mathbf{v} = [v_1 v_2 \cdots v_N]$ の間の距離を

$$d_H(\mathbf{u}, \mathbf{v}) = \sum_{i=1}^N \delta(u_i, v_i) \quad (4.8)$$

によって定義し、これを「Hamming 距離」と呼ぶ。ただし、

$$\delta(u, v) = \begin{cases} 0, & u = v \\ 1, & u \neq v \end{cases} \quad (4.9)$$

■

例 14 $\mathbf{u} = [011]$, $\mathbf{v} = [110]$ のとき

$$d_H(\mathbf{u}, \mathbf{v}) = \delta(0, 1) + \delta(1, 1) + \delta(1, 0) = 1 + 0 + 1 = 2 \quad (4.10)$$

である。これは、各ベクトルを 3 次元空間の位置ベクトルと考えると、大きさ 1 の立方体の頂点 (011) と (110) の間の辺にそった距離と考えられる。

距離が定義できれば符号語自身の「大きさ」に相当するものも定義できる。これを「Hamming 重み」と呼ぶ。

定義 10 (Hamming 重み)

長さ N のベクトル $\mathbf{u}$ の 0 でない成分の数を $\mathbf{u}$ の「Hamming 重み」と呼び、 $w_H(\mathbf{u})$ で表す。

$$w_H(\mathbf{u}) = \sum_{i=1}^N \delta(u_i, 0) \quad (4.11)$$

と考えて良いから、

$$w_H(\mathbf{u}) = d_H(\mathbf{u}, \mathbf{0}) \quad (4.12)$$

と表すことができ、また逆に、

$$d_H(\mathbf{u}, \mathbf{v}) = w_H(\mathbf{u} - \mathbf{v}) \quad (4.13)$$

と表すこともできる。 ■

符号語の集合である符号 C を 1 つ定めると、符号 C の中にどの程度の近さで符号語が詰め込まれているかを測ることができる。

²数学においては、距離の 3 公準と呼ばれる性質をもつものをすべて「距離」と呼ぶ。私たちが日常使用している「ユークリッド距離」はその 1 例に過ぎない。

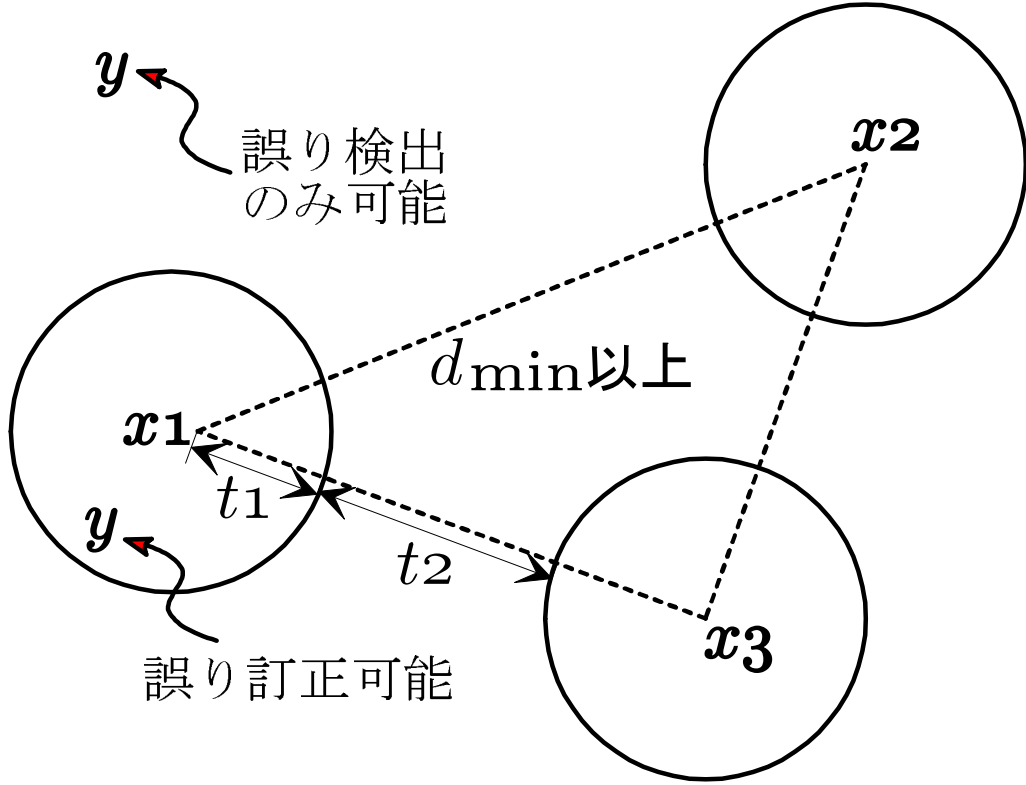


図 4.2: 誤り訂正の原理

定義 11 (最小距離)

符号 C の異なる 2 つの符号語間の Hamming 距離の最小値を符号 C の「最小 (Hamming) 距離」と呼び、 $d_{\min}$ で表す。

$$d_{\min} = \min_{u \neq v, u, v \in C} d_H(u, v) \quad (4.14)$$

■

ある符号語 x が受信語 y として受信された場合を考えると、

$$d_H(y, x) = d_H(x + e, x) = w_H(x + e - x) = w_H(e) \quad (4.15)$$

であり、 $w_H(e)$ は誤りの個数に他ならないから、通信路で t 個の誤りが発生すると、符号語と受信語は Hamming 距離で測って t だけ遠ざかることになる。

4.3 誤り検出・訂正の原理

前節で符号語間の距離が定義され、通信路における誤りとの関係が明らかになったので、これを用いて「誤り検出・誤り訂正」の原理を述べることができる。

最小距離 $d_{\min}$ を持つ符号 C を考えよう．符号には冗長性を持たせてあるので $d_{\min} > 0$ であり，各符号語の間には Hamming 距離の意味ですきまが空いている．今，通信路で 1 ブロック当たり t 個の誤りが発生したとすると，受信語 y は元の符号語 x から Hamming 距離で t だけ離れる．そこで，各符号語を中心に半径 t_1 ($d_{\min} \leq 2t_1 + 1$ とする) の重複しない球を考え，球内に入った受信語 y に対してはその球の中心の符号が送られたと推定 (誤り訂正) する．どの球にも入らない受信語は，ただ単に誤りと判定 (誤り検出) する．これが「限界距離復号法」と呼ばれる誤り検出・訂正法の原理である． t_1 の設定には自由度があるが，その最大値は，

$$t_{1\max} = \left\lceil \frac{d_{\min} - 1}{2} \right\rceil \quad (4.16)$$

によって制限される．

4.4 線形符号

誤り検出・訂正の原理が明らかになったので，具体的に誤り訂正符号を構成する方法を考えよう．ここで考える符号は，情報系列 (情報ビット) $i_1 i_2 \cdots i_K$ に対して単純に検査記号 (検査ビット) $p_1 p_2 \cdots p_{N-K}$ を付け加えて符号語を構成するものである．このような符号を「組織符号」 (systematic code) と呼ぶ．

$$\begin{aligned} \mathbf{i} &= [i_1 i_2 \cdots i_K] \\ \mathbf{x} &= [i_1 i_2 \cdots i_K p_1 p_2 \cdots p_{N-K}] \end{aligned}$$

以下では，情報ビット K ，符号長 N の組織符号を「 (N, K) 符号」と書くことにする．

情報ビットから検査ビットを構成するには多様な方法が有り得るが，実際によく用いられているのは検査ビットが情報ビットの線形な関数³となっているものである．これを「線形符号」と呼ぶ．線形符号では， p_{ij} を定数として，

$$\begin{aligned} p_1 &= p_{11} i_1 + p_{12} i_2 + \cdots + p_{1K} i_K \\ p_2 &= p_{21} i_1 + p_{22} i_2 + \cdots + p_{2K} i_K \\ &\vdots \\ p_{N-K} &= p_{N-K1} i_1 + p_{N-K2} i_2 + \cdots + p_{N-KK} i_K \end{aligned} \quad (4.17)$$

のように検査ビットを計算する．これを行列形式で書けば，以下ようになる．

$$[p_1 p_2 \cdots p_{N-K}] = [i_1 i_2 \cdots i_K] \begin{bmatrix} p_{11} & p_{12} & \cdots & p_{1K} \\ p_{21} & p_{22} & \cdots & p_{2K} \\ \vdots & & & \vdots \\ p_{N-K1} & p_{N-K2} & \cdots & p_{N-KK} \end{bmatrix} \quad (4.18)$$

簡単のために，これを $\mathbf{p} = \mathbf{iP}$ と書く．

³定数倍と加減算のみから成る関数．ただし，ここでの演算は，以前定義したように 0 と 1 だけから成るものである．

検査ビットだけではなく，符号語全体を行列形式で表してやることもできる．

$$\begin{aligned}
 x &= [i p] \\
 &= [i I i P] \\
 &= i [I P] \\
 &= i \begin{bmatrix} 1 & 0 & \cdots & 0 & p_{11} & p_{21} & \cdots & p_{N-K 1} \\ 0 & 1 & \cdots & 0 & p_{11} & p_{21} & \cdots & p_{N-K 1} \\ & & \ddots & & \vdots & & & \vdots \\ 0 & 0 & \cdots & 1 & p_{1K} & p_{2K} & \cdots & p_{N-K K} \end{bmatrix} \\
 &\equiv i G
 \end{aligned} \tag{4.19}$$

このとき，情報ビット i から符号語 x を作り出すという意味で行列 G を「生成行列」(generation matrix) と呼ぶ．なお， I は単位行列である．

一方，式 (4.17) は，

$$\begin{aligned}
 p_{11} i_1 + p_{12} i_2 + \cdots + p_{1K} i_K + p_1 &= 0 \\
 p_{21} i_1 + p_{22} i_2 + \cdots + p_{2K} i_K + p_2 &= 0 \\
 &\vdots \\
 p_{N-K 1} i_1 + p_{N-K 2} i_2 + \cdots + p_{N-K K} i_K + p_{N-K} &= 0
 \end{aligned} \tag{4.20}$$

のように変形することができ，これを行列形式で書き表せば以下ようになる．

$$x H^t = 0 \tag{4.21}$$

$$H = \begin{bmatrix} p_{11} & p_{12} & \cdots & p_{1K} & 1 & 0 & \cdots & 0 \\ p_{21} & p_{22} & \cdots & p_{2K} & 0 & 1 & & 0 \\ \vdots & & & & \vdots & & \ddots & \vdots \\ p_{N-K 1} & p_{N-K 2} & \cdots & p_{N-K K} & 0 & \cdots & 0 & 1 \end{bmatrix} \tag{4.22}$$

この式は，符号語 x が符号 C に属しているならば $x H^t = 0$ となることを表しているから，任意のベクトル y に対して，

$$\begin{cases} y H^t = 0 & \rightarrow y \text{ は } C \text{ の符号語である} \\ y H^t \neq 0 & \rightarrow y \text{ は } C \text{ の符号語でない} \end{cases} \tag{4.23}$$

のように y が C に属するかどうかをテストすることができる．このことから，行列 H を「検査行列 (あるいはパリティ検査行列)」(parity check matrix) と呼ぶ．

4.5 線形符号の復号

線形符号 C の符号語 x を送信し， $y = x + e$ を受信したとする． y を検査するためには，前節で述べたように

$$s = y H^t \tag{4.24}$$

を計算する．これを y に対する「シンドローム」(syndrome) と呼ぶ．

$s = 0$ であれば，誤りはなく ($e = 0$)， y をそのまま復元符号と考えてよい． $s \neq 0$ のときには，

$$\begin{aligned} s &= yH^t \\ &= (x + e)H^t \\ &= xH^t + eH^t \\ &= eH^t \end{aligned} \quad (4.25)$$

のようにシンドローム s は 誤り e だけで決まり，情報ビット x には依存しない．従って，訂正可能な誤りのパターンに対してあらかじめシンドローム s を計算して両者の対応表を作成しておけば，シンドローム s から誤り e を求め，これを用いて受信符号 y を訂正して x を復号することができる．

4.6 単一パリティ検査記号

以上述べてきた線形符号の最も簡単な例が「単一パリティ符号」である．単一パリティ検査符号では，

$$p_1 = i_1 + i_2 + \cdots + i_K \quad (4.26)$$

によって計算される 1 つの検査ビットのみを使用する．従って，情報系列のブロック長を K とすれば，単一パリティ検査符号は $(K + 1, K)$ 符号ということになる．

例 15 ($K = 3$ の場合) ブロック長が 3 ビットであるので情報系列としては，

$$i \in \{(000), (001), (010), (011), (100), (101), (110), (111)\} \quad (4.27)$$

の 8 種類が可能である．これに対して生成行列は，

$$G = \begin{bmatrix} 1 & 0 & 0 & 1 \\ 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 1 \end{bmatrix} \quad (4.28)$$

となる．これから符号語として可能なものを計算すると，以下ようになる．

$$x \in \{(0000), (0011), (0101), (0110), (1001), (1010), (1100), (1111)\} \quad (4.29)$$

この符号語の最小距離は，

$$d_{\min} = 2 \quad (4.30)$$

であり，従って，

$$t_1 = \lceil \frac{d_{\min} - 1}{2} \rceil = 0, \quad t_2 = d_{\min} - t_1 = 2 \quad (4.31)$$

となるので，この符号は誤り検出可能であるが，誤りの訂正はできない．検査行列は，

$$H = [1 \ 1 \ 1 \ 1] \quad (4.32)$$

である． $t_2 = 2$ より検出可能な誤りは符号語当たり 1 個であるので，可能な誤りのパターンは次の 4 個である．

$$e \in \{(0001), (0010), (0100), (1000)\} \quad (4.33)$$

これから，誤りパターンに対するシンドロームは，

$$s = eH^t \in \{(1), (1), (1), (1)\} \quad (4.34)$$

となる．

可能な x の形をよく眺めれば分かるように，単一パリティ検査符号では符号語中の 1 の数が偶数に固定されている．一方，受信語に対するシンドロームの計算は受信語中のすべての要素の和を取っているので（シンドロームが 0）→（誤り無し）（シンドロームが 1）→（誤り有り）のように容易に誤り検出が可能である．

4.7 Hamming 符号

単一パリティ検査符号では，誤りの検出は可能だが誤りの訂正はできなかった．線形符号において，誤り訂正の可能な最も簡単な符号としては，以下に述べる Hamming 符号がある．

定義 12 (Hamming 符号)

線形符号 (N, K) を考え， $M = N - K$ としたとき，0 でない M 次元列ベクトルをすべて並べて検査行列 H を作る．この H によって決まる符号を「Hamming 符号」と呼ぶ． ■

例 16 ($M = 3$ の場合) 検査行列は，

$$H = \begin{bmatrix} 1 & 1 & 1 & 0 & 1 & 0 & 0 \\ 0 & 1 & 1 & 1 & 0 & 1 & 0 \\ 1 & 1 & 0 & 1 & 0 & 0 & 1 \end{bmatrix} \quad (4.35)$$

であり，従って，この符号は $(7, 4)$ 符号になる．一方，生成行列は以下の通り．

$$G = \begin{bmatrix} 1 & 0 & 0 & 0 & 1 & 0 & 1 \\ 0 & 1 & 0 & 0 & 1 & 1 & 1 \\ 0 & 0 & 1 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 1 & 1 \end{bmatrix} \quad (4.36)$$

情報系列のブロック長は 4 ビットであるので，可能な系列は 16 通りである．

$$\begin{aligned} i \in \{ & (0000), (0001), (0010), (0011), (0100), (0101), (0110), (0111), \\ & (1000), (1001), (1010), (1011), (1100), (1101), (1110), (1111) \} \end{aligned} \quad (4.37)$$

これを生成行列 G に掛けて符号語を求めると，

$$\begin{aligned} x \in \{ & (0000000), (0001011), (0010110), (0011101), (0100111), \\ & (0101100), (0110001), (0111010), (1000101), (1001110), \\ & (1010011), (1011000), (1100010), (1101001), (1110100), (1111111) \} \end{aligned} \quad (4.38)$$

となる．計算の詳細は省略するが最小距離は，

$$d_{\min} = 3 \quad (4.39)$$

であり，Hamming 符号は符号語当たり 1 個の誤りを訂正可能である．

(注) 以上述べてきたような誤り訂正符号では，誤りは適当に散らばって発生する(「ランダム誤り」などと言う)ことを前提にしている．これは誤りの原因が雑音などの場合には妥当な仮定である．しかし，CDの表面に傷がついて発生する誤りなどを考えると，誤りは一度にまとまって発生する場合が多い．これを「バースト誤り」などと呼ぶ．バースト誤りの場合には，いくつもの符号語にまたがって連続して誤りが生じるので，符号語単位で誤り訂正を行う方法では誤りを訂正することができなくなる．CDなどに実際に使用されている符号化法ではバースト誤りに対する対策が行われているが，それでもCDのトラック方向(円周方向)にそって長い傷が付いたような場合には十分な誤り訂正が不可能になる．一方，CDの半径方向に傷が付いた場合は，傷自体は大きくとも符号語からみればランダム誤りとみなせるので，誤り訂正は容易である．

4.8 補遺

通信路における雑音の影響のところでも述べたように，このテキストで扱う通信路では数として 0 と 1 だけを用い， $1 + 1 = 0$ となる演算 (2 を法とする剰余演算) を用いている．このように，数学においては，

数の集合 + 演算 $\rightarrow$ 代数的構造

という考え方をしており，数の集合や演算の定義を変えることによって多様な代数的構造を考えることができる．以下に代数的構造の定義を 2 つ示す．これら以外にも多くの代数的構造が存在する．

定義 13 (群 (group))

集合 A がある演算 (便宜上，加算と呼び， $+$ で表す) に対し以下の 3 つの性質を持つとき， A は $+$ に関して「群」を成すと言う．

1. A の任意の元 a, b, c に対して結合則

$$(a + b) + c = a + (b + c) \quad (4.40)$$

が成り立つ．

2. A の元の中に以下の性質を持つ元 (単位元) e が存在する．

$$a + e = e + a = a \quad (4.41)$$

3. A の任意の元 a に対して以下の性質を満たす元 (逆元) a^{-1} が存在する．

$$a + a^{-1} = a^{-1} + a = e \quad (4.42)$$

■

定義 14 (体 (field))

集合 A に対して加算 $+$ と乗算 $\cdot$ が定義できて、以下の性質を満たすとき、 A は $+$ と $\cdot$ に関して「体」を成すと言う。

1. A は加算 $+$ に関して群を成す。
2. A から加算の単位元 e を除いた集合は、乗算 $\cdot$ に関して群を成す。

■

以下の例に述べるように、符号理論は実は代数的構造に密接に関わっており、このテキストで述べた事柄を超える範囲の符号理論を学ぶためには代数学の知識が必要となる。

例 17 通信路に関する演算は、

- 集合 $A = \{0, 1\}$
- 加算 $(a + b) \bmod 2$ (2 を法とする剰余演算)
- 乗算 $(a \cdot b) \bmod 2$ (2 を法とする剰余演算)

という条件で体を成している。これを「2 を法とするガロア体」と呼び、 $GF(2)$ で表す。

演習問題

問 7

1. 情報ベクトル i の長さが 7 のときの、単一パリティ検査符号の生成行列 G を書け。
2. 情報ベクトル $i = [1010101]$ に対する符号ベクトル x を求めよ。
3. 単一パリティ検査符号の検査行列 H を書け。
4. 受信ベクトルが $y = [10101011]$ であったとする。シンδροーム s を計算せよ。
5. 4. において情報ベクトルは正しく伝送されただろうか？

問 8 Hamming 符号について、次の問いに答えよ。

1. 符号長 $N = 7$ 、情報ビット長 $K = 4$ の Hamming 符号の冗長度はいくらか？
2. $N = 7$ 、 $K = 4$ の Hamming 符号の生成行列 G を示せ。
3. 情報ベクトル $i = [0110]$ に対する伝送符号 x を求めよ。
4. 2. で求めた生成行列 G に対応する検査行列 H を示せ。
5. 受信符号が $y = [0111011]$ であったとする。 y に対するシンδροームを計算せよ。
6. 4. の結果から考えて、 y は正しく伝送されただろうか、それとも誤って伝送されたのだろうか？

問 9 集合 $\{0, 1, 2\}$ に, 3 を法とする剰余演算を用いて加算と乗算を定義したい.

1. 加算を表現する表を描け.
2. 乗算を表現する表を描け.
3. この集合は 3 を法とする演算のもとで体になるだろうか? 考察せよ.

第5章 集合と論理

前章までは、情報に関する数学の基礎を学んだ。本章からは、情報を処理するデジタルコンピュータ内部の働きに関する数学の基礎を学ぶ。コンピュータ内部では情報に対して様々な論理演算を施すが、これらの基礎となるのは「集合論」「記号論理学」「ブール代数」といった数学である。本章では、まず集合と記号論理について述べる。

5.1 集合

5.1.1 集合の定義

数学においては、「相異なる (distinct) 対象 (object) の集まり」を「集合」(set) と呼ぶ。「相異なる」という形容詞の意味するところは、何であるか明確に区別できないような「ぬえ」的な対象に対してはうまく集合を定義できない、ということである。

集合 A を構成する対象のことを「要素」(element) あるいは「元」(member) と呼ぶ。ある要素 p が集合 A を構成する対象であることを $p \in A$ と記し、また p が A の要素ではないことを $p \notin A$ と書く。

具体的にある集合を指定するためには、以下の2つの方法がある。

1. $A = \{a, e, i, u\}$ のように要素を列記する。
2. $A = \{x | x \text{ は偶数でかつ } x > 0\}$ のように、縦棒の左に要素を代表する適当な変数名、右にその変数の満たすべき性質を書く。

数学では、1つの要素も含まない集合も集合として扱い「空集合」 ϕ と呼ぶ。 $\phi = \{\}$ である。また、2つの集合 P と Q が与えられたとき、 P のすべての要素が Q の要素でもあるならば、「 P は Q の部分集合である」と言い、 $P \subseteq Q$ で表す。特に、

1. P は P 自身の部分集合である
2. 空集合 ϕ は、任意の集合の部分集合である

という性質が成り立つ。

2つの集合 P と Q が与えられたとき、 P と Q がまったく同じ要素を含んでいるとき、そのときにのみ「 P と Q は等しい」と言い、 $P = Q$ と書く。 $P = Q$ は $P \subseteq Q$ かつ $Q \subseteq P$ と同じ意味である。また、 P が Q の部分集合でかつ P が Q に等しくないとき、つまり P に含まれない Q の要素が少なくとも1つは存在するとき、「 P は Q の真部分集合である」と言い、 $P \subset Q$ で表す。

5.1.2 集合の演算

集合に対してもある種の演算を定義することができる。

- 集合の和 (union): $S \cup T$
集合 S, T のどちらか一方あるいは両方に含まれる要素からなる集合 .
- 集合の積 (交わり , intersection): $S \cap T$
集合 S, T の両方に含まれている要素から成る集合 .
(注) 交わりが空集合となる2つの集合を「互いに素」(disjoint) という .
- 集合の差 (difference): $S - T$
 T に含まれない S の要素から成る集合 .
- 補集合 (complement): $\overline{S}$
集合 S がより大きな集合 U の部分集合である場合 , $U - S$ を「 U に関する S の補集合」
と呼び , $\overline{S}$ で表す . 特に述べなくとも U が明らかであるなら , 単に「 S の補集合」と言う .
このとき , U は普遍集合 (全体集合 , universe) と呼ばれる .
- 対称差 (symmetric deifference): $S \oplus T$
 S または T には含まれるが両方には含まれていない要素だけから成る集合 . $(S \cup T) - (S \cap T)$
に等しい .
- べき集合 (power set): $P(S)$
集合 S のすべての部分集合を要素とする集合 .

これらの演算には多くの法則があるが , 基本的なものを以下に示す .

1. べき等律 (Idempotent Laws)

$$A \cup A = A, \quad A \cap A = A$$

2. 交換律 (Commutative Laws)

$$A \cup B = B \cup A, \quad A \cap B = B \cap A$$

3. 結合律 (Associative Laws)

$$(A \cup B) \cup C = A \cup (B \cup C), \quad (A \cap B) \cap C = A \cap (B \cap C)$$

4. 分配律 (Distributive Laws)

$$A \cup (B \cap C) = (A \cup B) \cap (A \cup C), \quad A \cap (B \cup C) = (A \cap B) \cup (A \cap C)$$

5. 同一律 (Identity Laws)

$$A \cup \phi = A, \quad A \cap U = A, \quad A \cup U = U, \quad A \cap \phi = \phi$$

6. 対合律 (Involution Laws)

$$\overline{(\overline{A})} = A$$

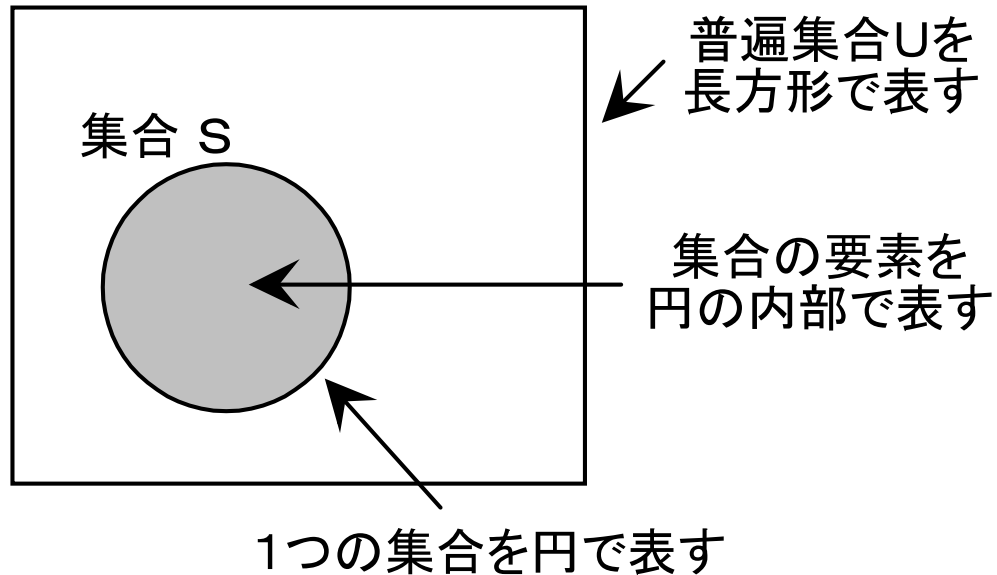


図 5.1: Venn 図

7. 補元律 (Complement Laws)

$$A \cup \bar{A} = U, \quad A \cap \bar{A} = \phi, \quad \bar{\bar{U}} = \phi, \quad \bar{\phi} = U$$

8. ド・モルガンの法則 (DeMorgan's Laws)

$$\overline{(A \cup B)} = \bar{A} \cap \bar{B}, \quad \overline{(A \cap B)} = \bar{A} \cup \bar{B}$$

集合の演算を視覚的に表す方法としては、図-5.1 のようなベン図 (Venn diagrams) がある。

5.1.3 有限集合と無限集合

我々が日常的に考える集合は有限個の要素から成るものがほとんどであるが、「整数の集合」などを考えれば分かるように無限個の要素から成る集合も考えることができる。前者を「有限集合」、後者を「無限集合」と呼ぶ。無限集合はさらに「可算無限集合」と「非可算無限集合」に分類される。これらを数学的に明確に定義するためには、そもそも「集合の大きさ」とは何かということとを定義しておかなければならない。

定義 15 (集合の濃度)

1. 集合 S と T の要素の間に 1 対 1 対応が成り立つとき、 S と T は同じ大きさ (有限集合の場合「個数」(cardinality)、無限集合の場合「濃度」と呼ぶ) を持つと定義する。
2. 空集合のみを用いて、以下のように自然数を定義する。

- (a) 空集合 ϕ に 0 という名前を付ける． $0 = \phi$
 - (b) 0 のみを要素として持つ集合を考え，これに 1 という名前を付ける． $1 = \{0\} = \{\phi\}$
 - (c) 0 と 1 のみから成る集合を考え，これに 2 という名前を付ける． $2 = \{0, 1\} = \{\phi, \{\phi\}\}$
 - (d) 以下同様にして， $3, 4, \dots, n, \dots$ という集合を考える．これによって自然数が定義される．
 - (e) これらの集合全体の集合を N と置く．これは自然数全体を表している．
3. 集合 S の要素が集合 $n \in N$ の要素と 1 対 1 対応が付くとき S は有限集合であり， S の濃度を n と定義し $|S| = n$ と書く．
4. ある集合 S の要素が $n \in N$ の要素と 1 対 1 対応が付かないとき， S は無限集合である．
5. N 自身も（可算）無限集合であり，その濃度を「アレフゼロ」（アレフはヘブライ語のアルファであるが，このテキストを書いているワープロのフォントにないので，ここではカタカナで書いておく）と呼ぶ．
6. では無限集合は可算無限集合だけなのであろうか？実はもっと濃度の高い集合があり，非可算無限集合と呼ばれている．

■

非可算無限集合の存在を証明するためには，ドイツの数学者カントールの考えた「対角線論法」と呼ばれるものが用いられる¹．

定理 1 (対角線論法)

$[0, 1)$ 区間の実数を考える．この区間の実数全体の集合 $R[0, 1)$ が可算無限集合であると仮定すると，

$$\begin{aligned}
 p_1 &= 0.a_{11}a_{12}a_{13}a_{14}\cdots a_{1n}\cdots \\
 p_2 &= 0.a_{21}a_{22}a_{23}a_{24}\cdots a_{2n}\cdots \\
 &\vdots \\
 p_n &= 0.a_{n1}a_{n2}a_{n3}a_{n4}\cdots a_{nn}\cdots \\
 &\vdots
 \end{aligned}$$

(a_{ij} は 0 から 9 までの整数, $0 \leq a_{ij} \leq 9$)

のように，すべての実数に $p_1, p_2, p_3, \dots$ と番号を振ることができるはずである．ここで，次のように新しい数を作る．

$$\begin{aligned}
 q &= 0.b_1b_2\cdots b_n\cdots \\
 b_i &= \begin{cases} 1, & a_{ii} = 9 \text{ のとき} \\ 9, & a_{ii} = 0 \sim 8 \text{ のとき} \end{cases}
 \end{aligned}$$

この数 q は区間 $[0, 1)$ にあり，各桁に 0 を含まないから必ず無限小数になる．しかも定義から，

$$q \neq p_n \quad \text{任意の } n \text{ について}$$

このことは， $R[0, 1)$ のすべての数に番号が振れる，つまり $R[0, 1)$ が可算無限集合であることに矛盾する．したがって， $R[0, 1)$ は可算無限集合よりも濃度の高い集合（非可算無限集合）である．

■

¹参考文献，竹内 外史：新装版 集合とは何か，講談社ブルーバックス

5.1.4 数学的帰納法

集合の要素が有限であるときには、その集合について成り立つ性質を証明するには、各要素毎に性質を証明してやることができる。しかし、無限集合に対しては当然そのような方法で証明を行うことはできない。このような場合によく用いられる方法が、「数学的帰納法」(Principle of mathematical induction)である。数学的帰納法には、以下に示すような2種類が存在する。

● 数学的帰納法

1. $n = n_0$ に対して、ある陳述が真である。
2. $n = k$ ($n \geq n_0$) に対してその陳述が真であるという仮定のもとで、 $n = k + 1$ に対してその陳述が真である。
- ↓
3. $n \geq n_0$ なるすべての n に対して、その陳述が真である。

● 強数学的帰納法

1. $n = n_0$ に対して、ある陳述が真である。
2. $n_0 \leq n \leq k$ に対してその陳述が真であるという仮定のもとで、 $n = k + 1$ に対してその陳述が真である。
- ↓
3. $n \geq n_0$ なるすべての n に対して、その陳述が真である。

いくつか例を挙げよう。

例 18 3 円と 5 円の 2 種類の切手があれば、8 円以上のすべての郵便料金を払うことができることを証明したい。

1. 8 円の郵便料金は、3 円切手と 5 円切手で払うことができる。
2. k 円 ($k \geq 8$) の料金を払うことができると仮定する。このとき、5 円の切手を 1 枚以上使っているならば、1 枚の 5 円切手を 2 枚の 3 円切手で置き換えることによって $(k + 1)$ 円払うことができる。5 円の切手を 1 枚も使っていないとすれば、少なくとも 3 枚以上の 3 円切手を使っているはずであり、3 枚の 3 円切手を 2 枚の 5 円切手に置き換えることによって $(k + 1)$ 円払える。
3. したがって、3 円と 5 円の切手を用いて 8 円以上のすべての郵便料金を払うことができる。

例 19 2 以上の任意の正整数は、素数であるかあるいは素数の積であることを証明せよ。

1. 2 は素数である。
2. $2 \leq n \leq k$ の任意の正整数は素数か素数の積であると仮定する。 $(k + 1)$ は素数であるか素数でないかのどちらかであり、素数であれば証明終わり。素数でない場合は、 $k + 1 = p \cdot q$ のように 2 つの正整数の積で表すことができ、しかも $2 \leq p, q \leq k$ から p, q は素数あるいは素数の積であり、したがって $k + 1$ も素数の積である。
3. よって、2 以上の任意の正整数は素数か素数の積であることが証明された。

演習問題

問 10 200 人の学生のうち、50 人が離散数学の講義を受講し、140 人が経済学の講義を受講し、24 人が両方の講義を受講しているものとする。双方の講義とも明日試験を予定しているので、どちらの講義も受講していない学生だけが今夜コンパに出かけることができる。全学生をあらわす集合を全体集合 U 、離散数学を受講している学生をあらわす集合を A 、経済学を受講している学生をあらわす集合を B と書くことにする。

1. この状況をベン図を用いて表せ。
2. コンパに出かけることのできる学生の集合を、 A 、 B および集合演算の記号を用いて表せ。
3. コンパに出かけることのできる学生の数は何人か？

問 11 $n \geq 1$ のすべての整数について、 $2^n \times 2^n - 1$ が 3 で割り切れることを、数学的帰納法を用いて証明せよ。

第6章 ブール代数と基本論理回路

集合の章で学んだ集合の和や交わり，あるいは論理における「かつ」(and)と「または」(or)などは，数学的に見るといずれも「ブール代数」という同じ構造を持っている．デジタル計算機は0と1だけからなる符号に対して種々の操作を行うものであるから，ブール代数は計算機ハードウェアの基礎でもある．本章では，ブール代数の定義とブール代数を図形的に扱うために提案されたMIL記号について説明する．

6.1 ブール代数

ブール代数は形式的には以下のように定義される．

定義 16 (ブール代数)

空でない集合 B に，2つの異なる要素 0 と 1，および2項演算 $+$ と $\cdot$ ，単項演算 $\overline{}$ を定義することができ，これらが B の任意の要素 a, b, c に対して以下の性質を満足するとき， B はブール代数と呼ばれる．

1. 交換律 $a + b = b + a, \quad a \cdot b = b \cdot a$
2. 同一律 $a + 0 = a, \quad a \cdot 1 = a$
3. 補充律 $a + \bar{a} = 1, \quad a \cdot \bar{a} = 0$

このとき，0 を零元 (zero)，1 を単位元 (unit)， $\bar{a}$ を a の補元 (complement)， $+$ を和， $\cdot$ を積， $\overline{}$ を補 (演算) などと呼ぶ．集合 B がブール代数であることを強調するときには，これらをまとめて $\langle B, +, \cdot, \overline{}, 0, 1 \rangle$ などと書く． ■

ブール代数の代表的な例は，符号理論の章で現れた 0 と 1 だけからなる代数系であり，これはデジタル計算機内部で行われている演算そのものである．

例 20 (ブール代数) 集合 $B = \{0, 1\}$ に対して，以下のように演算を定義するとブール代数となる．

$$\begin{aligned} \bar{0} &= 1, & \bar{1} &= 0 \\ 0 + 0 &= 0, & 0 + 1 &= 1, & 1 + 0 &= 1, & 1 + 1 &= 1 \\ 0 \cdot 0 &= 0, & 0 \cdot 1 &= 0, & 1 \cdot 0 &= 0, & 1 \cdot 1 &= 1 \end{aligned}$$

また，1 を論理的に正しい (真, true)，0 を論理的に誤り (偽, false) のように読み換えると， $\overline{}$ は「否定」， $+$ は「または (or)」， $\cdot$ は「かつ (and)」という論理を表していると解釈することもできる．

ブール代数の基本的な定理には以下のものがある．

定理 2 (ブール代数)

1. べき等律 $a + a = a, \quad a \cdot a = a$
2. 有界律 $a + 1 = 1, \quad a \cdot 0 = 0$
3. 吸収律 $a + (a \cdot b) = a, \quad a \cdot (a + b) = a$
4. 結合律 $(a + b) + c = a + (b + c), \quad (a \cdot b) \cdot c = a \cdot (b \cdot c)$
5. 補元の一意性 $a + x = 1$ かつ $a \cdot x = 0$ ならば $x = \bar{a}$
6. 対合律 $\bar{\bar{a}} = a$
7. $\bar{0} = 1, \quad \bar{1} = 0$
8. ド・モルガンの法則 $\overline{a + b} = \bar{a} \cdot \bar{b} \quad \overline{a \cdot b} = \bar{a} + \bar{b}$

■

なお定理の証明には、以下に述べる「双対性の原理」が有用である。

定理 3 (双対性の原理)

ブール代数 B において、元の式の $+$ と $\cdot$, 0 と 1 を交換してできる式を元の式の「双対」(dual)と呼ぶ。ブール代数において、ある定理が成り立つならば、その定理の双対もまた成り立つ。このことを「双対性の原理」と呼ぶ。

■

ド・モルガンの法則の左右両辺を見比べてみれば分かるように、同一の内容を表すブール式には複数の形式が存在する。したがって、多数の式を比較するときに一瞥しただけで同一の内容の式なのかどうかを判断することは難しい。そこで、どんなブール式でもこの形に一意に変形できるという形式が考えられている。

定義 17 (加法標準形 (積和標準形))

x, y のような単一の変数と $\bar{x}, \bar{y}$ のような単一の変数の補元をまとめて「リテラル」(literal)と呼ぶことにする。また、 $x \cdot y, y \cdot \bar{z}$ のように同じ変数が重複しては出てこないように作られたリテラルの積を「基本積 (最小項)」と呼ぶ。 $x \cdot y + x \cdot \bar{z}$ のような 2 つ以上の基本積の和を「加法標準形 (積和標準形)」と定義する。各基本積が考えているすべての変数を含んでいるときを特に、「完全加法標準形」と呼ぶ。

■

すべてのブール式は完全加法標準形に変形できることが証明されている。具体的な変形の仕方は以下の通りである。

[完全加法標準形への変形]

1. ド・モルガンの法則を用いて、補演算が個々の変数にのみかかるようにする。
2. 演算規則にしたがって括弧をはずしてゆく。
3. 各基本積において、考えている変数の中で現れていない変数について $x + \bar{x}$ の形の積を基本積に付け加える。
4. 再度括弧をはずす。

例を示せば，以下の通りである．

例 21 完全加法標準形の導出

$$\begin{aligned}
 E(a, b, c) &= \overline{(\overline{a \cdot b}) \cdot c} \cdot \overline{(\overline{a + c}) \cdot (\overline{b + c})} \\
 &= (a \cdot b + \overline{c}) \cdot (\overline{a + c} + \overline{b + c}) \\
 &= (a \cdot b + \overline{c}) \cdot (a \cdot \overline{c} + b \cdot \overline{c}) \\
 &= a \cdot b \cdot a \cdot \overline{c} + a \cdot b \cdot b \cdot \overline{c} + \overline{c} \cdot a \cdot \overline{c} + \overline{c} \cdot b \cdot \overline{c} \\
 &= a \cdot b \cdot \overline{c} + a \cdot b \cdot \overline{c} + a \cdot \overline{c} \\
 &= a \cdot b \cdot \overline{c} + a \cdot b \cdot \overline{c} + a \cdot (b + \overline{b}) \cdot \overline{c} \\
 &= a \cdot b \cdot \overline{c} + a \cdot b \cdot \overline{c} + a \cdot \overline{b} \cdot \overline{c}
 \end{aligned} \tag{6.1}$$

双対性の原理を考えれば分かるように，加法標準形が存在するならばその双対として「乗法標準形（和積標準形）」も存在する．

定義 18 (乗法標準形 (和積標準形))

$x + y$, $y + \overline{z}$ のように同じ変数が重複しては出てこないように作られたリテラルの和を「基本和（最大項）」と呼ぶ． $(x + y) \cdot (x + \overline{z})$ のような 2 つ以上の基本和の積を「乗法標準形（和積標準形）」と定義する．各基本和が考えているすべての変数を含んでいるときを特に「完全乗法標準形」と呼ぶ． ■

すべてのブール式は完全乗法標準形に変形できる．与えられた式を完全乗法標準形に変形するためには，元の式の補演算をとった式の完全加法標準形を求め，もう一度補演算を行えばよい．例を示せば，以下の通りである．

例 22 (完全乗法標準形の導出)

$$\begin{aligned}
 E(a, b, c) &= (a + \overline{b}) \cdot (a \cdot c + b \cdot \overline{c}) \\
 \overline{E(a, b, c)} &= \overline{(a + \overline{b}) \cdot (a \cdot c + b \cdot \overline{c})} \\
 &= \overline{a + \overline{b} + a \cdot c + b \cdot \overline{c}} \\
 &= \overline{a} \cdot b + \overline{a \cdot c} \cdot \overline{b \cdot \overline{c}} \\
 &= \overline{a} \cdot b + (\overline{a} + \overline{c}) \cdot (\overline{b} + c) \\
 &= \overline{a} \cdot b + \overline{a} \cdot c + \overline{a} \cdot \overline{b} + \overline{b} \cdot c \\
 &= \overline{a} \cdot b \cdot (c + \overline{c}) + \overline{a} \cdot (b + \overline{b}) \cdot c + \overline{a} \cdot \overline{b} \cdot (c + \overline{c}) + (a + \overline{a}) \cdot \overline{b} \cdot \overline{c} \\
 &= \overline{a} \cdot b \cdot c + a \cdot \overline{b} \cdot \overline{c} + \overline{a} \cdot b \cdot \overline{c} + \overline{a} \cdot \overline{b} \cdot \overline{c} \\
 E(a, b, c) &= \overline{\overline{E(a, b, c)}} \\
 &= (a + \overline{b} + \overline{c}) \cdot (a + \overline{b} + c) \cdot (\overline{a} + b + c) \cdot (a + b + c)
 \end{aligned} \tag{6.2}$$

6.2 MIL 記号

デジタル計算機においては，ブール代数の各基本演算は電子回路を用いて実現される．このとき，各基本演算を適当な図形で表現しておき，ブール式の計算を図形の間の結線で表現してや

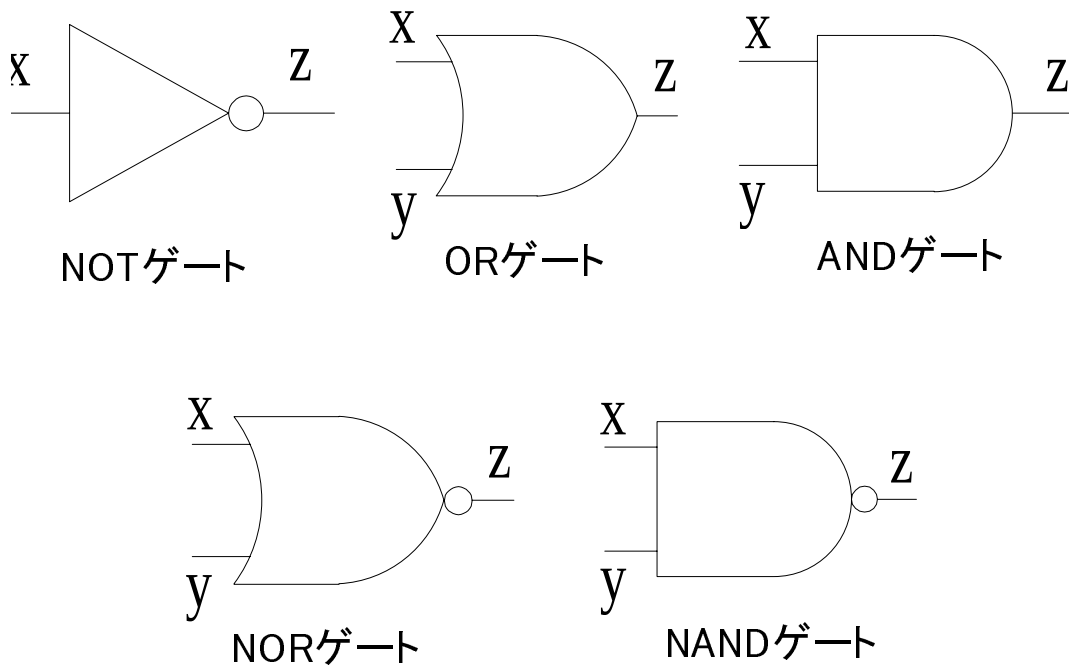


図 6.1: MIL 記号

ると，論理演算と実際のハードウェアの間に簡単な対応が付き便利である．このような目的のために考案された記号が「MIL (Military Standard Specification)」と呼ばれるものである．名前から想像できるように，MIL 記号は元々アメリカ軍の部品調達のための規格として作られたものであるが，その後広く一般に用いられ，現在では国際的な標準として認められている．

MIL 記号で定義されている主な記号は図 6.1 のようなものである．ブール代数の補演算，和，積に対応して NOT ゲート，OR ゲート，AND ゲートがあり，OR ゲート，AND ゲートと NOT ゲートを組み合わせた回路をまとめて表すために NOR ゲートと NAND ゲートが定義されている．また，補演算のように 0 と 1 が反転するような操作を表すために白丸が用いられている．

それぞれのゲート回路の入出力関係は以下の通りであり，ブール代数の演算をそのまま表すようになっている．

NOT		OR			AND			NOR			NAND		
x	y	x	y	z	x	y	z	x	y	z	x	y	z
0	1	0	0	0	0	0	0	0	0	1	0	0	1
0	1	0	1	1	0	1	0	0	1	0	0	1	1
1	0	1	0	1	1	0	0	1	0	0	1	0	1
		1	1	1	1	1	1	1	1	0	1	1	0

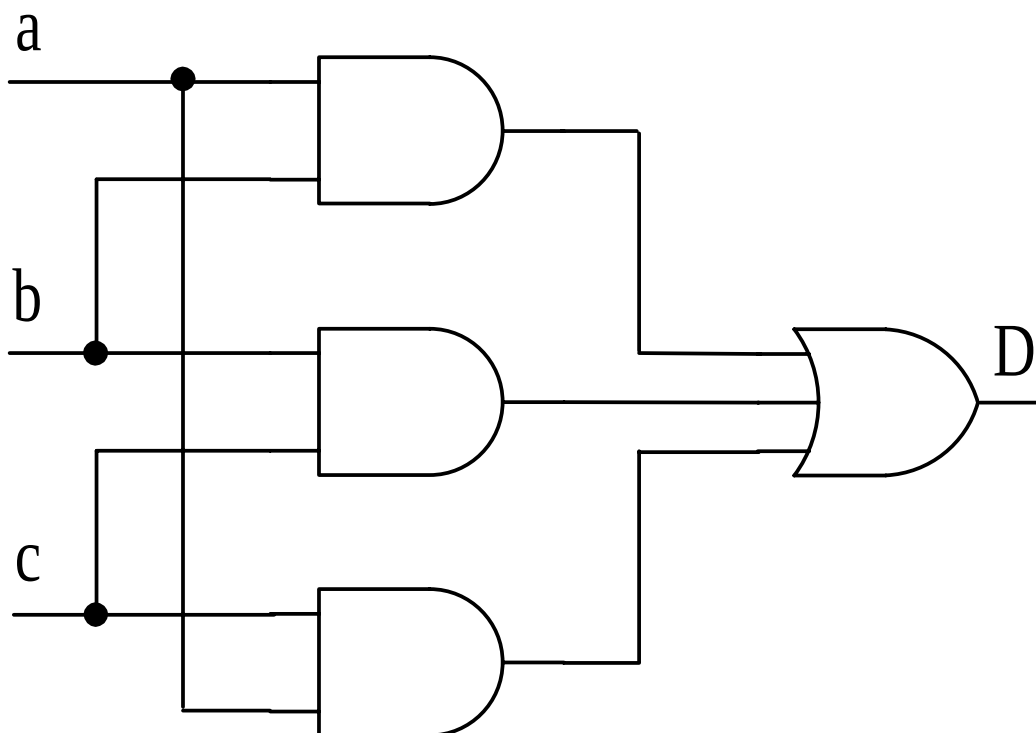


図 6.2: 多数決回路

例 23 (多数決回路) a, b, c という 3 つの変数の多数決を取る回路を考えてみよう．それぞれの変数が 1 (賛成) と 0 (反対) という 2 値を取り，2 つ以上の変数が 1 になると出力が 1 になるようにすればよいので，ブール式は次のようになる．

$$D(a, b, c) = a \cdot b + b \cdot c + c \cdot a \quad (6.3)$$

このブール式を MIL 記号で表すと，図 6.2 のようになる．

演習問題

問 12 ブール式 $E(x, y, z) = (x + y) \cdot \overline{(y + z)} \cdot (z + x)$ について以下の問に答えよ．

1. 式を完全加法標準形に変形せよ．
2. 真理値表を書け．表は辞書式順序で書くこと．
3. 式を MIL 記号で表現せよ．

第7章 状態機械

前章で述べたブール代数に基づく論理回路は、以前どのような入力がありどのような出力を行ったかには全く関係なく、現在の入力にのみ依存して出力を決定する（１歩歩くと昔のことはすべて忘れる鳥頭の回路というわけである）。このような回路を「組み合わせ回路」などと呼ぶ。デジタル計算機を実現するためには、組み合わせ回路だけではなく、以前の入出力にも依存して出力が決まるような、いわば「記憶のある」回路が必要である。このような回路を数学的に表現したものが「状態機械」と呼ばれるものである。本章では、状態機械の基礎について述べる。

7.1 有限状態機械

状態機械は過去の記憶を持っていないから、内部に記憶を保持するための「状態」を持っている。今、状態が有限個であると仮定し、それぞれの状態に $s_1, s_2, \dots, s_Q$ と名前を付けることにしよう。また、状態の作る集合を $S = \{s_1, s_2, \dots, s_Q\}$ とする。

次に、状態が変化するためには外部から何らかの刺激（入力）がなければならないので、 $i_1, i_2, \dots, i_P$ という P 種類の入力を考え、その集合を I としよう。

また、機械の変化を外部から観察するためには、機械から何らかの出力が出ていなければならない。これが $o_1, o_2, \dots, o_R$ という R 種類あるものとし、その集合を O としよう。

最後に、状態 s の機械に対して入力 i が与えられたとき、出力と次の状態を決定する仕組み（関数）が必要である。現在の状態と現在の入力から現在の出力を決定する関数を g 、現在の状態と現在の入力から次の状態を決定する関数を f と定義しよう。

以上定義した S, I, O, f, g の組み合わせによって「有限状態機械」（「順序機械」）を定義する。記号的に書く場合には、 $M < S, I, O, f, g >$ などと書く。

例 24（「仏の顔も三度まで」機械） 次のような有限状態機械を考えてみよう。

- 状態の集合 $S = \{s_0, s_1, s_2, s_3\}$
- 入力の集合 $I = \{i_1(= \text{まっとうに生きている}), i_2(= \text{悪事を働く})\}$
- 出力の集合 $O = \{o_1(= \text{にこやかに微笑んでいる}), o_2(= \text{仏罰を当てる})\}$
- 状態遷移を決定する関数

$$\begin{aligned} f(s_0, i_1) &= s_0, & f(s_0, i_2) &= s_1 \\ f(s_1, i_0) &= s_1, & f(s_1, i_2) &= s_2 \\ f(s_2, i_0) &= s_2, & f(s_2, i_2) &= s_3 \\ f(s_3, i_0) &= s_3, & f(s_3, i_2) &= s_0 \end{aligned}$$

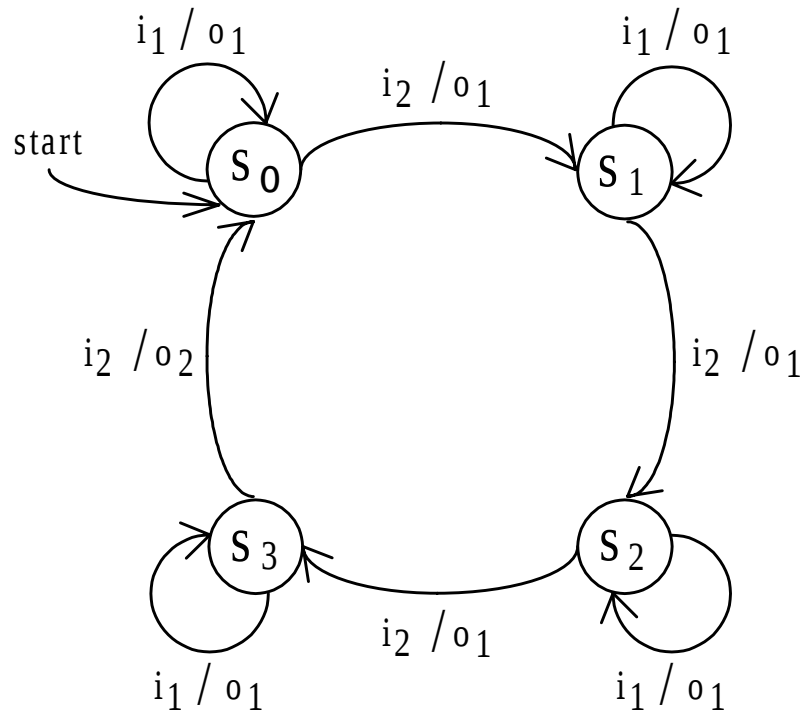


図 7.1: 「仏の顔も三度まで」機械

- 出力を決定する関数

$$\begin{aligned}
 g(s_0, i_1) &= o_1, & g(s_0, i_2) &= o_1 \\
 g(s_1, i_1) &= o_1, & g(s_1, i_2) &= o_1 \\
 g(s_2, i_1) &= o_1, & g(s_2, i_2) &= o_1 \\
 g(s_3, i_1) &= o_1, & g(s_3, i_2) &= o_2
 \end{aligned}$$

この機械においては、入力はある仏教徒の生活態度であり、出力はそれに対する仏の反応である。仏教徒が1回、2回と悪事(i_2)を働いても仏は変わらずにこやかに微笑んでいる(o_1)が、その内部状態は $s_0 \rightarrow s_1 \rightarrow s_2$ と遷移しており、仏教徒が4度目の悪事を働くと突如として仏罰(o_2)を当ててくる。仏罰を当てると、仏はまた慈悲の心を取り戻して状態 s_0 に戻る¹。

上の例題では状態の遷移を表すために関数 f と g を列挙しているが、少々煩雑である。そこで、

	i_1	i_2
s_0	s_0, o_1	s_1, o_1
s_1	s_1, o_1	s_2, o_1
s_2	s_2, o_1	s_3, o_1
s_3	s_3, o_1	s_0, o_2

¹言うまでもないが、この例題は世間一般の俗な仏教理解を茶化して描いてみたものであり、宗教としての仏教そのものには何の関係も無い。

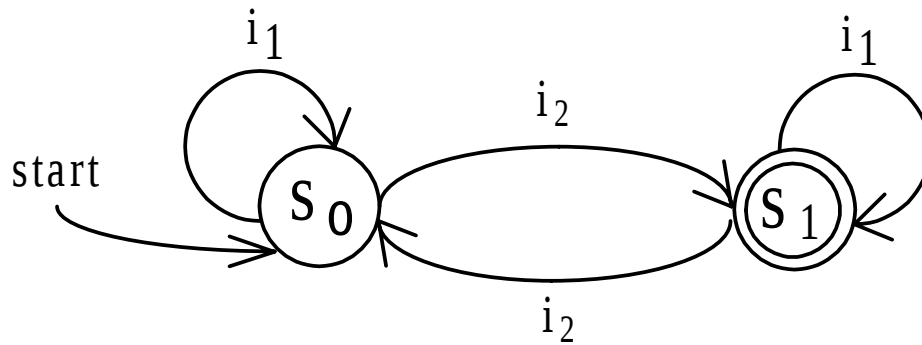


図 7.2: 有限オートマトンの状態遷移図

のように関数の入出力だけを書き並べた表を用いることがある。これを「状態遷移表」などと呼ぶ。表の最左列は現在の状態，表の最上行が入力を表し，残りの部分が対応する出力と新状態を表している。

状態遷移表は関数を書き並べるよりは効率が良いが，状態遷移を直観的に把握するには不十分である。そこで，図 7.1 のような「状態遷移図」を用いる。状態遷移図において， $\circ$ で表されているのが個々の「状態」であり，状態から出ている矢印付きの線が「状態遷移」を表している。矢印の向かう先が遷移後の新しい「状態」である。また，「状態遷移」の線の上に描かれている「 i_1/o_1 」等は，その状態遷移を引き起こした「入力」とその結果の「出力」を表している。図 7.1 は上述の「仏の顔も三度まで」機械を状態遷移図で表したものであるが，仏教徒が悪事を働く（ i_2 ）と仏の状態が遷移し，まっとうに暮らしている（ i_1 ）と仏の状態が変わらないことが，図から直観的に良く理解できるはずである。

7.2 有限オートマトン

有限状態機械においては機械の出力を観測していたが，出力を考えず状態が望ましい状態（「受理状態」）にあるか望ましくない状態（「拒否状態」）にあるかだけを考える場合を「有限オートマトン」と呼ぶ。したがって，有限オートマトンは状態の集合 S ，入力の集合 I ，受理状態の集合 A ($A \in S$)，状態遷移関数 $f(s, i)$ によって構成され， $M = \langle S, I, A, f \rangle$ などと書くことができる。

有限オートマトンの状態遷移図を書く場合には，図 7.2 のように受理状態を $\circ$ で表し，拒否状態と区別する。

有限オートマトンは，日本語のように人間が通常用いている言語（自然言語）や C 言語のように計算機が用いる言語（人工言語）の解析に用いられるが，詳細は省略する。

演習問題

問 13 入力として 0 か 1 を取り，1 が 3 つ連続した場合にのみ 1 を，その他の場合には 0 を出力する有限状態機械の状態遷移表および状態遷移図を描け。

第8章 読書案内

離散数学全般に関しては、

1. 柴田 正憲，浅田 由良：情報科学のための離散数学，コロナ社．
2. 林 晋，八杉 満利子：情報系の数学入門，オーム社．
3. C.L.Liu：コンピュータサイエンスのための離散数学入門，オーム社．
4. 黒澤 和人：情報数学入門，共立出版．
5. S.Lipschutz：離散数学，オーム社．
6. 大矢 雅則・編著：情報数理入門，サイエンス社．

などがある．1. と 2. は比較的コンパクトな教科書，3. は離散数学全般に渡る大部な教科書，4. と 5. は演習に重点を置いた本である．6. は、情報科学の基礎となる数学を細部にこだわらず簡潔にまとめたもので、本講義と同様な狙いで書かれているが、そのカバーする範囲は本講義よりもはるかに広い．この他にも「離散数学」「コンピュータのための数学」などの名前で多くの本が出版されている．

情報理論に関しては、

1. 甘利 俊一：情報理論，ダイヤモンド社．
2. 水野 弘文：情報数理の基礎，培風館．
3. 平澤 茂一：情報理論，培風館．
4. 豊田 正：情報の物理学，講談社サイエンティフィク．
5. 杉本 大一郎：エントロピー，中央公論社．

などを参照されたい．1. は、私自身がかって学生の頃、教科書として読んだ本．1970 年の刊行であり既に新刊では手に入らないかもしれないが、図書館などで探せば読むことはできるはずである．導入部の分かりやすさが印象的だった．2. と 3. は、最近の教科書である．エントロピーはもともと物理学から出て来た概念であって、後に通信理論・情報理論の研究者がこれを借用するようになった．そのような意味で、エントロピーの物理的な基礎を知っておくことも有用であろう．3. と 4. はそのための参考書である．

符号理論に関しては、

1. 今井 秀樹：符号理論，コロナ社．

が専門家の手になる符号理論の標準的な教科書である．

集合論，論理学：

1. 本橋 信義：新しい論理序説，朝倉書店．
2. 竹内 外史：新装版 集合とは何か，講談社（ブルーバックス）．

1. は，最近出た論理学の教科書．2. は集合論の専門家による定評のある啓蒙書であり，最近，新装版が出版されたので簡単に入手できるはずである．

ブール代数と基本論理回路：

1. 岡野 大介，松本 欣也：デジタル基本回路入門，森北出版．
2. 藤井 信生：なっとくするデジタル電子回路，講談社．

などを参照．

情報科学の歴史について：

1. 高橋 秀俊：岩波講座情報科学 1 情報科学の歩み，岩波書店．
2. 星野 力：誰がどうやってコンピュータを創ったのか？，共立出版．
3. 山崎 俊雄，木本 忠昭：新版 電気の歴史，オーム社．

情報科学の歴史に関しては，日本語で書かれた独立の著作は多くないが，「情報科学講座」のようなシリーズ本の一部として刊行されているものは比較的多い．1. は，そのような本の一冊である．2. は，今日のいわゆる「ノイマン型コンピュータ」の発明者は本当にフォン・ノイマンなのか，という問題に的を絞って情報科学の歴史を考察した本であり，読み物としても面白い．数学→自動計算機械→コンピュータという流れで考えた場合，コンピュータは必ずしも電子回路で実現される必要はないのであるが，歴史的には，電磁気学→電気回路→電子回路→コンピュータという流れがあったのも事実である．したがって，電気に関する歴史的知識を持っていればコンピュータの歴史について考える場合にも参考になることが多い．3. は，電気・電子に関する学問についての歴史を描いた一冊である．

数学について：

1. 瀬戸山 士郎：はじめての現代数学，講談社（現代新書）．
2. 数学セミナー編集部：数学完全ガイドンス，日本評論社．

本学科は数学者を養成する学科ではないので，数学の細部について知る必要は必ずしもないが，数学的な考え方を理解しておくことは将来役に立つであろう．上記は，そのための参考書である．

理工系の学問においては，事実と意見を峻別し，論理的に構成され，読みやすく誤解されにくい文章を書く必要がある．これは，いわゆる「作文」「感想文」「小論文」などといった文章とは大きく異なるものであり，意図的に努力しなければ書けるようにはならない．理工系の文書の書き方に関しては，下記の本が定評がある．

- 木下 是雄：レポートの組み立て方，筑摩書房（筑摩学芸文庫）．
- 木下 是雄：理科系の作文技術，中央公論社（中公新書）．

現在の大学のカリキュラムにおいては，「工学や技術とは何か」ということに関してまとまった講義は残念ながら存在しないのが普通なので，自ら参考書を探して自習する必要がある．とりあえず，下記を参照．

1. 大輪 武司：技術とは何か，オーム社．
2. 齊藤 了文：＜ものづくり＞と複雑系，講談社．
3. 飯田 賢一：一語の辞典 技術，三省堂．
4. 小宮山 宏：地球持続の技術，岩波書店 (岩波新書)．
5. 吉田 文和：ハイテク汚染，岩波書店 (岩波新書)．
6. 吉田 文和：IT 汚染，岩波書店 (岩波新書)．

第9章 練習問題略解

問1

1.

$$\begin{aligned} p(A_1, B_1) &= \frac{N_{11}}{N} = 0.3 & p(A_1, B_2) &= \frac{N_{12}}{N} = 0.1 \\ p(A_2, B_1) &= \frac{N_{21}}{N} = 0.2 & p(A_2, B_2) &= \frac{N_{22}}{N} = 0.15 \\ p(A_3, B_1) &= \frac{N_{31}}{N} = 0.1 & p(A_3, B_2) &= \frac{N_{32}}{N} = 0.15 \end{aligned}$$

2.

$$\begin{aligned} p(A_1) &= p(A_1, B_1) + p(A_1, B_2) = 0.4 \\ p(A_2) &= p(A_2, B_1) + p(A_2, B_2) = 0.35 \\ p(A_3) &= p(A_3, B_1) + p(A_3, B_2) = 0.25 \\ p(B_1) &= p(A_1, B_1) + p(A_2, B_1) + p(A_3, B_1) = 0.6 \\ p(B_2) &= p(A_1, B_2) + p(A_2, B_2) + p(A_3, B_2) = 0.4 \end{aligned}$$

3.

$$\begin{aligned} H(A, B) &= - \sum_{i=1}^3 \sum_{j=1}^2 p(A_i, B_j) \log_2(p(A_i, B_j)) \\ &= -0.3 \log_2(0.3) - 0.1 \log_2(0.1) - 0.2 \log_2(0.2) \\ &\quad - 0.15 \log_2(0.15) - 0.1 \log_2(0.1) - 0.15 \log_2(0.15) \\ &\doteq 2.47 \end{aligned}$$

4.

$$\begin{aligned} H(A) &= - \sum_{i=1}^3 p(A_i) \log_2(p(A_i)) \\ &= -0.4 \log_2(0.4) - 0.35 \log_2(0.35) - 0.25 \log_2(0.25) \\ &\doteq 1.56 \end{aligned}$$

5.

$$\begin{aligned} H(B) &= - \sum_{j=1}^2 p(B_j) \log_2(p(B_j)) \\ &= -0.6 \log_2(0.6) - 0.4 \log_2(0.4) \\ &\doteq 0.971 \end{aligned}$$

(注) 上記の結果は十分な桁数で途中計算を行い最後に四捨五入した結果で，先に四捨五入してから計算すると 0.966 になる．

6.

$$\begin{aligned} p_{A_1}(B_1) &= \frac{p(A_1, B_1)}{p(A_1)} = \frac{3}{4} & p_{A_1}(B_2) &= \frac{p(A_1, B_2)}{p(A_1)} = \frac{1}{4} \\ p_{A_2}(B_1) &= \frac{p(A_2, B_1)}{p(A_2)} = \frac{4}{7} & p_{A_2}(B_2) &= \frac{p(A_2, B_2)}{p(A_2)} = \frac{3}{7} \\ p_{A_3}(B_1) &= \frac{p(A_3, B_1)}{p(A_3)} = \frac{2}{5} & p_{A_3}(B_2) &= \frac{p(A_3, B_2)}{p(A_3)} = \frac{3}{5} \end{aligned}$$

7.

$$\begin{aligned} H_A(B) &= - \sum_{i=1}^3 \sum_{j=1}^2 p(A_i) p_{A_i}(B_j) \log_2(p_{A_i}(B_j)) \\ &= -0.4 \cdot \frac{3}{4} \log_2\left(\frac{3}{4}\right) - 0.4 \cdot \frac{1}{4} \log_2\left(\frac{1}{4}\right) \\ &\quad - 0.35 \cdot \frac{4}{7} \log_2\left(\frac{4}{7}\right) - 0.35 \cdot \frac{3}{7} \log_2\left(\frac{3}{7}\right) \\ &\quad - 0.25 \cdot \frac{2}{5} \log_2\left(\frac{2}{5}\right) - 0.25 \cdot \frac{3}{5} \log_2\left(\frac{3}{5}\right) \\ &\doteq 0.912 \end{aligned}$$

8.

$$I(A, B) = H(B) - H_A(B) = 0.971 - 0.912 = 0.059 \text{ bits}$$

問2

1. $1998_{(10)} = 11111001110_{(2)}$
2. $27.725_{(10)} = 11011.111100_{(2)}$
3. $-666_{(10)} = -1010011010_{(2)} = 11010011010_{(2)}$ (負号を 1 で表すと仮定した場合)

問3

1. $100010011_{(2)} = 275_{(10)}$
2. $10110.101_{(2)} = 22.625_{(10)}$
3. $-11011_{(2)} = -27_{(10)}$

問4

1. $1001011_{(2)} + 1100100_{(2)} = 10101111_{(2)}$
2. $1011000_{(2)} - 1000011_{(2)} = 10101_{(2)}$

3. $101111_{(2)} \times 1011_{(2)} = 1000000101_{(2)}$
4. $10011010_{(2)} + 110_{(2)} = 11001_{(2)}$ 余り $100_{(2)}$

問5

1. $10000_{(10)} - 1998_{(10)} = 8002_{(10)}$
2. $9999_{(10)} - 1998_{(10)} = 8001_{(10)}$
3. $1000000_{(2)} - 1010111_{(2)} = 101001_{(2)}$
4. $111111_{(2)} - 1010111_{(2)} = 101000_{(2)}$

問6

$$-4.75_{(10)} = -100.11_{(2)} = -0.101_{(2)} \times 10_{(2)}^{011} \rightarrow 11010011$$

問7

1.

$$\mathbf{G} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 \end{bmatrix}$$

2.

$$\mathbf{x} = \mathbf{iG} = \begin{bmatrix} 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 \end{bmatrix}$$

3.

$$\mathbf{H} = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \end{bmatrix}$$

4.

$$\mathbf{s} = \mathbf{yH}^t = 1$$

5. シンドローム $\mathbf{s}$ がゼロではないので、正しく伝送されなかったことが分かる。

問8

1. 冗長度 $= \frac{N-K}{N} = \frac{3}{7}$

2.

$$G = \begin{bmatrix} 1 & 0 & 0 & 0 & 1 & 1 & 1 \\ 0 & 1 & 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 & 1 & 1 \end{bmatrix}$$

3.

$$x = iG = \begin{bmatrix} 0 & 1 & 1 & 0 & 0 & 1 & 1 \end{bmatrix}$$

4.

$$H = \begin{bmatrix} 1 & 1 & 1 & 0 & 1 & 0 & 0 \\ 1 & 1 & 0 & 1 & 0 & 1 & 0 \\ 1 & 0 & 1 & 1 & 0 & 0 & 1 \end{bmatrix}$$

5.

$$s = yH^t = \begin{bmatrix} 0 & 1 & 1 \end{bmatrix}$$

6. シンドローム s がゼロベクトルではないので、正しく伝送されなかったことが分かる。

問9

1.

+	0	1	2
0	0	1	2
1	1	2	0
2	2	0	1

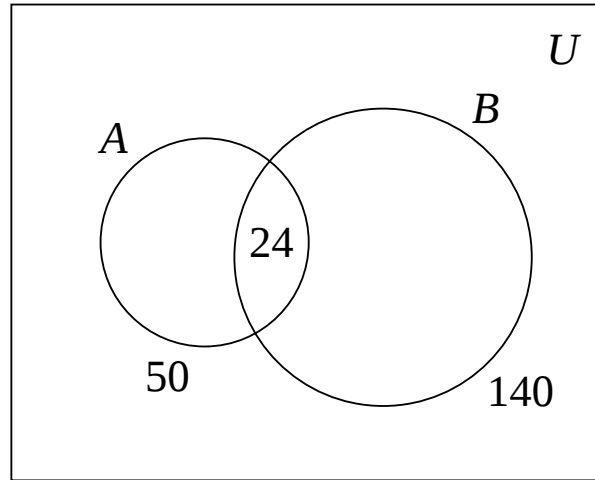
2.

×	0	1	2
0	0	0	0
1	0	1	2
2	0	2	1

- 3.
- 加算の表から加算の単位元を 0 、 $0, 1, 2$ の逆元をそれぞれ $0, 2, 1$ と選ぶことができるので、この集合は加算に関して群を成す。
 - $0, 1, 2$ から加算の単位元を除いた $1, 2$ に関して、乗算の単位元を 1 、 $1, 2$ それぞれの逆元を $1, 2$ と選ぶことができるので、 $1, 2$ の集合は乗算に関して群を成す。
 - よって、この集合は 3 を法とする演算に関して体を成す。

問10

1.



ベン図

$$2. \quad \overline{A \cup B} = U - A \cup B$$

$$3. \quad 200 - (50 - 24) - (140 - 24) - 24 = 34 \text{ 人}$$

問11

以下の式を使えばよい.

$$2^{k+1} \times 2^{k+1} - 1 = 4 \times 2^k \times 2^k - 1 = 3 \times 2^k \times 2^k + 2^k \times 2^k - 1$$

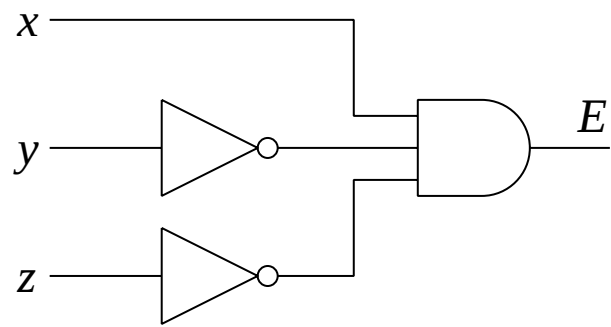
問12

$$1. \quad E(x, y, z) = x \cdot \bar{y} \cdot \bar{z}$$

2.

x	y	z	E
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	0
1	0	0	1
1	0	1	0
1	1	0	0
1	1	1	0

3.

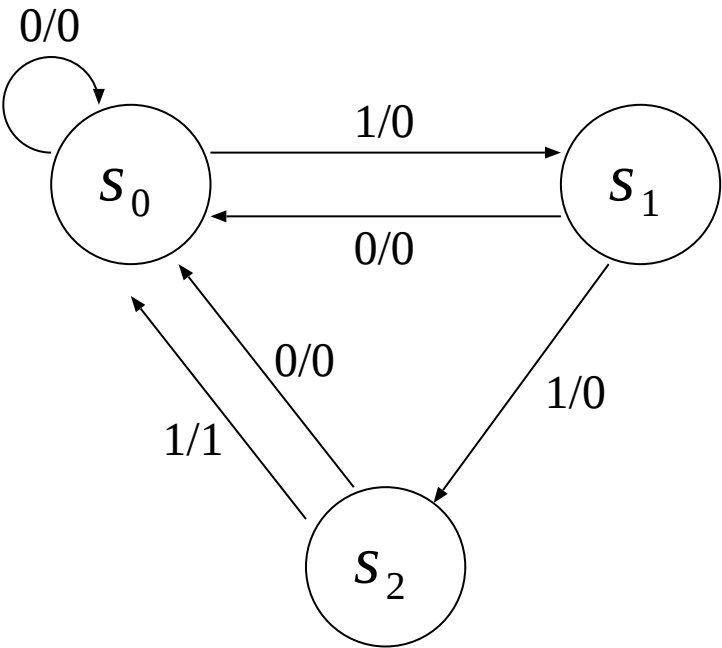


MIL 記号図

問 13

状態遷移表

		入力	
		0	1
状態	s_0	$s_0, 0$	$s_1, 0$
	s_1	$s_0, 0$	$s_2, 0$
	s_2	$s_0, 0$	$s_0, 1$



状態遷移図