アルゴリズム論

2010年度

問題のクラス **P** と **NP** [2]

判定問題と最適化問題

- 判定問題を高速に解く方法があれば，多くの場合，最適化問題も高速に解ける
- 判定問題がクラス **P** に属す問題であれば，多くの場合，最適化問題もクラス **P** に属す

- 従って，問題の難しさ（易しさ）を考えるときは，判定問題だけを議論する

多項式時間アルゴリズム

- 個別問題のサイズの多項式オーダーの時間計算量をもつアルゴリズム
  - 易しい問題には多項式時間アルゴリズムがある
  - 多項式は乗算や合成で閉じている
  - 計算機械のモデルに依存しない

- 多項式時間アルゴリズムで解くことのできる問題のクラス（集まり，集合）を **P** と表す
  → クラス **P** に属す問題は易しい問題

クラス **P** に属す問題

- オイラーグラフ判定問題：$O(m)$　m は枝数
- ソーティング問題：$O(n \log n)$　n は要素数
- 二つの n 次正方行列の積の計算：$O(n^{2.81})$
- 全頂点間の最短経路を見つける問題：$O(n^3)$
- ２頂点間の最短経路を見つける問題：$O(n^2)$

　……

☆ オイラーグラフ判定問題以外は“判定問題”ではないので，厳密にはクラス **P** に属すとは言い難い。

☆［最短経路判定問題］指定した２頂点間の距離は L 以下か？

O($N^{2.81}$) の行列積

- 行列のサイズが２のべき乗でないときは？

  $A \in \mathbb{R}^{m \times m},\ m \neq 2^n$ のとき, 必要なだけ 0 を加えて

$$\underbrace{\begin{bmatrix} A & 0 \\ 0 & 0 \end{bmatrix}}_{2^n} \underbrace{\begin{bmatrix} B & 0 \\ 0 & 0 \end{bmatrix}}_{2^n} = \underbrace{\begin{bmatrix} C & 0 \\ 0 & 0 \end{bmatrix}}_{2^n} \qquad A, B, C \in \mathbb{R}^{m \times m}$$

$$m = 100 \Rightarrow n = 7 \quad (2^7 = 128) \qquad m = 1025 \Rightarrow n = 11 \quad (2^{11} = 2048)$$

- 実際にどれくらいのサイズで有利になるか？

計算結果をファイルに出力

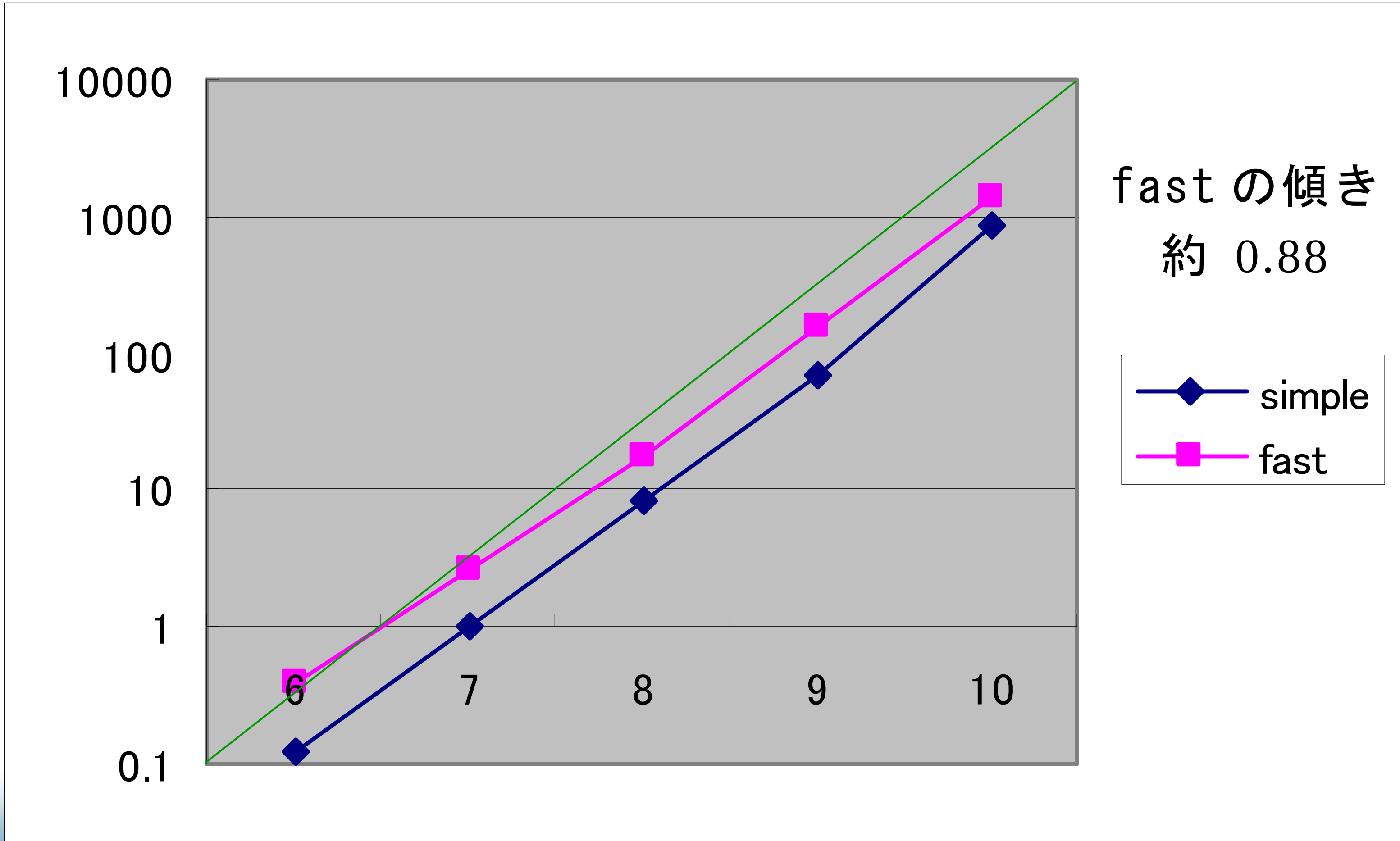
```
def experiment(sizes, values, fileN):
    f = open(fileN, "w")                      # w は write
    print >>f, datetime.datetime.now()        # 実験日時
    for n in sizes:               # n は２のべき。n = 10 が限度
        print >>f, "size = 2**%d = %d" % (n, 2 ** n)
        A = randM(2 ** n, values)             # ランダムな行列生成
        B = A[:]                              # A のコピー
        random.shuffle(B)                     # B をシャッフル
        print >>f, "simple: ", simplePT(A, B)
        print >>f, "  fast: ", fastPT(A, B)
    f.close()
    return
# usage: experiment([6, 7, 8, 9], [-1, 0, 1], "output.rslt")
```

補助関数

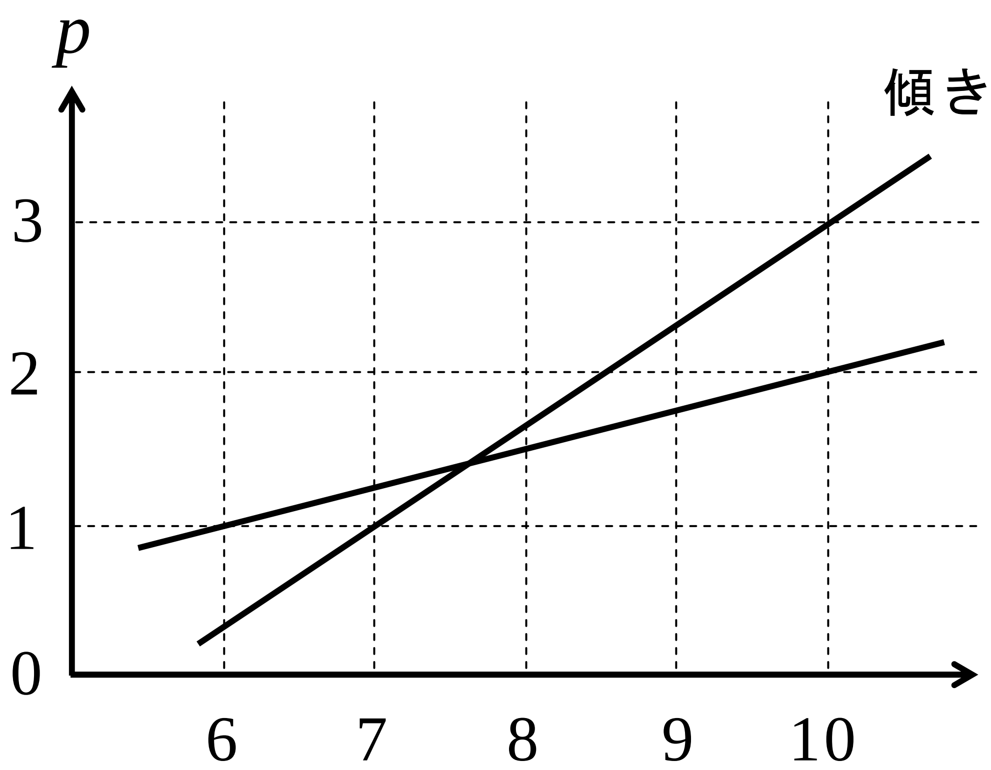
```
def randM(size, values):
    A = []
    for k in range(size * size):
        A.append(random.choice(values))
    return A

def simplePT(A, B):
    start = datetime.datetime.now()
    multiple(A, B)
    delta = datetime.datetime.now() - start
    return delta.seconds + delta.microseconds / 1000000.0
```

関数 fastPT(A, B) も同様

計算時間の比較



グラフから読み取れるもの



傾き $\frac{2}{3}$： $p=\frac{2}{3}n-\frac{11}{3}$

傾き $\frac{1}{4}$： $p=\frac{1}{4}n-\frac{1}{2}$

実験式

$$p = an + b \approx an \qquad\qquad \lg N \equiv \log_2 N$$

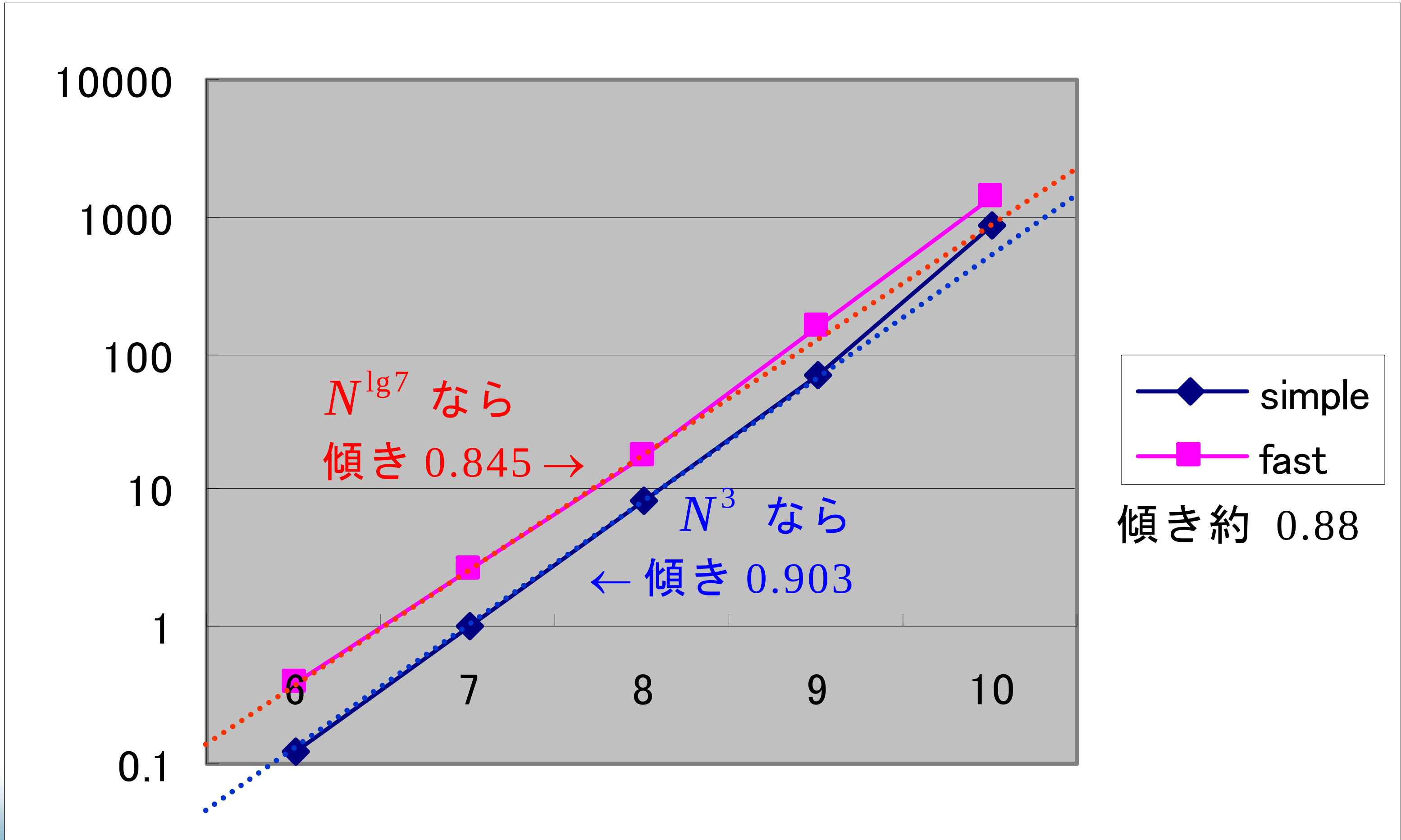
$$T(n) = 10^{p} \approx 10^{an} = 10^{a \lg N} = 10^{\lg N^{a}}$$
$$= (N^{a})^{\lg 10} = (N^{a})^{3.32} \approx N^{0.88 \times 3.32} \approx N^{2.93}$$

理論上は …

$$T(n) = N^{a \lg 10} = N^{\lg 7} \Rightarrow a = \lg 7 / \lg 10 \approx 0.845$$

$$T(n) = N^{3} \Rightarrow a = 3 / \lg 10 \approx 0.903$$

計算時間の比較


10000
1000
100
10
1
0.1
6
7
8
9
10
$N^{\lg 7}$ なら
傾き 0.845 →
N^3 なら
← 傾き 0.903
simple
fast
傾き約 0.88

クラス NP とは？

判定問題 Q に関する次のようなアルゴリズム A が存在するとき，Q は**クラス NP** に属するという

① Q の判定が "yes" となる個別問題 I に対して，その証明書 C_I が存在して，I と C_I を A に入力すると "yes" と判定する

② Q の判定が "no" となる個別問題 I に対しては，① のような証明書はとれなくてよい

③ アルゴリズム A は多項式時間で演算する

- 個別問題 instance （具体例）
- アルゴリズム A は判定問題 Q を解くのではなく，証明書の正しさを判定するだけ

クラス NP の解説

- クラス NP とは，証明書があれば，答が正しいかどうかを容易に(多項式時間で)判定することができる問題のクラス
- 答が "yes" である個別問題に対して，その証明書を作る効率の良いアルゴリズムは要求していない
- 例え話：難しい定理の証明を考え出すことを要求していない。誰かが考えた証明が正しいかどうかを容易に確かめることができればよい

$\mathbf{P} \subseteq \mathbf{NP}$

- クラス $\mathbf{P}$ に属する判定問題 Q においては，任意の個別問題 I に対して，″yes″ か ″no″ かを判定する多項式時間アルゴリズム A が存在するので，わざわざ証明書を発行する必要はない。

従って

$Q \in \mathbf{P} \Rightarrow Q \in \mathbf{NP}$ であり

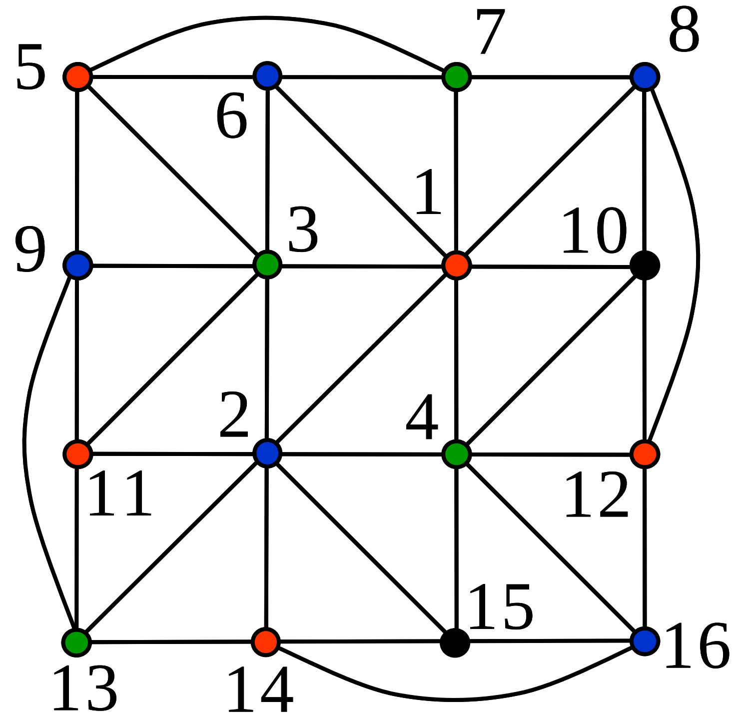
$\mathbf{P} \subseteq \mathbf{NP}$ である

ただし

$\mathbf{P} \subset \mathbf{NP}$（真部分集合）かどうかは不明

k-彩色問題はクラス NP

- このグラフは４-彩色問題で "yes" となる
- その証明書は …
- この証明書が正しいことは容易に確かめられる



クラス NP に属す問題

- 独立集合問題 (independent set problem)
  ［入力］グラフ G と正整数 k
  ［性質］k 個の頂点からなる集合 S で，S に含まれる頂点は互いに（どの２頂点も）隣接していない（枝で結ばれていない）ようにできる

- クリーク問題 (clique problem)
  ［入力］グラフ G と正整数 k
  ［性質］k 個の頂点からなる集合 S で，S に含まれる頂点は互いに隣接しているようにできる

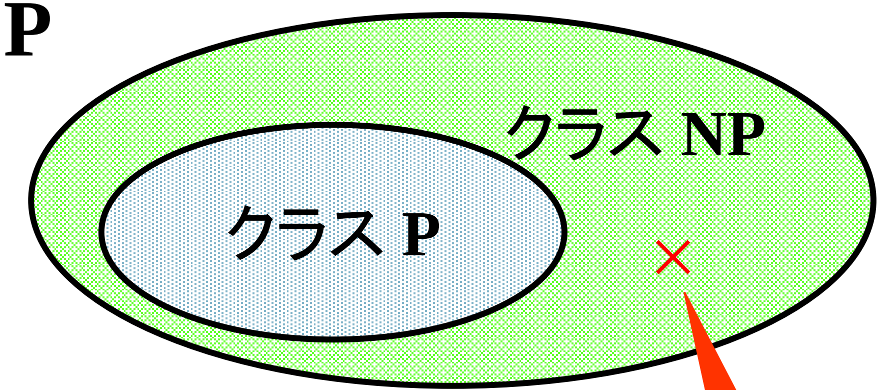
クラス NP に属す問題

- 巡回セールスマン問題 (traveling salesperson problem: TSP)
 ［入力］n 都市と都市間の距離行列，および制限距離 L
 ［性質］n 都市の全部を回って出発地に戻ってくる道順で，その距離が L 以下になるものがある

その証明書は？

問題のクラス

$\mathbf{P} \subseteq \mathbf{NP}$



$\mathbf{P} \subset \mathbf{NP}$?

$\mathbf{P} = \mathbf{NP}$?

$A \subset B \Leftrightarrow A$ は B の真部分集合

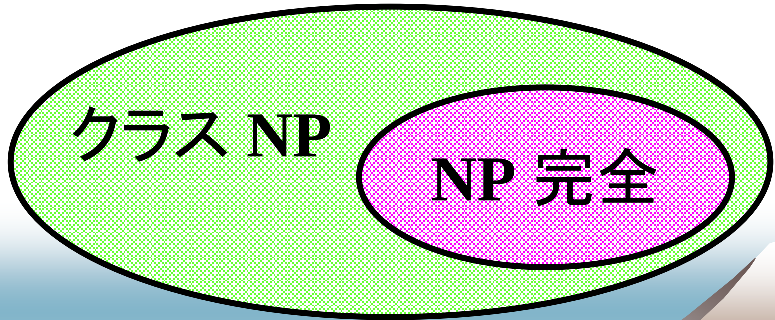
$\Leftrightarrow x \in B$ かつ $x \notin A$

となる要素 x がある

$\mathbf{P} \subset \mathbf{NP} \Leftrightarrow \mathbf{NP}$ に属し, $\mathbf{P}$ に属さない 問題 がある

NP-complete（NP完全）

- どれも難しそうで，いまのところ多項式時間アルゴリズムが発見されていない問題
- クラス **P** に属さないことは証明されていない
- もし，**NP** 完全であるどれか一つの問題がクラス **P** に属することが証明されたら，クラス **NP** に属するすべての問題もクラス **P** に属す（ **P** = **NP** ）ことが保証されている



課題 c1：クラス NP

- クラス **NP** に属す問題を３つ探せ
  - 最適化問題ではなく，判定問題であること
  - ［入力］個別問題として何が与えられるか
    ［性質］どのような性質があることを判定するのか
  - もしわかれば，**NP** 完全であるかどうかも
  - 通常は，簡潔な和英の名称がある
  - 出典を明記：書名／URL／論文誌(Vol. No. pp.)

☆ **NP** 困難 (NP-hard) な問題：その問題を解く多項式時間アルゴリズムがもし存在すれば，**P** = **NP** であることが保証されるが，クラス **NP** には属さない(判定問題ではない)問題。