Vol. 31 No. 4　　情　報　処　理　　Apr. 1990

解　説

実際的計算可能性の拡張について†

戸田誠之助††

1. はじめに

従来の計算量理論では，与えられた問題が逐次的な多項式時間アルゴリズムをもつとき「実際的に計算可能」であると考え，実際的に計算可能な問題と計算困難な問題との境界を明確にすることを主要課題の一つとしている．この課題を研究する枠組みの一つとして ***NP*** 完全性の理論が出現し，多くの工学的・数学的諸問題が ***NP*** 完全となることを示すことによって，それらが計算困難であることの根拠を与えてきた．一方，情報処理技術の発展にともない，さまざまな問題に対して並列アルゴリズムや確率的アルゴリズムが提案されるようになり，逐次的には計算困難な問題に対しても並列的または確率的に有効な解法が存在するのではないかとの期待がもたれるようになった．最近の計算量理論では，並列計算や確率計算に基づく実際的計算可能性の形式的概念やより一般的かつ理論的な概念を導入することによって，この期待に関する議論が行われている．その議論の最終目標の一つは，「***NP*** 完全問題が拡張された意味で実際的に計算可能ならばそれは逐次的にも実際的に計算可能である」ことを示すことであり，多くの研究者はこれが成立することを予想している．

本稿では，計算量理論を専門としない読者のためにこの議論の内容を可能なかぎり直観的に解説する．まず，**2.** では，基本的な用語や記法について直観的に説明する．**3.** では，確率的計算及び並列計算のモデルを説明した上で，これらのモデルに基づく実際的計算可能性の概念を導入する．**4.** では，中心的な研究課題を提示し，かつ，その問題をより深く考察するための理論的概念を導入する．また，現在までに得られている研究成果の一部を紹介する．最後に，本稿で述べる研究領域を勉強してみようと思われた方のために，参考文献の簡単な案内を示す．

2. 準　備

問題の計算上の複雑さを論じるためには，形式的な計算モデルを導入し，そのモデルに基づいて複雑さの尺度を定める必要がある．通常，チューリング機械と呼ばれるモデルが使用されるが，ここではより直観的に通常のプログラミング言語を計算モデルとして考えることにする．複雑さの尺度としては，計算時間量のみを扱うことにする．プログラミング言語でもっとも初等的と考えられる演算もしくは実行文（たとえば，単精度整数間の加算・比較など）の実行に要する時間を1単位時間とし，プログラム全体の実行時間はこれを基準にして計ることとする（CPU の1クロックを1単位時間としてもよい）．プログラム中で処理するデータは，すべて0と1からなる有限2進列とする．直観的には，計算機内での各データの内部表現をそのまま扱っていると考えてよい．とくに，入力データを2進列に制限することは「問題の計算時間量」をより一般的に定義する上で本質的である．

本稿で扱う「問題」は，各入力データに対してある種の条件が成立するか否かを判定する判定問題のこととする．判定問題は2進列上の述語として形式化される．たとえば，次の最長路問題を考えてみよう：無向グラフ G, G の二つの頂点 a と b，自然数 $k \geq 0$ が与えられたとき，G 上の a と b を結ぶ単純道（同じ頂点を2度以上通過しない道）でそれに含まれる辺の個数が k 個以上のものが存在するか否かを判定せよ．この問題は，$L(G, a, b, k)$＝真となるときかつそのときにかぎり G, a, b, k が上の条件を満たす，と定義される述語 L のことと考えればよい．

問題の計算時間量は，入力データの長さの関数として定義される．今，t を自然数上の関数とする．**プログラム *Q* が $t(n)$ 時間限定** ($t(n)$ time bounded) である（すなわち，Q の計算時間量が高々 $t(n)$ である）

† Some Extended Notions of Tractability by Seinosuke TODA (Department of Computer Science and Information Mathematics, University of Electro-communications).

†† 電気通信大学電気通信学部情報工学科

とは，任意の入力データ x に対して，Q が $t(|x|)$ 単位時間以内に停止するときをいう．ここで，$|x|$ は入力データ x の長さを表す（なお，入力データが複数の2進列からなるときは，それらの長さの総和を入力データの長さと考える）．**問題 A が $t(n)$ 時間限定**であるとは，A を計算する $t(n)$ 時間限定のプログラムが存在するときをいう．

次に，問題のクラスを幾つか定義しよう．多項式時間限定の問題からなるクラスを $\boldsymbol{P}$ と表す．$\boldsymbol{P}$ に属する問題は，どちらかといえば計算しやすい問題といえる．一方，計算困難な問題を含むクラスの代表として $\boldsymbol{NP}$ がよく知られている．ここでは，論理学（特に第1階述語論理）の用語を借りて，やや天下り的に $\boldsymbol{NP}$ を定義しよう．さきに述べた最長路問題が $\boldsymbol{NP}$ に属することが知られているので，少しの間，この問題を考える．まず，述語 A を次のように定義する：$A(G, a, b, k, w) \Leftrightarrow w$ は k 個以上の辺を含む頂点 a から頂点 b への（無向グラフ G 上の）単純道（を表す2進列）である．このとき，最長路問題 L は，論理学の存在記号を用いて，

$$L(G, a, b, k) \Leftrightarrow (\exists w) A(G, a, b, k, w)$$

と定義できることが理解されよう．ここで，束縛変数 w の変域は，2進列全体と解釈されるのが一般的である．しかし，もう少しよく考えてみると，G, a, b, k の長さの総和と w の長さの間には大きな差はないと考えられる．少なくとも次が成り立つと考えてよいであろう：適当な多項式 q が存在して，任意の G, a, b, k, w に対して，$A(G, a, b, k, w)$＝真ならば $|w| \leqq q(|G| + |a| + |b| + |k|)$ が成り立つ．このことより，最長路問題 L を，

$$L(G, a, b, k) \Leftrightarrow (\exists w)[|w| \leqq q(n) \text{ かつ } A(G, a, b, k, w)]$$

と定義するほうがより厳密である（ただし，$n = |G| + |a| + |b| + |k|$）．言い換えると，束縛変数 w の変域に $q(n)$ より長い2進列を加えても，それらは L の真偽判定に関与しないといえる．さらに，この定義において，A が多項式時間限定であることがもう一つの重要な点である．以上をまとめると，最長路問題は，適当な多項式と適当な多項式時間限定の問題によって上述した形式で定義できる，となる．実際に，このことはすべての $\boldsymbol{NP}$ に属する問題について成り立つ．そこで，$\boldsymbol{NP}$ を次のように定義する．

定義　問題 $B(x_1, \cdots, x_m)$ が $\boldsymbol{NP}$ に属するのは，多項式時間限定の問題 A と多項式 q が存在して，任意の $x_1, \cdots, x_m (n = |x_1| + \cdots + |x_m|$ とおく）に対して，

$$B(x_1, \cdots, x_m) \Leftrightarrow (\exists w)[|w| \leqq q(n) \text{ かつ } A(x_1, \cdots, x_m, w)]$$

となるときかつそのときに限る．

論理学では，存在記号と双対な概念として全称記号がある．そこで，存在記号と全称記号を複数個使用して新たなクラスを導入する．まず，存在限量化及び全称限量化に関する略記法を定義する．これまで，多項式 q が与えられたとき，

$$(\exists w)[|w| \leqq q(|x|) \text{ かつ } \alpha(x, w)]$$

という形の論理式がしばしば出現した．そこで，この論理式を $(\exists_q w)\alpha(x, w)$ と略記する．$\exists_q$ と双対な全称記号 $\forall_q$ を

$$(\forall_q w)\alpha(x, w) \Leftrightarrow \neg(\exists_q w)\neg\alpha(x, w)$$

により定める．すなわち，$(\forall_q w)\alpha(x, w)$ は，

$$(\forall w)[|w| \leqq q(|x|) \rightarrow \alpha(x, w)]$$

という論理式のことである．

定義　各 $k \geqq 1$ に対して，問題のクラス Σ_k^p を次により定める．問題 $B(x_1, \cdots, x_m)$ が Σ_k^p に属するのは，多項式時間限定の問題 A と多項式 q が存在して，任意の $x_1, \cdots, x_m$ に対して，

$$B(x_1, \cdots, x_m) \Leftrightarrow (Q_1 w_1)(Q_2 w_2)\cdots(Q_k w_k) A(x_1, \cdots, x_m, w_1, w_2, \cdots, w_k)$$

となるときかつそのときに限る．ただし，各 Q_i は，i が奇数のときは $\exists_q$ を表し，i が偶数のときは $\forall_q$ を表す．以上のクラスを総称して，**多項式時間階層**（polynomial time hierarchy）と呼ぶ．

以上で述べたクラスの間の関係を明確にすることは計算量理論における基本的な課題である．特に，「$\boldsymbol{P} = \boldsymbol{NP}$ となるか否か」という課題は「$\boldsymbol{P} = \boldsymbol{NP}$ 問題」と呼ばれ，最重要課題となっている．ほとんどすべての研究者は，$\boldsymbol{P} \neq \boldsymbol{NP}$ となることを予想しているが，20年以前より今日まで未解決のままである．さらに，多項式時間階層の各クラス間の関係もあまりよく分かっていないが，次のような相対的な結果が知られている．

命題　各 k に対して，$\Sigma_k^p \neq \Sigma_{k+1}^p$ ならば $P \neq NP$.

この命題から，「$\Sigma_k^p \neq \Sigma_{k+1}^p$ か否か」という課題は，第 k 番目の「$\boldsymbol{P} = \boldsymbol{NP}$ 問題」と捉えることもできる．

3. 実際的計算可能性

従来の計算量理論では，「多項式時間限定のプログラムによって計算できる」ことを実際的に計算可能で

あることと同一視してきた．しかしながら，この思想は，近年の情報処理技術の発展にともなって，やや古典的になりつつある．本章では，アルゴリズム論的な観点から，確率的計算及び並列計算に基づく実際的計算可能性の形式的概念を紹介する．

確率的計算のモデルから説明を始めよう．われわれのモデルは，従来のプログラミング言語に乱数発生関数を付加したものである．この乱数発生関数は，無入力で，かつ，0または1を等確率で発生するものである．また，その1回の実行に要する時間を1単位時間と仮定する．以後，混同の恐れがあるときには，従来のプログラムを逐次型プログラムと呼び，乱数発生関数を使用したプログラムを確率型プログラムと呼ぶ．また，議論を単純化するために，必ず停止するプログラムのみを考察の対象とする．

確率型プログラム Q が与えられたとき，入力データ x に関して $Q(x)$=真となる確率を定義しよう．入力データ x に関して Q を実行したとき，Q が停止するまでに乱数発生関数が出力した乱数列を x に関する**履歴**と呼ぶことにする．入力 x に関する Q の履歴の集合を $Z_Q(x)$ と表す．$Z_Q(x)$ は有限2進列からなる有限集合である．$Z_Q(x)$ の各履歴 y が生起する確率を $2^{-|y|}$ と定める．このとき，

$$\sum_{y\in Z_Q(x)} 2^{-|y|}=1$$

となることが示せる．さらに，各履歴ごとに Q の出力結果は一意的に定まる．したがって，入力データ x に関して Q が真を出力する履歴の集合は，確率論的にいうと，標本空間 $Z_Q(x)$ 上の確率事象になる．そこで，入力データ x に関して，**Q が真を出力する確率** $Pr(Q(x)=\text{真})$ を

$$Pr(Q(x)=\text{真})=\sum_{y\in T} 2^{-|y|}$$

と定める．ここで，T は真を出力する履歴の集合である．**Q が偽を出力する確率** $Pr(Q(x)=\text{偽})$ も同様に定義される．

以上の定義に基づいて，***BPP*** (Bounded Probabilistic Polynomial time の略) という問題のクラスを定義する．問題 A が ***BPP*** に属するのは，多項式時間限定の確率型プログラム Q が存在して，任意の x に対して，

$A(x)$ ならば $Pr(Q(x)=\text{真})\geqq 0.9$ かつ
$\neg A(x)$ ならば $Pr(Q(x)=\text{偽})\geqq 0.9$

となるときかつそのときに限る．ただし，Q が多項式時間限定であるとは，任意の入力 x に関する Q のどのような履歴についても必ず（$|x|$ の）多項式時間以内に停止するときをいう．より直観的に言うと，問題 A が ***BPP*** に属するのは，計算時間上効率的で，かつ，信頼性の高い確率型プログラムによって計算できるときである．この「***BPP*** に属すること」を実際的計算可能性の拡張概念の一つと考えることにする．なお，上で示した信頼性の下限値 0.9 は便宜的なものであり，1に近いどのような値（たとえば，$1-10^{-10}$）を設定しても ***BPP*** の定義に影響を与えないことを注意しておく．**4.** の結果を述べるために若干変わったクラスを定義しておく．問題 A が ***RP*** (Random Polynomial time の略) に属するのは，多項式時間限定の確率型プログラム Q が存在して，任意の入力データ x に対して，

$A(x)$ ならば $Pr(Q(x)=\text{真})\geqq 0.9$ かつ
$\neg A(x)$ ならば $Pr(Q(x)=\text{偽})=1$

となるときかつそのときに限る（ここでも，0.9 は便宜的な値である）．定義から明らかに，$\boldsymbol{RP\subseteq BPP}$ が成り立ち，また，$\boldsymbol{P\subseteq RP\subseteq NP}$ が成り立つ．しかし，これらのクラスが異なるか否かは知られていない．

次に並列計算について考える．近年の並列計算技術の進展にともなって，計算量理論においてもさまざまな並列計算モデルが提案されている．並列計算モデルにおいて注目される**複雑さの尺度**は，計算時間量とハードウェア量である．ハードウェア量の典型的なものは，プロセッサ数・共有メモリ量・データバス数などであろう．本稿では，並列計算のモデルとして，**組合せ論理回路**（以後，単純に論理回路という）を用いる．これまでに述べてきたプログラムや自然に想定できる並列型のプログラムは，通常，あらゆる長さの入力を処理できるものである．一方，論理回路はある特定の長さの入力だけを処理するものであるため，論理回路に基づく議論を行うとき，各長さごとに用意された論理回路の列を考える．以後，$\alpha=\alpha_1, \alpha_2, \cdots$ を論理回路の可算無限列とする．ここで，各 α_n は n 個の入力ゲートと1個の出力ゲートをもつものとする．また，長さ n の入力 x に対する α_n の出力を $\alpha_n(x)$ と表す．

定義　s を自然数上の関数とする．**論理回路列 α が $s(n)$ サイズ限定** ($s(n)$ size bounded) であるとは，各 α_n のゲート数が高々 $s(n)$ 個であるときをいう．論理回路列 α が問題 A を計算するとは，任意の入力 x に対して，$A(x)$ が真となるときかつそのときに限り $\alpha_n(x)=1$ となるときをいう．**問題 A が $s(n)$ サイ**

ズ限定であるとは，A を計算する $s(n)$ サイズ限定の論理回路列が存在するときをいう．多項式サイズ限定の問題からなるクラスを ***PSIZE*** と表す．

われわれは，アルゴリズム論的な観点から，与えられた問題が並列計算において，実際的に計算可能であることのぎりぎりの限度を，その問題を計算するために必要とされるハードウェア量と計算時間量が，ともに，入力の長さの多項式で抑えられることと考える．この限度内で計算可能な問題のクラスは，各並列計算モデルに応じて変化することが考えられるが，これまでに計算量理論において提案されたすべての並列計算モデルに関して，この限度内で計算可能な問題のクラスが ***PSIZE*** に包含されることが示されている．また，その幾つかは ***PSIZE*** と一致している．そこで，「***PSIZE*** に属すること」を実際的計算可能性の拡張概念の一つと考えることにする．

4. 研究課題と成果

前章で述べた ***BPP*** や ***PSIZE*** は，その定義からも ***P*** とはかなり異なるクラスであることが予想されよう．実際，***PSIZE*** は逐次型のプログラムでは（どのように時間をかけても）原理的に計算不能な問題を含んでいる（したがって，$\boldsymbol{P \neq PSIZE}$ が成り立つが，$\boldsymbol{P \neq BPP}$ となるか否かは知られていない）．このような相違から，たとえ $\boldsymbol{P \neq NP}$ が成立したとしても，そのことがただちに $\boldsymbol{NP \not\subseteq BPP}$ や $\boldsymbol{NP \not\subseteq PSIZE}$ を導くことはなく，むしろその逆が成立する余地が残されているといってよい．われわれの主要な関心は，$\boldsymbol{NP \subseteq BPP}$ や $\boldsymbol{NP \subseteq PSIZE}$ となる余地がどの程度残されているかを明らかにすることであり，より具体的には，次の課題を考察してきている．

課題 1 $\boldsymbol{NP \subseteq BPP \Rightarrow P = NP}$ が成り立つか？

課題 2 $\boldsymbol{NP \subseteq PSIZE \Rightarrow P = NP}$ が成り立つか？

$\boldsymbol{P \subseteq BPP \subseteq PSIZE}$ となることが知られているので，課題 2 が肯定的に解決されれば課題 1 も肯定的に解決される．このことから，これまで課題 2 に多くの精力が注がれている（なお，両方の逆が成立することは明らか）．この課題を直観的にいうと，***NP*** に属するすべての問題が拡張された意味において実際的に計算可能であることともっとも古典的な意味において実際的に計算可能であることが同値であることを示そうとしていることになる．この課題はいまだ解決されていないが，次の少し弱い結果が示されている．

定理[8] $\boldsymbol{NP \subseteq PSIZE \Rightarrow \Sigma_1^p = \Sigma_3^p}$.

2. の終わりで，「$\Sigma_2^p \neq \Sigma_3^p$ となるか否か」は第 2 番目の $\boldsymbol{P = NP}$ 問題として捉え得ることを述べた．この定理は，$\boldsymbol{NP \subseteq PSIZE}$ という仮定が第 2 番目の $\boldsymbol{P = NP}$ 問題に対してやや期待に反する帰結をもたらすという意味において，$\boldsymbol{NP \not\subseteq PSIZE}$ となることの弱い根拠を与えていると考えることができる．上述した課題に関して比較的単純に述べることができる結果は，この定理だけであるといってよい．この研究領域における結果のほとんどは，***PSIZE*** を理論的に特徴付けることから提出された課題に関するものである．以後しばらくの間，***PSIZE*** の理論的特徴付けについて直観的に説明する．

まず初めに多項式時間帰着可能性の概念について説明する．この概念を定義するためには，オラクルチューリング機械と呼ばれる計算モデルを定義する必要があるが，ここではもっと直観的にプログラミング言語を基礎とするモデルを考えることにしよう．まず逐次型プログラミング言語にデータベースの検索機能を追加する．ただし，データベースとしては，現実とはやや異なり，ある（判定）問題 X が与えられる（このようなデータベースのことを計算量理論では oracle と呼ぶ）．X への検索は次の形で実行される．データベース X を使用するプログラムは，計算の途中で X への質問データ z を生成し，データベース X へ質問を行う．データベース X（または，X を管理している DBMS）は，その質問の結果として $X(z)$ を返す．プログラムは，この結果を受けてさらに計算を続行する．計算が終了するまでに，プログラムは何度でもデータベースへの質問を行うことができる．本稿では，与えられたデータベースへ質問を行う機能が，一つの手続きかまたは命令として実現されていると仮定し，かつ，この手続きの 1 回の実行に要する時間を 1 単位時間と仮定する．以上に基づいて，帰着可能性の概念を定義しよう．

定義 問題 A が問題 X に $\leqq_T^p$-**帰着可能**（polynomial time Turing reducible）であるとは，X をデータベースとして使用する多項式時間限定のプログラムによって A が計算できるときをいう．データベース X に $\leqq_T^p$-帰着可能な問題のクラスを $\boldsymbol{P_T(X)}$ と表す．また，データベースの任意のクラス $\boldsymbol{C}$ に対して，$\boldsymbol{P_T(C) = \bigcup_{X \in C} P_T(X)}$ と定める．

$\leqq_T^p$-帰着可能性に関する具体的な結果を一つ紹介しよう．L を 2. で定めた最長路問題とするとき，次が成り立つ．

定理 $NP \subseteq P_T(L)$. すなわち，NP に属するすべての問題は最長路問題に$\leqq_T^P$-帰着可能である．

この定理より，最長路問題は NP の中でもっとも難しい問題の代表であると考えてよい．なぜなら，最長路問題が多項式時間限定の逐次型プログラムによって計算できたならば，$P=NP$ となるからである．このことを別の観点からみると，最長路問題をデータベースとして実現できたならば NP に属するすべての問題を効率よく計算できることになる．しかし，最長路問題をデータベースとして実現した場合，(実現に要する時間を無視したとしても) それを格納するために要する2次記憶容量はおそらく膨大なものとなることが予想される (なお，理論的にはそのデータベースは無限のエントリをもつことになるが，ここでは実用上十分と思われる長さまでのエントリを対象として考えている)．そこで，小規模に実現し得るデータベースの中に最長路問題と同等の能力をもつものが存在するか否かという課題を検討してみよう．このために次の定義を行う．

定義 問題 X が**スパース問題** (**コスパース問題**) であるとは，多項式 p が存在して，任意の $n \geqq 0$ に対して，$X(z)=$真 ($X(z)=$偽) となる長さ n の2進列 z の個数が高々 $p(n)$ 個であるときをいう．スパース問題 (コスパース問題) のクラスを $SPARSE$ ($CoSPARSE$) と表す．

スパース問題 (コスパース問題) は，小規模に実現できるデータベースの代表例と考えてよい．なぜなら，それらを実現しようとするとき，$X(z)=$真 ($X(z)=$偽) となる z のみをエントリとして登録すればよく，かつ，そのエントリ数は比較的小さいからである．そこで，先に述べた検討課題の具体例として，「$NP \subseteq P_T(SPARSE)$ となるか」や「$NP \subseteq P_T(CoSPARSE)$ となるか」などが考えられる．

さて，当初の目的であった $PSIZE$ の特徴付けに話を戻そう．結論を言うと次の命題が知られている．

命題 $PSIZE = P_T(SPARSE) = P_T(CoSPARSE)$.

この命題より，前段落の最後で述べた課題はこれまでの課題と密接に関連していることが分かる．特に，課題2は次のように述べ直すことができる．

課題 2′ $NP \subseteq P_T(SPARSE) \Rightarrow P=NP$ が成り立つか？

課題 2″ $NP \subseteq P_T(CoSPARSE) \Rightarrow P=NP$ が成り立つか？

理論的には，この二つの課題から多くの研究課題が発生し，かつ，その幾つかが解決されてきている．その成果を述べるためには，さらに，帰着可能性のさまざまな概念を定義する必要がある．

定義 問題 A が問題 X に$\leqq_{tt}^P$-**帰着可能** (polynomial time truth-table reducible) であるとは，A が次の形をした多項式時間限定のプログラムによって計算できるときをいう：

Phase 1：入力 x が与えられたとき，まず質問のリスト $z_1, z_2, \cdots, z_m$ を作成する．次に，各 z_i に関してデータベース X からの結果を取得する．

Phase 2：入力 x と上の検索結果を用いて計算を行い，最終的に真または偽を出力する (この phase ではデータベースへの質問は行わない)．

$\leqq_T^P$-帰着可能性と $\leqq_{tt}^P$-帰着可能性の違いは，$\leqq_T^P$-帰着可能性ではデータベースへの各質問の結果に応じてその後の質問内容が変化する (すなわち，データベースごとに質問内容が変化する) 可能性があるのに対して，$\leqq_{tt}^P$-帰着可能性では使用するデータベースとは独立に各入力ごとに生成される質問の内容が一定である，という点である．

定義 $k \geqq 1$ を任意の正整数とする．問題 A が問題 X に$\leqq_{k-T}^P$-**帰着可能**であるとは，データベース X への質問回数が (入力によらず) 高々 k 回であるような多項式時間限定の逐次型プログラムによって A が計算できるときをいう．

定義 データベースを使用するプログラム Q が**単調** (通常，monotone ではなく positive という用語を用いる) であるとは，任意のデータベース X と Y に対して，

$$(\forall z)[X(z) \rightarrow Y(z)] \Rightarrow (\forall x)[Q^X(x) \rightarrow Q^Y(x)]$$

が成り立つときをいう．ここで，$Q^X(x)$ はデータベース X を使用したときの入力 x に関する Q の出力を表す．

この単調性の概念は，比喩的に言うと，論理関数の単調性の概念に類似している．たとえば，$\leqq_{tt}^P$-帰着可能性の定義で示したプログラムの Phase 2 は，(入力 x を固定したとき) 検索結果の列 $X(z_1), \cdots, X(z_m)$ を入力とする論理関数とみなすことができ，かつ，そのプログラムが上の意味で単調ならばこの論理関数も単調となる．上で定義した単調性の概念は，このような考えを一般化したものである．

定義 問題 A が問題 X に $\leqq_{pT}^P$-**帰着可能** (polynomial time positive Turing reducible) であると

は，データベースXを使用する多項式時間限定の単調な逐次型プログラムによってAが計算できるときをいう．AがXに$\leqq^{P}_{ptt}$-**帰着可能** (polynolial time positive truth-table reducible) であるとは，$\leqq^{P}_{tt}$-帰着可能性の定義で示した条件を満たしかつ単調であるプログラムによってAが計算できるときをいう．$\leqq^{P}_{k\text{-}pT}$-帰着可能性の概念も同様に定める．問題Aが問題Xに$\leqq^{P}_{m}$-**帰着可能** (polynomial time many-one reducible) であるとは，Aが次の形をした多項式時間限定の逐次型プログラムによって計算できるときをいう：

Phase 1：入力xが与えられたとき，データベースXへの質問zを（一つ）作成する．

Phase 2：zに関する検索結果（すなわち，$X(z)$）をそのまま出力する．

$\boldsymbol{P_{tt}(X)}$, $\boldsymbol{P_{k-T}(X)}$, $\boldsymbol{P_{pT}(X)}$, $\boldsymbol{P_{tt}(C)}$, $\boldsymbol{P_{k-T}(C)}$, $\boldsymbol{P_{pT}(C)}$…を $\boldsymbol{P_T(X)}$, $\boldsymbol{P_T(C)}$ と同様に定める．

以上の定義に基づいて，$\boldsymbol{P_{tt}(SPARSE)}$, $\boldsymbol{P_{k-T}(SPARSE)}$, …などのクラスを定義することができる．これまで主に考察されてきた課題は，一般に次のように述べられる．

課題 3 $\boldsymbol{NP\subseteq P_*(SPARSE)\Rightarrow P=NP}$ が成り立つか？

課題 4 $\boldsymbol{NP\subseteq P_*(CoSPARSE)\Rightarrow P=NP}$ が成り立つか？

ここで，$\boldsymbol{P_*(SPARSE)}$, $\boldsymbol{P_*(CoSPARSE)}$ はこれまでに定義したクラスの一つを表す．この課題の理論上の意図は，$\boldsymbol{P_m(SPARSE)}$, $\boldsymbol{P_{k-T}(SPARSE)}$…などの制限されたクラスから考察を開始して，徐々に $\boldsymbol{P_T(SPARSE)}$ に接近しようというものである．以下，研究結果の一部を示す．

定理

$\boldsymbol{NP\subseteq P_m(CoSPARSE)\Rightarrow P=NP}$[5].

$\boldsymbol{NP\subseteq P_m(SPARSE)\Rightarrow P=NP}$[12].

$\boldsymbol{NP\subseteq \cup_k P_{k-pT}(SPARSE)\Rightarrow P=NP}$[21].

$\boldsymbol{NP\subseteq P_{1-T}(SPARSE)\Rightarrow RP=NP}$[19].

$\boldsymbol{NP\subseteq P_{1-T}(SPARSE)}$⇒一方向関数は存在しない[19].

$\boldsymbol{NP\subseteq \cup_k P_{k-T}(SPARSE)\Rightarrow P=NP}$[13].

ここで，一方向関数 (one-way function) とは，（直観的にいうと）多項式時間計算可能でかつその逆関数が多項式時間計算不能な1対1関数のことである．これまでに提案された公開鍵暗号系の暗号化関数は，(少なくとも）上の意味での一方向関数となることが期待されている．このように，一方向関数は公開鍵暗号系の安全性に関する理論的研究において導入された概念である．

これまで，***SPARSE***, ***CoSPARSE*** を扱ってきたが，これら以外のクラスについても考察されている．

定義 問題Xが**タリー問題**であるとは，0を含む任意の2進列xに対して，$X(x)$が必ず偽となるときをいう．タリー問題は，1だけからなる有限列全体の上で定義された述語と考えてもよい．タリー問題は，スパース問題の非常に特徴なものであることに注意されたい．タリー問題のクラスを ***TALLY*** と表す．次に，問題Xが **p-選択可能** (p-selective) であるとは，多項式時間で計算可能な2変数関数fが存在して，任意の2進列xとyに対して，次が成り立つときをいう：

(1) $f(x,y)=x$ かまたは $f(x,y)=y$，かつ

(2) $X(x)\vee X(y)\rightarrow X(f(x,y))$.

p-選択可能問題のクラスを ***PSEL*** と表す．

SPARSE のときと同様に，***TALLY*** 及び ***PSEL*** に関して次が示されている．

命題 $\boldsymbol{PSIZE=P_T(TALLY)=P_{tt}(TALLY)}$.

定理[9),14)] $\boldsymbol{PSIZE=P_T(PSEL)}$.

この ***PSIZE*** の特徴付けから，***SPARSE*** の場合と同様の研究課題が考察されている．その結果の一部を次に示す．

定理

$\boldsymbol{NP\subseteq P_m(TALLY)\Rightarrow P=NP}$[3].

$\boldsymbol{NP\subseteq \cup_k P_{k-T}(TALLY)\Rightarrow P=NP}$[18].

$\boldsymbol{NP\subseteq P_{ptt}(PSEL)\Rightarrow P=NP}$[22].

$\boldsymbol{NP\subseteq P_{tt}(PSEL)\Rightarrow RP=NP}$[17].

$\boldsymbol{NP\subseteq P_{tt}(PSEL)}$⇒一方向関数は存在しない[17].

なお，***TALLY*** に関する結果は，今では，Mahaney[12] 及び荻原と渡辺[13]の結果の直接の系となっている．

5. おわりに

本稿では，実際的計算可能性の拡張概念に関する研究の一端を紹介した．この研究の最終目標は，古典的な意味で計算困難と予想される問題がより拡張された意味においても計算困難であることの理論的根拠を示すことである．前章の結果をみても分かるように，最終目標への道はまだ遠く，多くの課題を残している．この研究領域は，残されている課題を徐々に解決しながら進展していくであろう．またさらに，これまでとは異なる実際的計算可能性の形式的概念が提案される

ことも考えられる．そのような提案は，計算量理論に関与する者のみの問題ではなく，計算機科学に関与するすべての研究者が行い得ることであることを述べておきたい．

最後に参考文献について簡単な案内をしよう．この分野を勉強してみようと思われた方は，まず最初に，3)，5)，12)，8)の順に読むことを薦める．ただし，8)で定義された ***P/poly*** というクラスについては，20)を読むことによってより良く理解できるであろう．これらを理解した後，13)，18)，19)，21)を読むとよいであろう．また，本稿で述べた課題と若干異なる課題を扱っているが，7)，11)などは証明手法を勉強する上で有益である．以上の文献は ***SPARSE***，***Co-SPARSE***，***TALLY*** に関するものである．***PSEL*** については，14)，22)，9)，17)をこの順に読むとよい．本稿で述べた確率的計算モデルを最初に形式化したのは 6)である．しかし，この文献は若干読みにくいと思われるので，2)を参照するとよい．本稿で述べた研究の一般論が 10)，16)において展開されている．特に，Schöning[15),16)]が導入した Low 階層・High 階層と呼ばれる概念は，計算量理論において標準的なものとなりつつあり，かつ，本稿の研究課題をより一般的に議論するための枠組みを与えている．最後に，本稿で述べた研究課題を最初に意識させたのは，Berman と Hartmanis[4)]であることを記しておきたい．

参 考 文 献

1) Allender, E.: The Complexity of Sparse Sets in P, Proc. 1st SICT Conference, LNCS 223, Springer-Verlag, pp. 1-11 (1986).
2) Balcázar, J. L., Diaz, J. and Gabarró, J.: Structural Complexity I, EATCS Monograph on Theoretical Computer Sci., Vol. 11, Springer-Verlag (1988).
3) Berman, P.: Relationship between Density and Deterministic Complexity of NP-Complete Languages, Proc. 5th ICALP, LNCS 62, Springer-Verlag, pp. 63-71 (1978).
4) Berman, L. and Hartmanis, J.: On Isomorphism and Densities of NP and Other Complete Sets, SIAM J. Comput., Vol. 6, pp. 305-322 (1977).
5) Fortune, S.: A Note on Sparse Complete Sets, SIAM J. Comput., Vol. 8, pp. 431-433 (1979).
6) Gill, J.: Computational Complexity of Probabilistic Turing Machines, SIAM J. Comput., Vol. 6, pp. 675-695 (1977).
7) Kadin, J.: $P^{NP[\log]}$ and Sparse Turing-Complete Sets for NP, Proc. 2nd SICT Conference, IEEE, pp. 33-40 (1987).
8) Karp, R. and Lipton, R.: Some Connections between Nonuniform and Uniform Complexity Classes, Proc. 12th ACM Symp. on Theory of Comput., pp. 302-309 (1980).
9) Ko, K.: On Self-Reducibility and Weak p-Selectivity, J. Comput. Sys. Sci., Vol. 26, pp. 209-221 (1983).
10) Ko, K. and Schöning, U.: On Circuit-Size Complexity and the Low Hierarchy in NP, SIAM J. Comput., Vol. 14, pp. 41-51 (1985).
11) Long, T. J: A Note on Sparse Oracles for NP, J. Comput. Sys. Sci., Vol. 24, pp. 224-232 (1982).
12) Mahaney, S.: Sparse Complete Sets for NP: Solution of a Conjecture of Berman and Hartmanis, J. Comput. Sys. Sci., Vol. 25, pp. 130-143 (1982).
13) 荻原，渡辺: On Polynomial Bounded Truth-Table Reducibility of NP Sets to Sparse Sets, 研究報告書，東工大情報科学科 (1989年10月).
14) Selman, A. L.: P-Selective Sets, Tally Languages, and the Behavior of Polynomial Reducibilities on NP, Math. Sys. Theory, Vol. 35, pp. 55-65 (1979).
15) Schöning, U.: A Low and High Hierarchy within NP, J. Comput. Sys. Sci, Vol. 27, pp. 14-28 (1983).
16) Schöning, U.: Complexity and Structure, LNCS 211, p. 200, Springer-Verlag (1985).
17) Toda, S.: On Polynomial-Time Truth-Table Reducibilities of Interactable Sets to p-Selective Sets, Math. Sys. Theory, to appear (1990).
18) Ukkonen, E.: Tow Results on Polynomial Time Turing Reductions to Sparse Sets, SIAM J. Comput., Vol. 12, pp. 580-587 (1983).
19) Watanabe, O.: On $\leq^{P}_{1-tt}$-Sparseness and Nondeterministic Complexity Classes, Proc. 15th ICALP, LNCS 317, Springer-Verlag, pp. 687-709 (1988).
20) Yap, C.: Some Consequences of Non-Uniform Conditions on Uniform Classes, Theoret. Comput. Sci., Vol. 27, pp. 283-300 (1983).
21) Yesha, Y.: On Certain Polynomial-Time Truth-Table Reductions to Sparse Sets, SIAM J. Comput., Vol. 12, pp. 411-425 (1983).
22) Selman, A. L.: Reductions on NP and p-Selective Sets, Theoret. Comput. Sci., Vol. 19, pp. 287-304 (1982).

(平成元年10月26日受付)