

# アルゴリズムとデータ構造⑩

## ～ 問題の難しさ ～

鹿島久嗣

# 問題の難しさのクラス：

---

- クラスP
- クラスNP
- $P \neq NP$  予想
- NP完全
- NP困難

# 問題の難しさ：

## もっとも計算量の小さいアルゴリズムで測る

- 我々は、アルゴリズムの効率の良し悪しを計算量で測る
- では、問題の難しさはどう測るか？
- 問題 $Q$ に対するアルゴリズムの集合 $A(Q)$ を考える
- 問題 $Q$ の難しさを、最も計算量が小さい  $a \in A(Q)$  の計算量で測ることにする
- 例：ソート問題の計算量の下界は  $O(n \log n)$ 
  - （理論的には）これより高速なアルゴリズムを探す必要はない

クラスP :

多項式時間アルゴリズムが存在する問題のクラス

---

- 決定問題を考える

- 決定問題 : YesかNoで答えられる問題

- 理論的には、  
問題が難しいか？ = 問題が多項式時間で解けるか？  
に興味がある

- クラスP :  
多項式時間アルゴリズムが存在するような問題のクラス

# クラスNP :

## 非決定的な計算機で多項式時間で解けるクラス

### ■ 非決定的な計算機を考える :

#### – 好きなだけ並列化できる理想的な計算機

順列、 $C(n, m)$ など

- プロセスを $k$ 分岐して、並列実行できる命令がある

#### – $O(n)$ 時間で $n$ ビットのベクトルをすべて調べることができる

### ■ NP : 問題の答えが「Yes」のときには、非決定的な計算機を使って、多項式時間でこれを調べることのできる問題

#### – もう少し簡単な定義 : 答えの候補が与えられたときに、答えの正しさを（決定的な計算機で）多項式時間で検証できる問題

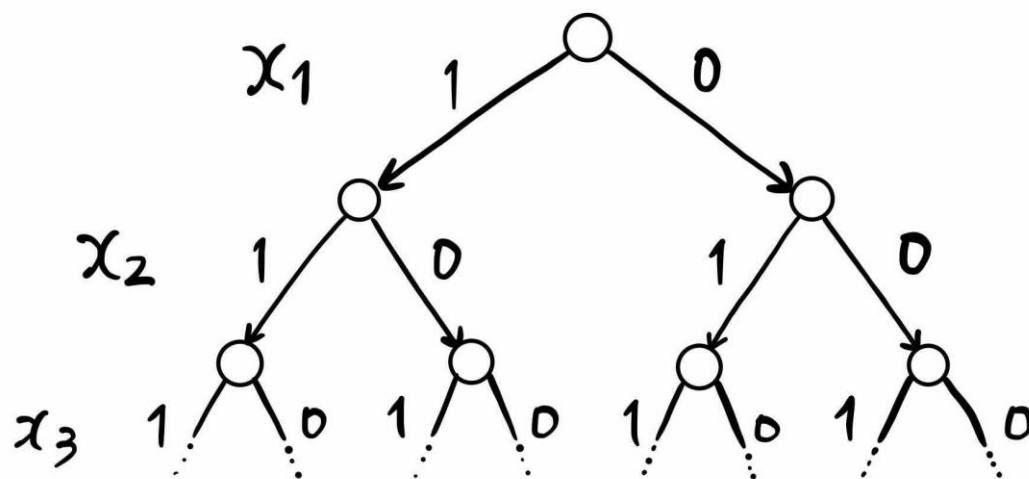
### ■ 実用上重要な問題は、ほぼこのクラスに入る

# 非決定的な計算機で多項式時間で解けるクラス

- 順列、 $C(n, m)$ など

- もう少し  
しさを (シ

- ## ■ 実用上重



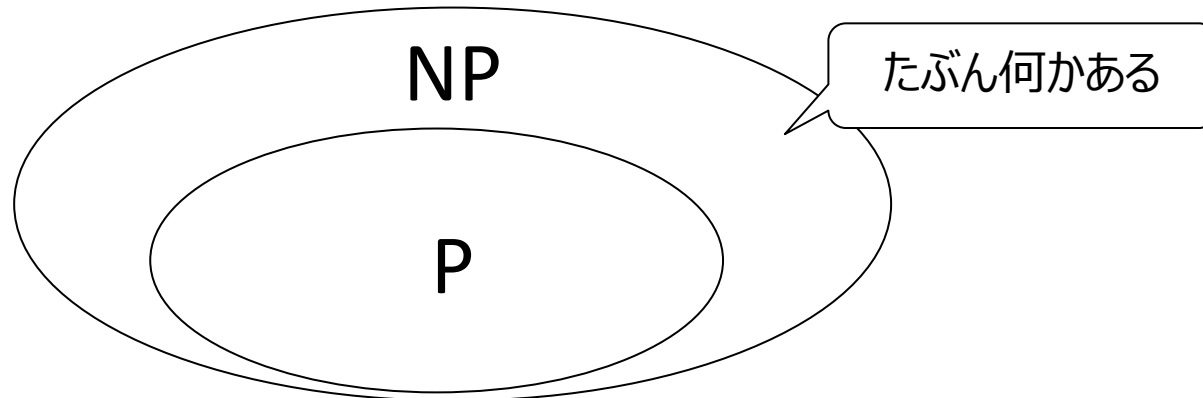
# 定的な計算機 きる問題

次に、答えの正  
誤でできる問題

P≠NP予想：

おそらく、NPはPよりも難しい

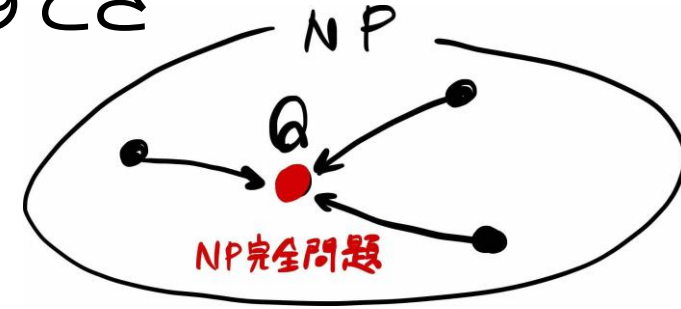
- 「クラスNPに属するが、Pには属さない問題」は、  
これまでに見つかっていない（！）
  - $P \subseteq NP$  であるが、 $P \neq NP$ かどうかは、まだ分かっていない  
(※ 解けたら100万ドルの賞金)
  - おそらく、 $P \neq NP$ であるとみんな思っている



# NP完全：

## NPの中でもっとも難しい問題

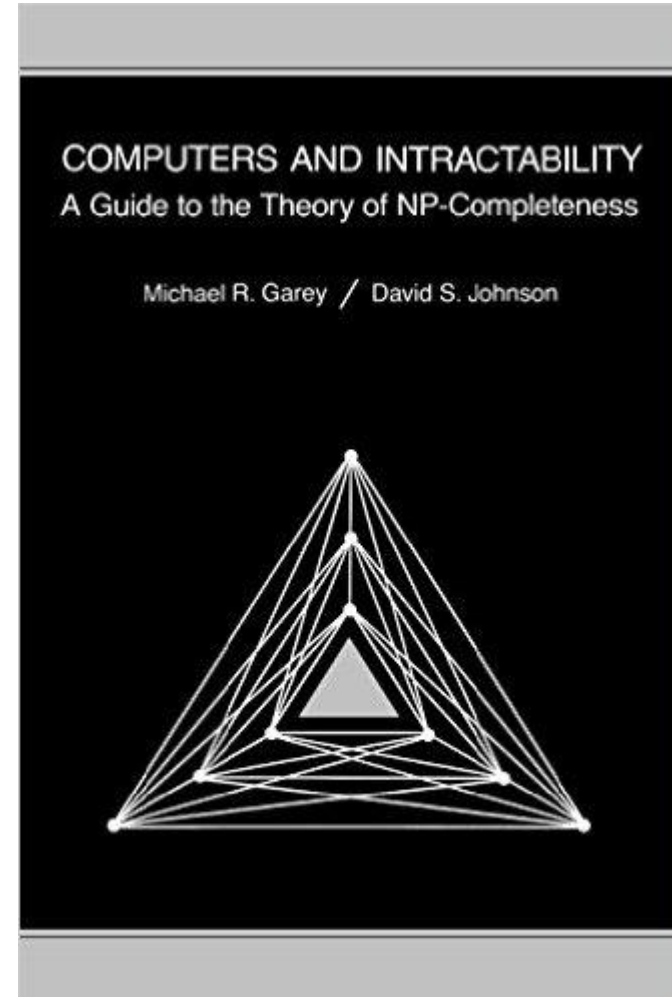
- NP完全： 判定問題Qは以下を満たすときNP完全に属するという
  - QはNPに属する
  - NPの任意の問題が多項式時間でQに変換できる
- もし、NP完全問題QがクラスPに属するならば、NPに属する全ての問題はPに属することになる
  - なぜならば、NP完全問題Qが多項式時間で解ければ、ほかのNP問題も多項式時間で解けることになるから





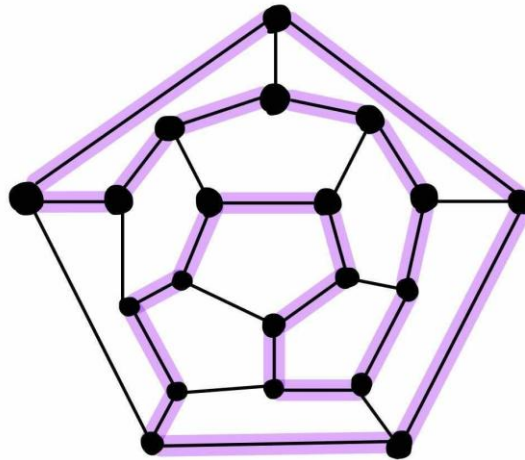
# NP完全問題の例： 多くの重要な問題がNP完全

- ハミルトン閉路問題
- クリーク問題
- 頂点被覆問題
- 集合被覆問題
- 新たな問題に出会ったときには、それがNP完全などでないかをチェックすべし



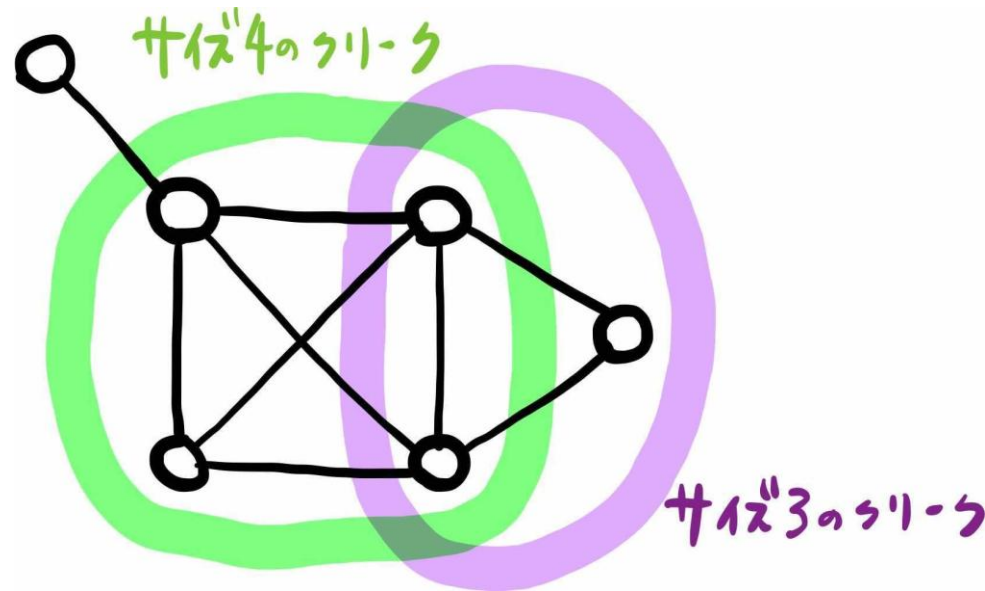
# NP完全問題の例： ハミルトン閉路問題

- ハミルトン閉路：  
グラフ  $G = (V, E)$  において、すべての頂点をちょうど1回ずつ訪れて出発点に戻る道（閉路）
- ハミルトン閉路問題：  
 $G$  がハミルトン閉路をもつかどうかの決定問題



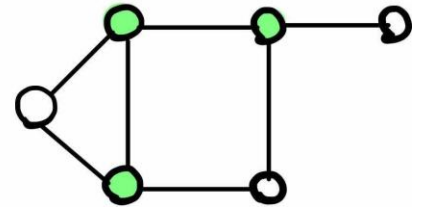
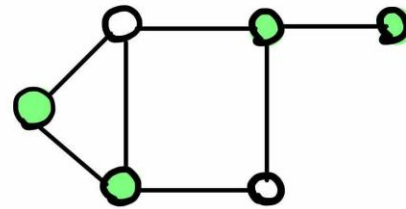
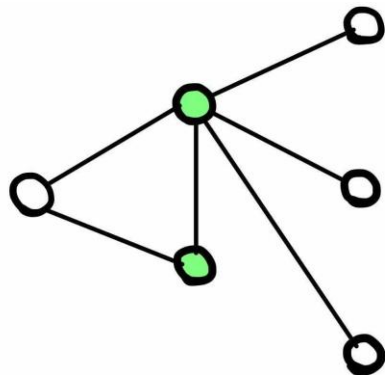
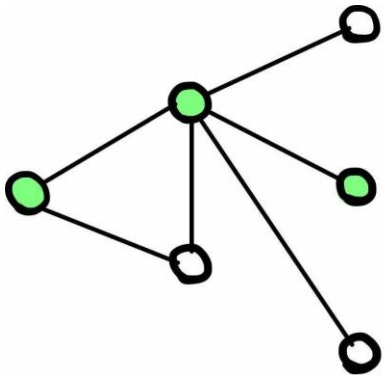
# NP完全問題の例： クリーク問題

- クリーク：  $G$  の部分グラフ  $G'$  であって、 $G'$  の全ての頂点对が互いに辺でつながれているもの
- クリーク問題：  $G$  が頂点数  $k$  のクリークをもつかどうかの決定問題



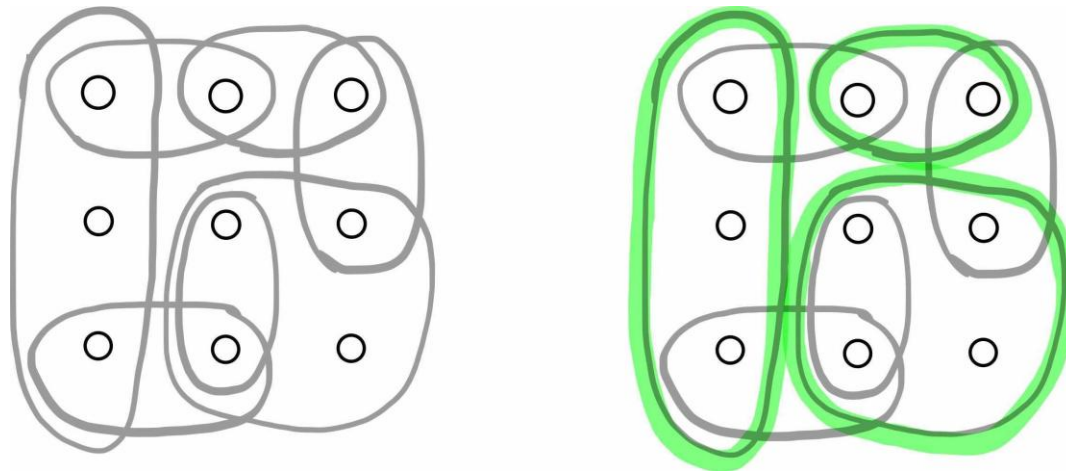
# NP完全問題の例： 頂点被覆問題

- グラフ  $G = (V, E)$  において全ての  $e \in E$  の少なくとも一方が  $V' \subseteq V$  に含まれているとき、 $V'$  を頂点被覆という
- 頂点被覆問題 (vertex cover) : 頂点「で」被覆  
 $G$  が頂点数  $k$  の頂点被覆をもつかどうかの決定問題



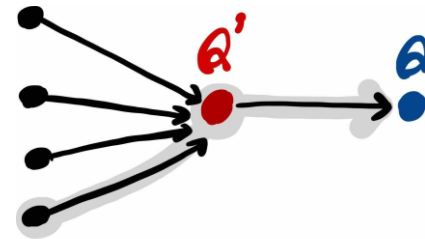
# NP完全問題の例： 集合被覆問題

- $n$ 個の集合  $S_1, S_2, \dots, S_n$  があるとき、そのうち  $k$  個を用いて  $\bigcup_{j=1}^n S_j = \bigcup_{j=1}^k S_j$  とできるとき、大きさ  $k$  の集合被覆という
- 集合被覆問題（set cover）：集合「で」被覆  
 $S_1, \dots, S_n$  が大きさ  $k$  の集合被覆をもつかどうかの決定問題



# NP完全であることの証明法： 多項式時間帰着によって示す

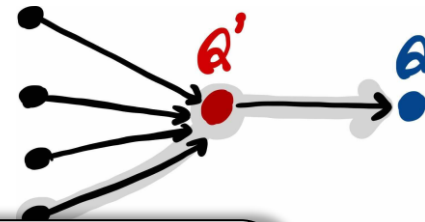
- ある問題QがNP完全であることを示すためには、あるNP完全問題Q'からQに多項式時間で帰着できることを示す
  - Q'の任意の問題例をQの問題例に、多項式時間で対応付けることができることを示す
  - Qの問題例の解がYesなら、Q'の問題例の解もYes、逆に、QがNoならQ'でもNo
- なお、最初のNP完全問題は定義に従い示す必要がある
  - 充足可能性問題（SAT）：与えられた論理式を真にする論理変数の割り当てが存在するかを判定する問題



# NP完全であることの証明法： 多項式時間帰着によって示す

- ある問題QがNP完全であることを示すためには、あるNP完全問題Q'からQに多項式時間で帰着できることを示す

- Q'の任意の問題例をQの問題例に、多項式時間で対応付けることができることを示す



- Qの  
逆

$(x_1 \vee \neg x_2) \wedge (\neg x_1 \vee x_3)$ は  
 $x_1 = \text{True}, x_2 = \text{False}, x_3 = \text{True}$  のとき真になる

- なお、最初のNP完全問題は定義に従い示す必要がある

- 充足可能性問題 (SAT) : 与えられた論理式を真にする論理変数の割り当てが存在するかを判定する問題

## NP完全性の証明の例：

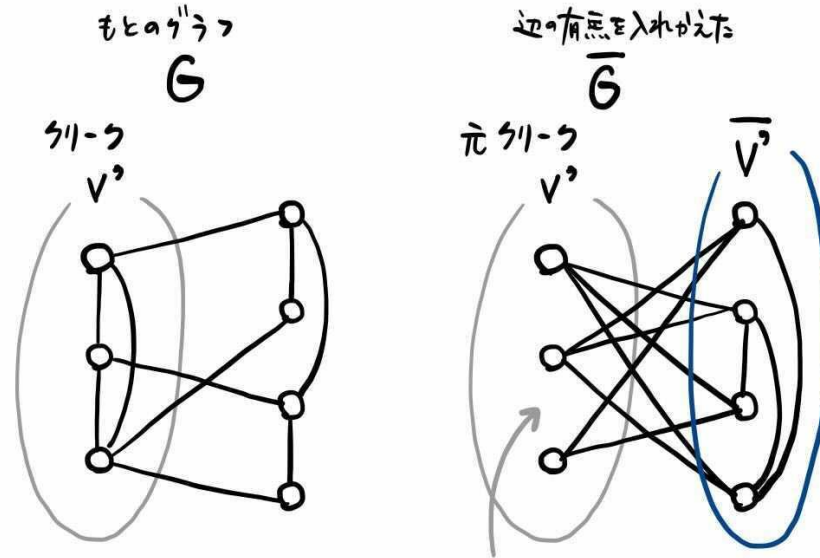
クリーク問題→頂点被覆問題→集合被覆問題の順に帰着

- クリーク問題がNP完全であることは既に分かっているとする
- 頂点被覆問題はNP完全
  - クリーク問題から頂点被覆問題に多項式時間帰着
    - もとのグラフ $G$ で、辺のあるところとないところを逆転したグラフ $G'$ 上での、（サイズ $|V| - k$ の）頂点被覆問題
- 集合被覆問題はNP完全
  - 頂点被覆問題から集合被覆問題に多項式時間帰着



# NP完全性の証明の例： クリーク問題→頂点被覆問題→集合被覆問題の順に帰着

クリーク問題がNP完全であることは既に分かっている

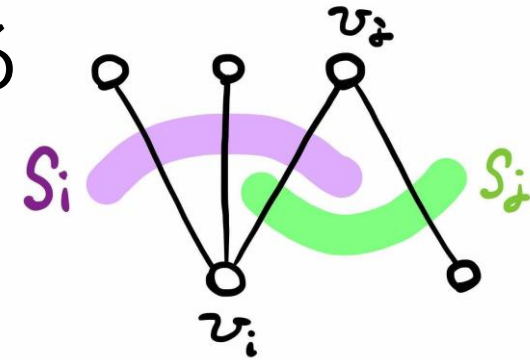


どちらにも辺がないので  $V'$  の頂点は被覆に不要  $\rightarrow \bar{V}'$  の頂点 ( $|V| - k$  個) で被覆できる。  
逆に、 $\bar{V}'$  が頂点被覆ならば、 $G$  において  $V'$  内に辺はない。

$G$  において  $V'$  がサイズ  $k$  のクリークをなす  $\leftrightarrow \bar{G}$  において  $\bar{V}'$  がサイズ  $|V| - k$  の頂点被覆

# 集合被覆問題のNP完全性： 頂点被覆問題からの帰着法

- 集合被覆問題のNP完全性を、頂点被覆問題からの帰着によって証明する
- $V$ の頂点 $v_1, \dots, v_n$ のそれぞれに対して、 $v_i$ に接続するすべての辺の集合を $S_i$ とする  
(頂点被覆において $v_i$ が被覆できる辺の集合)
- 頂点被覆問題  $\Rightarrow S_1, \dots, S_n$ に対する集合被覆問題
  - 頂点 $v_i$ を選ぶことと、辺集合 $S_i$ がカバーされることが対応
- 問題の変換は $O(|V|^2)$ でできる



## NP困難：

NPのどの問題と比較しても、それ以上に難しい問題

- NP困難問題 $Q \doteq$  NP完全問題に対応する最適化問題
  - NPの任意の問題が多項式時間で $Q$ に変換できる  
(必ずしも $Q$ がNPに入っている必要はない)
  - 判定問題とは限らない問題 $Q$ が、  
NP完全問題 $Q'$ を部分問題として含むもの
- 巡回セールスマン問題：  
ハミルトン閉路のなかでコストが最小のものを求める問題
  - これが解ければついでにハミルトン閉路も解ける
- 同様に、最大クリーク、最小頂点被覆、最小集合被覆、...

## まとめ： 問題の難しさ

- 問題の難しさ：その問題の最良のアルゴリズムの計算量
- クラスP：多項式時間で解ける問題クラス
- クラスNP：好きなだけ並列化できる仮想的な計算機で多項式時間で解ける問題クラス
- $P \neq NP$ 予想：おそらく成り立つ
- NP完全：多項式時間で（おそらく）解けない問題クラス
- NP困難：NP完全の最適化版みたいなもの