8 $\boldsymbol{NP}$ 完全問題

これまで探索，整列などの問題を効率よく解くアルゴリズムについて考えてきた．しかし，これらの問題は「効率のよい」アルゴリズムが知られている比較的幸運な例である．問題のなかには(それらが非常に有用であるにもかかわらず)効率よく解くアルゴリズムが知られていないものが多数ある．このような問題を**手に負えない問題**(intractable problem)というが，ここではそのような手に負えない問題について考える．

§8.1 手に負えない問題

手に負えない問題というのはどういうものかというと，可能な解の候補をすべて**しらみつぶし**(exhaustive)で検討するしか有効な方法のないような問題である．

例(しらみつぶしによる整列) n 個の整数を整列するしらみつぶしのアルゴリズムは，次のようなものになる．

（i）n 個の整数のすべての順列を生成する

（ii）生成した順列を一つずつチェックして整列されたものを探す

n 個の整数の順列は $n!$ 通りあるから，このアルゴリズムの計算量は少なくとも $O(n!)$ である．n を少し大きくしただけで計算量が爆発することがわかる．

すでに紹介したように，整列に関しては効率のよいアルゴリズムがあるので，整列は手に負えない問題ではない．しかし，実際に効率のよいアルゴリズムが必要とされている問題であるにもかかわらず，手に負えない問題が多数知られている．

ナップザック問題 n 個の品物があって，それぞれに値段と点数がついている．n 個からいくつか選んで，値段の総計がある制限値を越えず，しかも点数の総和が最大になるようにせよ．[*1]

巡回セールスマン問題 あるセールスマンが，お得意様を電車を使って回るのに，最寄りの駅からスタートして，お得意様の家のある駅ですべて降りてもとの駅に帰ってきたい．ただし，効率よく回りたいので，同じ駅を二度通らずに，しかもできるだけ電車に乗っている時間を短くしたい．どのようなルートで回ればよいか？[*2]

3 彩色問題 無向グラフが与えられたとき，すべての頂点を，辺で結ばれたどの 2 つの頂点も同じ色にならないように 3 通りの色で塗り分けよ．

論理式の充足問題 論理和 $\vee$(and)，論理積 $\wedge$(or)，論理否定 $\neg$(not)を含む論理式を満たすような真偽値の割当てを探せ．

(例) 論理式 $(x \vee y) \wedge \neg y$ を考える．
この論理式を満たす真偽値の割当ては $x = \mathit{true}$, $y = \mathit{false}$

§8.2 クラス $\boldsymbol{P}$ とクラス $\boldsymbol{NP}$

効率のよい問題の集合と手に負えない問題はどのようにして区別できるだろう．また，手に負えない問題はどのような性質をもつ集合(**クラス**(class)という)だろうか．

*1 例えば，遠足に持っていくおやつを買うとき，使ってよいお金の範囲で，できるだけ自分の好きなお菓子(点数であらわす)を多く買うには何をいくつ買えばよいか，という問題を考えよ．
*2 ただし，電車はすべて各駅停車で，特急や快速などはないとする．

(a)　クラス $\boldsymbol{P}$

手に負えない問題を考える前に，効率のよいアルゴリズムを持つ問題について考えよう．効率のよいアルゴリズムとは，しらみつぶしを使わないアルゴリズムと言えるから，次のように定式化できる．

定義 8.1　ある問題が多項式時間程度の計算量，すなわち計算量 $O(n^k)$（k は適当な自然数）のアルゴリズムを持つとき，その問題は**クラス $\boldsymbol{P}$**（class of **P**olynomial complexity）[*3] に属するという．□

以後，効率のよいアルゴリズムとはクラス P に属するアルゴリズムのことを言う．

(b)　クラス $\boldsymbol{NP}$

われわれが本当に解きたい手に負えない問題とは，§8.1 で示したような問題だが，ここではそれらをもう少し簡単にした問題を考えよう．

ナップザック問題　値段の総計が制限値を越えない範囲で，点数の総和がある与えられた点数 p 以上になるような品物の組合せがあるか？

巡回セールスマン問題　電車に乗っている時間がある与えられた時間 t 以下になるようなルートが存在するか？

3 彩色問題　すべての頂点を 3 通りの色で，辺で結ばれたどの 2 つの頂点も同じ色にならないように塗り分けることができるか？

論理式の充足問題　与えられた論理式を満たすような真偽値の割当てが存在するか？

もとの問題では，問題の条件を満たす組合せ（たとえば，ナップザック問題であればどの品物をいくつ選べばよいか）まで答えなければならなかったが，これらの問題はそのような組合せがあるかどうかだけ答えればよい．したがって，これらの

*3　polynomial = 多項式

問題の求める答はすべて Yes か No である．このような問題を**決定問題**（decision problem）という．以後，手に負えない問題はすべて決定問題の形のもののみ考えることとする．

手に負えない問題を，クラス P に属さない問題として定式化するのはあまり適当でない．なぜなら，上にあげた問題に関しては，たしかに多項式時間の計算量で解くアルゴリズムはまだ知られていないが，多項式時間の計算量で解けないという証明も与えられていないからだ．

ここで，手に負えない問題を解くのを難しくしているのは何かを考えてみよう．困難の原因は，しらみつぶしですべての解の組合せを当たってみなければならないところにある．いま，仮に，しらみつぶしで当たってみなければならない組合せのうち，もっとも有望なものを一瞬にして与えてくれる，魔法のような手続きがあったとしよう．そうすると，上記の問題は効率よく解くことができる．なぜなら，魔法の手続きで得られた解の組合せが問題の条件を満たしているかどうかをチェックして Yes/No を答えるのには多項式時間程度ですむからである．

手に負えない問題を次のように定式化する．

定義 8.2　ある決定問題が，次のようなアルゴリズムを持つとき，その決定問題は**クラス $\boldsymbol{NP}$**（class of **N**on-deterministic **P**olynomial complexity）に属するという．

- アルゴリズムは，解の候補をあてずっぽう（**非決定的**（non-deterministic））につくった後，その候補が条件を満たせば Yes，満たさなければ No を返す．
- 解の候補をつくるのに $O(1)$，条件を満たすかどうかのチェックに $O(n^k)$（k は適当な自然数）の計算量を要する．□

上記 4 つの問題がすべてクラス NP に属することは，容易にわかる．

§8.3　$\boldsymbol{P = NP}$ と $\boldsymbol{NP}$ 完全問題

計算機科学における，未解決問題の一つは，

$$\text{クラス } P = \text{クラス } NP$$

かどうかである．すなわち，§8.2(b) で定義した手に負えない問題のクラスが，本当に手に負えないものなのか，ひょっとしたら多項式時間で解くアルゴリズムが存在するのではないかという疑問である．

(a)　$\boldsymbol{NP}$ 完全問題

$P = NP$ 問題を考えるときに重要なのが，**$\boldsymbol{NP}$ 完全**(NP-complete)の考え方である．NP 完全な問題とは，クラス NP に属する問題のうち，解くのがもっとも難しいものをいう．ただし，問題の難しさを比べるのには次に示す尺度を使う．

定義 8.3（**多項式還元性**(**polynomial reducibility**)）ある決定問題 Π_1 が決定問題 Π_2 に多項式還元可能とは，問題 Π_1 の入力を問題 Π_2 の入力の形に翻訳する，次のようなアルゴリズム T が存在することをいう．

- アルゴリズム T はクラス P に属する．
- 問題 Π_1 を解くアルゴリズムに直接入力して計算した結果と，問題 Π_2 を解くアルゴリズムに T で変換したものを入力した結果が一致する．(図 8.1 参照)

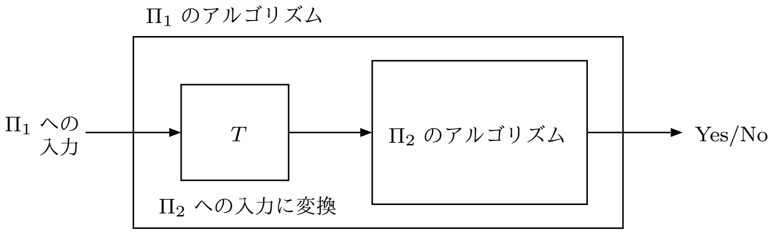

図 8.1　多項式還元

このとき，$\Pi_1 \propto \Pi_2$ とかく．　□

定義 8.4　ある問題 Π が NP 完全問題であるとは，

- 問題 Π がクラス NP に属し，かつ
- すべてのクラス NP に属する問題 Π' について，$\Pi' \propto \Pi$

であることをいう．　□

NP 完全問題が存在することは知られている．

定理 8.5（**Cook**）論理式の充足性の決定問題は NP 完全である．　□

定義 8.4 より，適当な NP 完全問題 Π_1 をクラス NP に属する問題 Π_2 に多項式還元可能(すなわち $\Pi_1 \propto \Pi_2$)ならば，問題 Π_2 もまた NP 完全であることが分かる．上記の定理は，実は次の定理からそのようにして導き出すことができる．

定理 8.6　論理式を**論理積標準形**(conjunctive normal form)に制限した場合もその充足性の決定問題は NP 完全である．

ただし，論理積標準形とは次の形で表されるものをいう．

$$\bigwedge_{i=1}^{n}(\bigvee_{j=1}^{m(i)} L_{i,j})$$

ここで，$L_{i,j}$ は $x, \neg x$ のいずれかの形をしているとする．$L_{i,j}$ のことを**リテラル**(literal)という．　□

論理積標準形の充足性決定問題を，以下 SAT であらわす．

SAT をさらに制限した 3-SAT も NP 完全であることが知られている．

問 8.1　3-論理積標準形の充足性の決定問題(3-SAT)が NP 完全であることを，SAT から 3-SAT への多項式還元によって示せ．ただし，3-論理積標準形とは次の形で表されるものをいう．

$$\bigwedge_{i=1}^{n}(\bigvee_{j=1}^{3} L_{i,j})$$

ここでは例として，3 彩色問題の NP 完全性を多項式還元によって示す．

定理 8.7　3 彩色問題は NP 完全である．

［証明］3-SAT を 3 彩色問題に多項式還元する．

任意の 3-SAT の論理式

$$P = \bigwedge_{i=1}^{n}(\bigvee_{j=1}^{3} L_{i,j})$$

に対して

$$P \text{ が充足可能} \longleftrightarrow G \text{ が 3 彩色可能}$$

となるような無向グラフ G が多項式時間で構成できることを示せばよい．

論理式 P の中に現れる変数の集合を $\{x_1, \cdots, x_m\}$ とせよ．図 8.2 のようにグラフ G を構成すればよい．

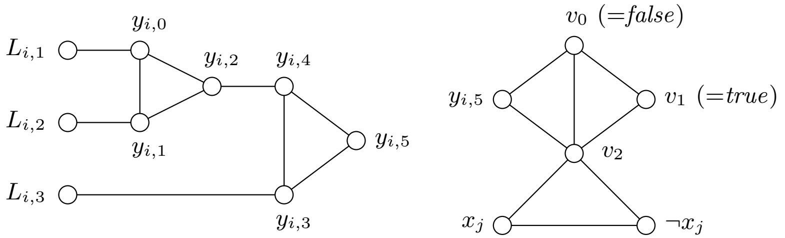

図 8.2　3-SAT に対応する 3 彩色問題のグラフ

ただし，図 8.2 は，すべての $i = 1, 2, \ldots, n$ および $j = 1, 2, \ldots, m$ に関し，図中のすべての頂点と辺を合わせ持つグラフを表す．ただし，同じ名前を持つ頂点は同一視する．たとえば，左図の頂点 $y_{i,5}$ と右図の頂点 $y_{i,5}$ は各 $i\,(i = 1, 2, \ldots, n)$ ごとに同じ頂点と考える (すなわち，左右 2 つのグラフは頂点 $y_{i,5}$ を介してつながっている)．また同様に，左図のリテラル $L_{i,k}$ が x_j の形のときは，左図のリテラルに対応する頂点と右図の頂点 x_j を同一視し，リテラル $L_{i,k}$ が $\neg x_j$ の形のときは，右図の頂点 $\neg x_j$ を同一視する．

この 3-SAT から 3 彩色問題への翻訳の意図するところは，もし論理式中のある変数 x が真 (*true*) ならば，x に対応する頂点の色は頂点 v_1 と同じ色に，偽 (*false*) ならば，x に対応する頂点の色は頂点 v_0 と同じ色になるようにグラフを構成する点にある．

3 彩色問題が NP 完全問題であることは，このグラフが多項式時間で構成できることからわかる．∎

問 8.2　上に定義したグラフ G が 3 彩色可能であることと，3-SAT の論理式 P が充足可能であることが同値であることを示せ．また，グラフ G が多項式時間で構成できることを示せ．

(b)　$\boldsymbol{P = NP}$ ？

NP 完全問題はクラス NP に属する問題のなかでも，もっとも難しい問題のことであるから，$P = NP$ を証明するためには，たくさんある NP 完全問題のうち，どれか一つに対して多項式時間のアルゴリズムを発見すればよい．[*4]

もし，$P = NP$ が証明されたとすると，これはたいへんなことである．なぜなら，いままで現実に必要であるにもかかわらず，効率の悪いしらみつぶしによるアルゴリズムで計算していた問題が，劇的に効率のよいアルゴリズムで計算できることを意味するからだ．しかし，残念ながら，$P = NP$ はまだ証明されていない．逆に，$P \neq NP$ であることも証明されていない．つまり，この問題はいまだ未解決である．

$P = NP$ 問題に関する，大方の研究者の見方は悲観的 (というよりも現実的) で，$P \neq NP$ であろうという予想がされている．つまり，NP 完全問題を，しらみつぶしによる方法より本質的に効率よく解くことのできるアルゴリズムは存在しないと考えている．

*4　現在までに何百個もの NP 完全問題が知られている．