PとNPの半世紀

茨木　俊秀

クラス P とクラス NP が等しいかどうかは，計算の複雑さの理論における最大の未解決問題として知られているだけでなく，OR に現れる多くの問題がやさしいのかそれとも困難なのかに関わる問いでもあるので，その結末は気になるところである．P と NP の概念はおよそ半世紀の歴史をもっている．本稿では，その理論の発展の流れをざっと追っかけてみたい．

キーワード：P，NP，P ≠ NP，未解決問題

1. はじめに

P = NP なのか，それとも P ≠ NP なのかという問いは，21 世紀中に解決したい 7 つの数学問題の一つに含まれていたためすっかり有名になってしまった．これは OR の研究者，とくに最適化，その中でも離散最適化の研究者にとって，切っても切れない話題でもある．クレイ数学研究所が掲げた 7 つの問題は，素数に関連するリーマン予想やトポロジーのポアンカレ予想，乱流理論のナビエ・ストークス方程式など，純粋数学と数理物理学の話題の中で，P ≠ NP 予想のみが異質であってコンピュータ・サイエンスに関係している．7 問題のうち，ポアンカレ予想がすでに肯定的に解かれたことは，テレビの番組に取り上げられたりしたので，ご存知の方も多いと思う．

さて，P と NP の概念が意識され始めて，およそ半世紀になる．その間，計算の複雑性理論，NP 完全性，P ≠ NP 予想など重要な話題が次々と生まれ，それらの解決の糸口を見つけるという目的に，アルゴリズム理論もさまざまな方向に進歩発展した．これらが扱う具体的な対象には，線形計画 (LP)，整数計画 (IP)，グラフ・ネットワーク最適化，スケジューリングなど，OR にとって身近な話題が多数含まれている．私は一研究者として，この新しい分野に身を置き，大した貢献はできなかったものの，一つの分野が誕生し発展していく様子を見，また体験することができた．大変幸せに思っている．本稿では，この半世紀の間に理論がどのように形成され，現在どの位置にあるかを簡単に述べることにする．

いばらき　としひで
京都情報大学院大学
〒 606–8225 京都府京都市左京区田中門前町 7

2. 1960 年代，LP から IP へ

1960 年代に入ると，トランジスタ式の大型コンピュータが現実のものとなり，科学技術計算に革命的な変化をもたらした．たとえば，LP に対して G. B. Dantzig が 1950 年代に提案したシンプレックス法は，石油産業など広範な領域で利用され，実用的アルゴリズムとして大きな成果をあげた．LP の次に注目を浴びたのが IP である．IP 問題は，LP 問題に含まれる変数の一部あるいは全部に整数条件を加えたもので，離散最適化問題を扱うことができるため，応用範囲は格段に広がる．IP 問題を効率的に解くことができれば，産業や社会生活に与える影響はきわめて大きい．

IP のアルゴリズムは 1960 年代，R. E. Gomory による切除平面法 [1] に始まった．これは，LP の制約条件に，実数最適解は除去するが整数最適解は除去しないという切除平面制約を次々と追加するというアイデアで，シンプレックス法のエレガントな拡張になっている．いくつかのタイプの切除平面法に対して，Gomory は有限収束性の証明も与えた．世間では，一瞬，これで IP も LP と同じように処理できるという楽観論が支配したが，実際に用いてみると，収束がきわめて遅くて，簡単には実用にならないことが次第にわかってきた．当時，この方法で試みられた問題に巡回セールスマン問題があるが，街の数が数十程度の問題例でも，実用的な計算時間では最適解に到らないことが多いと報告されている．

ところで，当時私がどうしていたかと言うと，イリノイ大学の室賀三郎先生の研究室に滞在して，研究のお手伝いをさせてもらっていた．室賀先生はコンピュータの論理設計の権威であり，また，神経細胞ニューロンの動作を抽象化した「しきい論理」の創始者の一人としても知られている．先生は，論理設計に IP が使え

るのではないかと考えておられ，一緒に勉強させてもらったことが，私にとっては，最適化へ踏み入るきっかけになった．論理設計の問題に Gomory のアルゴリズムを試みたが，やはりあまり成功しなかった．論理設計のような 0-1 の問題には，ルーマニアの研究者，E. Balas による加法的アルゴリズム [2] が適しているということも聞き，これも試した．これは今でいう分枝限定法のアイデアである．確かに切除平面法よりかなりよい成績を与えたが，まだ実用には遠いというのが，その時点の結論だった．

このような流れを受けて，研究者たちの間ではさまざまな疑問が生まれ，解決へ向けてのマグマが蓄積しつつあった．最短路問題など効率よいアルゴリズムを持つ問題はたくさんあるのに，なぜ巡回セールスマン問題や論理設計問題などの IP 問題はそうではないのか，LP と IP に本質的な違いがあるのか，その前に研究の出発点として実用的に解けるという意味を厳密に定義する必要があるのではないか，などである．これらが，その後の展開につながっていくのである．

3. 問題の定義とその計算量

アルゴリズムの効率を議論するには，まず計算量を客観的に評価するための基準が必要である．コンピュータが対象とする問題は，通常，多数の（一般には無限個の）問題例からなっていて，どのような問題例に対しても正しい答えを与えることが要求される．典型的には，次のように書かれる．

問題 X
入力：問題例を記述するデータ．
出力：データが指定された性質をみたす解を持つならば yes，そうでなければ no．

入力のデータは，問題 X の規約をみたすものでなければならない．たとえば X がグラフ問題ならば，データは問題例であるグラフの点と辺を記述することが要求される．LP 問題なら，目的関数を示す 1 次式と制約条件を示す 1 次不等式群の係数がデータであり，IP 問題ならば，さらにどの変数が整数変数であるかを示さねばならない．出力の「指定された性質」とは，X の解がみたすべき性質で，たとえば，X がハミルトン閉路問題というグラフ問題であるとすると，全点をちょうど 1 度ずつ訪問して元に戻る閉路が存在すれば yes，存在しなければ no を出力することになる．

なお，ここでは，後の議論を簡単にするために，yes，no の答えを要求する判定問題に限ったが，OR でよく扱われる最適化問題も，目的関数の設定値 c を導入して，c 以下の解が存在するか？という形に書けば，判定問題になる．さらに答えが yes であるような c の最小値を探索すれば，最小化問題を解くことができるので，両者に本質的な違いはない．

さて，問題 X を解くアルゴリズムの計算量であるが，問題例には小さなものから大きなものまであるので，一つの問題例について何分，何秒かかったといってもあまり意味はない．そこで入力データのサイズ（データの文字を 1 列に並べたときの長さ）N を基準にとって，計算量が N の関数としてどのように表されるかを調べることになった．計算量というのは，コンピュータによる実行のステップ数である．これらの定義は，今ならばどの教科書にも載っていて，何の疑問もなく通り過ぎるところであるが，当時は，関連分野の研究者があれこれ議論しながら，一歩ずつ積み上げていったものである．

4. クラス P

以上の準備によって，ようやくクラス P を定義できる．ある問題 X が P に属する ($X \in$ P) とは，X を解く多項式時間量のアルゴリズムが存在することをいう．多項式時間とは $O(N^k)$ と書けるという意味である．ここに N は入力長，k はある定数（N によらない），$O(\cdot)$ はオーダーと読み，中身の定数倍は無視するという意味である．オーダーでは多項式の一番次数の高い項だけが重要なので，要は，ある定数 a，k を用いて aN^k 以下のステップ数であると評価できれば多項式時間である．

調べてみると，以前から効率よく解ける問題として知られていた問題，たとえば，最短路問題，最小木問題，マッチング問題，ネットワークフロー問題などはすべて多項式時間で解ける（つまり，P に属する）ことがわかった．その後（1979 年になって）LP にもシンプレックス法とは異なる多項式時間アルゴリズムが見つかって，P の一員であることが判明した．

クラス P の定義に大きな影響を与えたのは，J. Edmonds による一般グラフの最大マッチング問題に対するアルゴリズム [3] ではないかと思う．この論文は親しみやすいタイトルであるにもかかわらず，内容は大変複雑なアルゴリズムで，グラフ理論を勉強する学生たちは間違いなく苦労した経験を持っているに違いない．彼はこれが $O(n^3)$（n はグラフの点の数）の計算量を持つことを示し，good algorithm であると言った．つまり，多項式時間が実用アルゴリズムの基準であることを主張したのである．

余談になるが，Edmonds の論文はどれも読むのが大変という定評がある．内容自体が複雑だということもあるが，非常にコンパクトに書かれていて，無駄な記述が全くないというのもその理由ではないかと思っている．あるとき，研究仲間に「もう少しわかりやすく書いてもよいのではないか」と苦情をいったところ，「彼は天才だからそれでよいのだ」という返事だった．そういえば，数学者の中には，放浪の数学者として有名な P. Erdös をはじめ，天才・奇人がたくさんおられるが，J. Edmonds も間違いなくその一人と言ってよいだろう．

ところで，多項式時間ならば何であっても実用的であるというと抵抗を感じる人も少なくないに違いない．$O(N)$ とか $O(N^2)$ 程度であれば確かに，$N \to \infty$ のときの計算量の増加速度は大きくないので，コンピュータを使えば結構大規模な問題例まで処理できよう．しかし，$O(N^{100})$ となれば，そうは言えない．そこで，もしたとえば $O(N^2)$ までを実用性の範囲と考え，そのような問題のクラスを P_2 と書くと，実は $\mathrm{P}_2 \neq \mathrm{NP}$ を簡単に示すことができるので，P と NP の問題はそもそも生まれなかったことになる．もちろん，$O(N^2)$ 説はやはり不自然であって，じゃあ，$O(N^{2.1})$ や $O(N^3)$ は実用的ではないのかと問われると困ってしまう．これに対し，多項式時間という概念は，理論家にとっては大変自然である．たとえば多項式の多項式はまた多項式になるという閉包性が成り立つ．結果として，彼らは躊躇なくクラス P に着目して，その解明に邁進していったのである．

5. クラス NP

NP もある性質をみたす問題のクラスであるが，その定義は，非決定性チューリング機械というやや人為的なモデルに基づいている．この機械は，yes–no の問題を解くことを前提としていて，ステップごとに一定数（q と書く）の分岐を許し，そのすべてを「同時に」実行できるという（仮想的な）機能を持っている．したがって，n ステップの計算で q^n 個の分岐路をチェックできることになるが，その中に一つでも yes となる計算路があれば，結論として yes を出力する．計算終了時どこにも yes が出力されていなければ no である．NP はこの機械を使って，多項式時間で解くことのできる問題のクラスと定義される．

ここで，P と NP の名前であるが，P は多項式 polynomial の頭文字，NP は nondeterministic（非決定性）polynomial であって，（まだ！）nonpolynomial のことではないことを注意しておく．なお以後，通常の逐次計算のモデルを上述の非決定性チューリング機械のモデルに基づく非決定性計算と区別して述べる必要のある場合は，決定性計算と書くことにする．

非決定性計算によれば多項式時間でどのような計算ができるだろうか．まず，n 次元の 0-1 ベクトルは 2^n 個あるが，そのすべてを n ステップで生成することができる．また，n 要素の順列のすべて（$n!$ 個ある）も，少し工夫すれば $n \log n$ ステップで生成できることがわかる．離散問題の多くは，すべての解をチェックするという列挙法で解くことができるが，これを決定性計算で実行すると，解の数が多いため，多項式時間には収まらない．しかし，非決定性モデルだと，上述のように解の列挙が多項式時間で可能であるため，それぞれの解について判定すべき性質を決定性多項式時間で調べることさえできれば，多項式時間である．実際，クラス NP が注目されたのは，OR に現れる問題の多くが列挙法で解けるので，NP に属するからである．

定義から明らかなように，非決定性計算は決定性計算より強力である．したがって，集合として $\mathrm{P} \subseteq \mathrm{NP}$ が成り立つ．$\mathrm{P} \neq \mathrm{NP}$ 予想は，NP の問題の中には決定性の多項式時間では解けないものが存在すると主張しているのである．

NP の任意の問題は，決定性計算の指数時間 $O(k^{p(N)})$（ただし，k は定数，$p(N)$ は N の多項式）をかければ列挙法で解くことができるので，指数時間で解くことのできる問題のクラス EXP と混同されることがある．しかし，NP は EXP に比べると計算能力がかなり限定されているため，EXP の問題で NP ではないものが存在して，$\mathrm{NP} \neq \mathrm{EXP}$ であると予想されているが，これも未解決である．ただし，$\mathrm{P} \neq \mathrm{EXP}$ は既知であって，P と EXP の間では P と NP のような問題は生じない．

$\mathrm{P} \neq \mathrm{NP}$ 予想の本質は，

「解の発見は解の検証より難しい」

ことを証明することにある，とも説明される．NP の問題では，生成された解のそれぞれについて，それが求められる性質をみたすかどうかの判定は決定性多項式時間で可能，と想定されている．つまり，解の検証は容易だということである．では解の発見はどうかというと，非決定性計算では多項式時間ですべての解を列挙できるので，これも容易であるが，通常の決定性計算では，解を一つずつ全部生成するという手順をとるとすれば多項式時間ではない．つまり，もし，P =

NP であるなら，求められる性質を持つ解を，すべての解を列挙することなしに発見できなければならない．それができない，つまり解の発見は解の検証より難しいというのが，$P \neq NP$ の意味である．

突然話題が変わるが，最近，東野圭吾「容疑者 X の献身」という推理小説を読んだ．ガリレオと呼ばれる大学教授が主人公のシリーズの一つで，直木賞受賞作である．映画とテレビでも放映されたらしい．この中に，4 色問題とかリーマン予想といった数学の話題が出てきて，$P \neq NP$ 問題については，自分で考えて答えを出す（解の発見）のと，他人から聞いた答えが正しいかどうか確認する（解の検証）のではどちらが簡単か，という問いだと説明されている．完全犯罪を考案する犯人と，それを解決する探偵のどちらのレベルが高いか，という議論である．小説の内容（大変よくできている）は原作を読んでもらうとして，推理小説にも，このような数学の話題がスパイスとして用いられる，そういう時代になったらしい．

本節のはじめに，非決定性計算を人為的なモデルと書いたが，理論の世界では以前から決定性計算の自然な拡張と考えられていたようである．非決定性チューリング機械の他にも非決定性有限オートマトンなど種々のモデルが知られている．これらのモデルでは，できるできないの議論では決定性と非決定性の能力に違いはないということはわかっていたが，効率性の観点からは，モデルによって違いが生じる．ここではチューリング機械の多項式時間のところではどうなのか，というのが重要な問いとして提示されているのである．

6. 1970 年代，NP 完全性の登場

ある問題がやさしい（多項式時間で解ける）ことを示すには，それを解くための多項式時間アルゴリズムを与えればよい．方法論としては簡単である．では，ある問題が多項式時間では解けないことを示すにはどうすればよいだろうか．いくら努力してもそのようなアルゴリズムが見つからなかった，では証明にならない．この問いに一挙に答えるわけではないが，その前段階として，二つの問題 A と B の間で，B は A 以上に難しいことを示せる場合がある．つまり，A の任意の問題例 I に対し B の問題例 $f(I)$ を多項式時間で構成でき，しかも I の答え (yes, no) と $f(I)$ の答えが一致するならば，B のアルゴリズムを用いて A を解くことができる．この場合，A は B に帰着可能（reducible, 還元可能とも訳される）であると言い，$A \preceq B$ と記す．B は A 以上に難しいという記号である．なお，この道具自体は，以前から知られていたものである．

ここに，S. A. Cook が登場する．1971 年の論文 [4] で，NP の任意の問題 A に対し，$A \preceq C$ をみたす問題 $C \in NP$ が存在すれば NP 完全 (NP-complete) であると定義した．つまり，C は，NP のどの問題を持ってきてもそれ以上に難しい，換言すれば，クラス NP の中で一番難しい問題であるということである．さらに，ある形式の論理式を充足できるかどうかを判定する充足可能性問題 SAT が NP 完全であることを示した．この研究に刺激を受けて，R. M. Karp [5] は SAT の他にもたくさんの NP 完全問題があることを見出した．先に言及したハミルトン閉路問題，0-1 変数の整数計画問題（後に，一般の整数計画問題も），巡回セールスマン問題，多くのスケジューリング問題などである．Cook の証明は，非決定性チューリング機械の動作に基づくやや複雑な議論によっているが，いったん NP 完全問題が見つかると，その後の NP 完全性の証明は比較的簡単である．既知の NP 完全問題の一つ C を持ってきて，対象とする問題 B が $C \preceq B$ をみたすことと，$B \in NP$ を示すだけで B の NP 完全性の証明が完結するからである．この証明法を用いて，その後，数学のあらゆる分野から，たくさんの NP 完全問題が発見された．たとえば M. R. Garey and D. S. Johnson [6] には数百個の問題がリストアップされている．

クラス NP の中で一番難しいという NP 完全問題が複数個あることは奇異に映るかもわからないが，これは帰着可能性の証明にある f の構成に多項式時間を許しているからで，結果として問題間の多項式時間の差は無視しているわけである．

どの分野でもそうだと思うが，使われている用語が定着するにはある程度の時間が必要である．上の議論の要である NP 完全という用語も，概念自体にも微妙な揺れがあって，定着までにかなりの曲折を経ている．Cook や Karp の頃は多項式完全 (polynomial complete)（NP 完全問題は互いに多項式時間で帰着できるので）と呼ばれていたが，それだと NP 完全問題も P に入るという印象を与えかねないので，D. Knuth が確か ACM のニュースレターで意見を募り，最終的に NP 完全という名称を全員が使うようになった．この辺の経緯は，文献 [6] に詳しく述べられている．

ここで，これまでに出てきた問題クラスの包含関係を図示しておく．クラス P と EXP の境界が点線になっているのは，NP と一致する可能性が残されているからである．

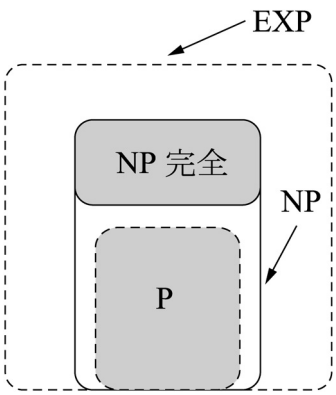

図 1　クラス間の包含関係

7.　P ≠ NP 予想とその先

NP 完全問題の発見は，P と NP の議論を大いに単純化するものである．もし，NP 完全問題の一つ C に対し，それを多項式時間で解くアルゴリズムを見つけることができれば，定義からわかるように NP のすべての問題が多項式時間で解けることになる．P = NP である．一方，多項式時間では解けない NP の問題が一つでも存在する（つまり P≠ NP）とすれば，どの NP 完全問題も多項式時間では解けないわけだから，そのような候補を NP 完全問題に限定することができる．

実際，いくつかの NP 完全問題に対して，それを解く多項式時間アルゴリズムを見つけたという主張（つまり，P = NP）が現れたが，すべて間違いであることが判明している．一方，NP 完全問題は数学の広い分野に存在しているが，そのどれもがその分野における過去の研究の中で解決困難な問題として認識されているものばかりであって，これらのすべてが多項式時間で解けるとは到底考えられない．これは P ≠ NP であることの強力な根拠である．その結果，この分野の研究者のほとんどは P ≠ NP 側に立っていると思われる．

ところで，仮に P ≠ NP であったとしても，現実の問題を解決しなければならない OR にとって，それで終わることはできない．OR に現れる多くの NP 完全問題を目の前にして，立ちはだかる NP の壁を何とか乗り越える手立てを考えねばならない．

1970 年代の中頃，前出の Karp 先生を訪ねたことがある．てっきり先生は NP 完全問題の探求をさらに進めておられると思って質問したところ，「自分は実は，NP 完全問題を見つけることはもう卒業して，今はそのような困難な問題をどう解くかに興味がある」というお話だった．実際，その当時，確率的な意味で問題を解くための枠組みを集中的に研究しておられた．一流の研究者はさすが違う，と感銘を受けたものである．

NP 完全問題を現実的に解くためには，まず，問題を解くということの定義を考え直す必要があるだろう．たとえば，すべての問題例を多項式時間で解くことはできなくても，ほとんどの場合実用的な時間で解けるのであれば，十分役に立つに違いない．これは解くことの確率的な解釈にもつながるものである．つぎに，必ず正確な解を求めるという目標を緩和して，近似解で我慢するという設定にするのも一つの可能性である．

このような現実的な目標の下でのアルゴリズムは，非決定性計算が多項式時間でチェックする広大な解空間を，決定性計算を用いてその一部を調べるだけで，現実的に処理すると言い換えてもよかろう．たとえば，n 個の 0-1 変数が作る状態は 2^n 個あるが，その全部でなく，対象問題の条件をみたす解が存在しそうな領域を，問題の構造を利用して「効率的」に選別して探索するということである．この観点に立って新しいアルゴリズムを研究することがその後の大きな流れになり，つぎに述べるように，さまざまな方向に発展した．

8.　1980 年代以後，アルゴリズムの展開

NP 完全問題を扱うためのアルゴリズムの研究は多岐にわたっている．以下，私が興味深いと考えている 4 分野に絞って，ごく簡単に説明しよう．

8.1　IP への正統的アプローチ

IP（整数計画）問題は高い汎用性を持つので，実用的なアルゴリズムを開発できれば意味するところは大きい．1960 年代以降もこの方面の研究は強力に進められ，現在では，多項式時間とは言えないものの，高い効率を持つアルゴリズムが商用パッケージとして開発されるまでになっている．基本的な考え方は，切除平面法に分枝限定法を加味したもので，分枝カット法と総称されている．

分枝限定法とは，まず，n 変数空間を，$x_i \leq a$ と $x_i \geq a+1$ （a はある整数） というタイプの制約によって二つに分割する操作（これを分枝操作という）を反復適用して，全空間を多数の部分領域に分割していく．つぎに，生成された各部分領域に対して，その上の LP 解を求めるなどのテストによって，不要な部分領域をできるだけ早めに発見してその後の考察から除く（これを限定操作という）ことで，探索すべき部分領域を絞っていくのである．分枝操作と限定操作をうまく設計できれば，計算効率を高めることができる．

切除平面も，最適整数解を含まない領域を規定する一つの制約であるから，分枝限定法との親和性は高い．両者の機能を組み合わせた複合アルゴリズムが分枝カット法である．このタイプのアルゴリズムの実用性は高く，開発された商用パッケージによって，多くの現実問題が日常的に解決されるまでになっている．

8.2 乱択アルゴリズム

非決定性計算が多項式時間でたどる計算路のすべてではなく，その一部分を確率的に選んで試す，というアプローチである．乱択 (randomized) アルゴリズムとか確率 (probabilistic) アルゴリズムと呼ばれるものである（厳密には細かい区別がある）．どのパスをどのような確率で選ぶかを，問題の特性を生かしてうまく設計すると，1/2 より大きな確率で正解を得ることができる場合がある．もしそうなら，その試行を反復すれば，正解の確率はいくらでも高めることができるので，言葉から受ける印象よりずっと精度の高いアプローチである．

乱択アルゴリズムは，アルゴリズム設計の新しい可能性の一つとして，研究者の興味を集めた．一般に，乱択アルゴリズムは比較的単純な手順であっても，高い性能を持つことが多い．また，乱択アルゴリズムから確率によらないアルゴリズムを構成できる場合がある．脱乱択化と呼ばれているが，これが新しいアルゴリズムに導く例も報告されている．

8.3 メタ・ヒューリスティクス

解くべき問題の構造を解析して，解が存在しそうな領域を特定し，その周辺を集中的に探索する方法である．今調べている探索解の近傍内に改良解が存在すればそれに移動するという操作を反復する，いわゆる局所探索のアイデアが基本になっている．しかし，具体的にアルゴリズムを構成するには，近傍をどう定義するか，探索解の生成法をどうするか，探索解を一つだけ持つかそれとも複数の解を並行して探索するか，探索にランダム性を組み込むかどうか，改良解が存在しない場合でも何がしかの基準で改悪解に移動することを許すのはどうか，局所探索を何度も反復して精度を高める，など，種々の変形が提案されていて，多様なアルゴリズムが含まれ，メタ・ヒューリスティクスと総称されている （詳しくは [7] など参照）．代表的なアルゴリズムとして，タブー探索，遺伝アルゴリズム，アニーリング法などがこの範疇に入る．これまで数多くの NP 完全問題に対して，この原理に基づく近似アルゴリズムが開発されていて，実用的に大きな成功を収めている．

8.4 量子計算

量子力学の世界では，非決定性計算が生成する指数個の状態が重なった状態を物理的に実現できるという．観測によって，その中の一つを出力できるが，何が出力されるかは確率的に定まり，観測とともにその時点の状態は破壊されてしまう．そこで，量子力学の法則に則った回路をうまく構成して，正しい解が発見される確率を高めることができれば，対象問題を（確率的に）解くことができる．非決定性計算との違いは，重なっている指数個の状態を個別に見ることはできない点にある．従来のコンピュータとは異なる原理に基づく新しい可能性を提供するものであって，未来の計算モデルの一つと期待されている．

9. ケーススタディ：巡回セールスマン問題

巡回セールスマン問題は，n 個の街を一度ずつ訪問して元に戻る巡回路の中で最短のものを見つけよ，というものである．本稿の最初のところで述べたように，初期の IP アルゴリズムでは，n が高々数十であっても解くことが困難であった．この問題は代表的な NP 完全問題として，またパズル的興味もあって，たくさんの人がチャレンジしてきた結果，この半世紀に大きな進歩があった．たとえば，ウェブの TSPLIB には数十年前からたくさんの問題例が掲示されていて，その最適解を募っていたが，当時の問題例はすべて厳密に解かれたということである．問題例の規模を示す n は，数百から数千，中には数万というものまである．用いられたアルゴリズムは，（大規模な問題例に対しては）すべて分枝カット法と思われるが，切除平面の生成法や探索の仕方，またデータ構造などに，この問題に特化した特殊なアイデアが多数組み込まれている．

と言っても，数万の街を持つ任意の問題例に対して常に最適解を得ることができるという訳ではない．問題例によって困難さに大きな違いがあって，同じ n であっても，計算量が大幅に異なるのが普通である．これは NP 完全問題の一つの特徴と考えられている．

巡回セールスマン問題に対する近似アルゴリズムも大きな進歩を遂げている．詳細は略すが，きわめて短時間に最適値から数%の誤差範囲の近似解を求めることができると考えてよい．

10. アルゴリズム工学

巡回セールスマン問題の例からもわかるように，アルゴリズムとコンピュータのハードウエアの進歩は目覚しく，NP 完全問題であっても現実的に対応できる

ようになってきている．しかし，その成果が必ずしも現実の場では活かされていないという思いがあった．1990 年代の頃である．当時から計算の複雑さやアルゴリズム理論の分野には優秀な研究者が多く集まっていて，役に立ちそうな成果も生み出されていたにもかかわらず，現実の応用に携わっている人たちは，あまりそれらに興味を示しているとは思えなかった．そこで，理論的な成果を役に立つ形で示すことを目的に，文部省科研費の特定領域研究に「新しいパラダイムとしてのアルゴリズム工学」というテーマで申請したところ，幸い 1998 年から 3 年間のプロジェクトとして採択された．

このプロジェクトには約 100 名の研究者が参加して，文献 [8] にあるように，多くの研究成果が得られた．少なくとも「アルゴリズム工学」という名前は定着したように思える．これを契機に，当時京都大学にあった私の研究室でも，実用を念頭においたアルゴリズム研究を進めた．同僚や学生たちの努力の結果，いろいろな成果が得られたが，ここで全体に言及することはできない．その中では，柳浦睦憲さん（現，名古屋大学）との前出の共著 [7] や野々部宏司さん（現，法政大学）が主力になって開発した制約充足問題のアルゴリズム [9] など，メタ・ヒューリスティクスに基づく結果が，今も応用サイドの人たちに最も利用されているようである．

11. 現時点でのまとめ

NP の壁を克服するための試みの中から，さまざまなアルゴリズムのパラダイムが提案され，視野は格段に広がった．しかし残念ながら，P $\neq$ NP 問題の解決という観点からは，本質的な進歩があったとは言えない．入力長 N の定数乗ならば何でも良いという P の定義があまりに柔軟性が高いため，NP との違いが存在してもごく僅かなのであろう．その違いを際立たせるための数学的手段が見つからないということである．クラス間の計算量の違いを証明する標準的な道具に対角化論法があって，以前からたとえば P $\neq$ EXP などの証明に用いられてきたが，P と NP に対しては，この方法は無力らしいことが数学的に示唆されている．したがって，これまでにない新しい道具を見つけなければ，解決につながらないのである．

解決への努力は世界中で続けられているに違いないが，国内でも，たとえば文科省科研費の新学術領域の一つとして，東工大の渡辺治教授を代表者として「多面的アプローチの統合による計算限界解明」というプロジェクトが進行している．関係分野から多くの研究者（とくに若手）が集結している．プロジェクトの究極の目的は，当然 P $\neq$ NP の証明であるが，少なくとも頂上への階段を何段かでも登ることができるよう期待している．

P $\neq$ NP 問題が簡単には解決しないことは，NP 完全性の概念が提唱された当時から予想されていた．その頃は 20 世紀中に解決できるだろうか？と言われていたが，すでに 21 世紀に入ってしまい，今世紀中に解決したい代表的な問題として取り上げられるまでになった．ともあれ，21 世紀はまだまだ残されている．いつか，できれば近い将来に，その解決を見たいものである．

なお，本稿は以前，応用数理学会誌に連載した記事 [10] とオーバーラップしている部分がある．興味をお持ちの方は，そちらも読んでいただけるとありがたい．

参考文献

[1] R. E. Gomory, “An algorithm for integer solutions to linear program,” *Recent Advances in Mathematical Programming,* McGraw-Hill, pp. 269–302, 1963.

[2] E. Balas, “An additive algorithm for solving linear programs with zero-one variables,” *Operations Research,* **13**, 517–546, 1965.

[3] J. Edmonds, “Paths, trees, and flowers,” *Canadian Journal of Mathematics*, **17**, 449–467, 1965.

[4] S. A. Cook, “The complexity of theorem-proving procedures,” *Proceedings of 3rd Annual Symposium on Theory of Computing*, 151–158, 1971.

[5] R. M. Karp, “Reducibility among combinatorial problems,” *Complexity of Computer Computations*, Plenum Press, pp. 85–103, 1972.

[6] M. R. Garey and D. S. Johnson, *Computers and Intractability*, W. H. Freeman and Co., 1979.

[7] 柳浦睦憲，茨木俊秀，『組合せ最適化―メタ戦略を中心として―』，朝倉書店，2001.

[8] 杉原厚吉，茨木俊秀，浅野孝夫，山下雅史（編），『アルゴリズム工学』，共立出版，2001.

[9] K. Nonobe and T. Ibaraki, “An improved tabu search method for the weighted constraint satisfaction problem,” *INFOR*, **39**, 131–151, 2001.

[10] 茨木俊秀，“NP 困難性の 35 年，”応用数理，**17** (1–4), 2007 年 3, 6, 9, 12 月．

