# 固定パラメータ容易性

岡本　吉央

1990年代，組合せ最適化分野では解くことが難しいとされる問題（いわゆるNP困難な問題）に対してどのようにアプローチをしたらよいか研究者が盛んに議論し，新たな潮流を生んでいった．顕著なものの1つが理論的保証のある近似アルゴリズムに関する研究であり，半正定値計画法に基づくアルゴリズム設計技法や計算量理論分野で進展したPCP定理および一意ゲーム予想に基づく近似不可能性の結果に代表される．本特集にも関連する記事が掲載されているので御覧いただきたいが，その流れの1つの帰結は「NP困難問題の中にも近似可能性/不可能性が紡ぎ出す階層がある」ということである．

一方，Downey, Fellowsらは厳密アルゴリズムの世界にそのような階層があることを指摘し出した．その理論は現在「パラメータ化計算量理論（parameterized complexity theory）」という名のもとに研究されている分野であるが，そこでの重要な概念が**固定パラメータ容易性**（fixed-parameter tractability）と**固定パラメータ困難性**（fixed-parameter intractability）である．本稿ではアルゴリズム設計技法である前者に絞って図を交えて基本を解説したい．

## 1.　基本的な枠組と例とする問題

無向グラフ$G$において，その頂点部分集合$U$が$G$の頂点被覆であるとは，$G$の任意の辺の端点の少なくとも一方が$U$に属することである．無向グラフ$G$と自然数$k$が与えられたとき，$G$に要素数$k$以下の頂点被覆が存在するかどうか判定する問題が頂点被覆問題であり，代表的なNP完全問題として知られている．単純なアルゴリズムとして次のようなものが考えられる．与えられたグラフの頂点数は$n$であるとする．このとき，その$n$個の中から$k$個を選んできて，それらが頂点被覆になるか調べる．これをすべての選び方に対して試すというものである．このアルゴリズムにおける「すべての選び方」の総数はオーダー$n^k$であり，$k=10$なら$n^{10}$，$k=10000$なら$n^{10000}$となり，とても使いものにならない．

しかし，以下に示すアルゴリズムではどんなに$k$が大きくなっても$n$の肩に乗る指数が変わらず，$k$に依存して変わるのはオーダー記法に隠された定数部分のみである．この依存性が$k$に関する多項式であれば，全体として多項式時間アルゴリズムとなってしまうが，NP完全問題に対してそれは望めず，実際，依存性は$k$に関して超多項式（例えば指数関数）となる．このように，$k$が大きくなっても$n$の肩に乗る指数が変わらず，定数部分のみが$k$に依存するようなアルゴリズムを**固定パラメータ・アルゴリズム**（fixed-parameter algorithm）と呼び，これが本稿の主眼である．このとき$k$を問題のパラメータと呼ぶ．例えば，$O(10^k n^3)$といった計算量をもつアルゴリズムは$k$をパラメータとする固定パラメータ・アルゴリズムである．固定パラメータ・アルゴリズムをもつ問題のことを**固定パラメータ容易**（fixed-parameter tractable）と呼び，そのような問題全体のクラスをFPTと呼んでいる．この話題の本が今までに3冊出版されているので[2]～[4]，詳細はそちらを参照いただきたい．アルゴリズム設計技法については特にNiedermeier[4]を推薦する．関連する「高速指数時間アルゴリズム設計」に関しては著者による記事[5]を参照いただきたい．

## 2.　技法Ｉ：探索木の縮小

問題解決に対する一般的な方法に「場合分け」がある．それを組織立って行うと「木の探索」になる．ここでは，分枝規則（場合分けの方法）の工夫によって，探索木が小さくなるようにすることを考える．そこから，高速なアルゴリズムが得られるのである．

まず次のような単純なアルゴリズムを考える．頂点被覆は任意の辺のどちらか一方の端点を必ず含まなければならない．そのため，任意の辺のどちらを頂点被

おかもと　よしお
豊橋技術科学大学 情報工学系
〒441-8580 豊橋市天伯町雲雀ヶ丘1-1

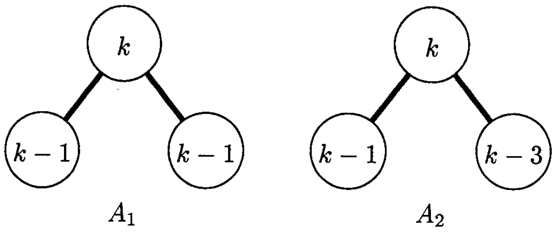


図1　アルゴリズム $A_1$ と $A_2$ での分枝．丸の中にパラメータ値が書かれていて，$A_2$ では右部分木のパラメータ値が大幅に減少している

覆に含めるのかによる場合分けを行う．具体的に次のようになる．与えられたグラフ $G=(V, E)$ に大きさ $k$ の頂点被覆 $C$ があるかどうか判定するアルゴリズム $A_1(G, k)$ では，まず任意の辺 $e=\{u, v\}\in E$ を選ぶ．いま，$u$ を $C$ に含めれば，$C$ に含めるべき残りの $k-1$ 頂点は $G-u$ に所属する．同様に，$v$ を $C$ に含めれば $C$ に含めるべき残りの $k-1$ 頂点は $G-v$ に所属する．つまり，$A_1(G, k)$ の出力は $A_1(G-u, k-1)$ と $A_1(G-v, k-1)$ の出力のいずれかが Yes ならば Yes であり，そうでなければ No である．このようにして再帰的なアルゴリズムが設計でき，探索木のノード数がおよそ $2^k$ となるので，これは固定パラメータアルゴリズムである．図1左が分枝の様子を示している．

このアプローチでは，探索木のノード数がそのまま計算量に効いてくるので，それを減らすような工夫を施せば計算量が改善できる．そのために，頂点 $v\in V$ を考えてみよう．この頂点は $C$ に含まれるか含まれないかのいずれかであるので，それに応じて場合分けをする．この頂点が含まれる場合，先程と同様に $G-v$ の中から残り $k-1$ 個の頂点を見つければよい．では，$v$ が含まれない場合，$v$ に接続する辺をすべて被覆するためには $v$ に隣接する頂点をすべて $C$ に含めなければならない．よって，$v$ に隣接する頂点の集合を $N(v)$ と書くとき，$G-(N(v)\cup\{v\})$ の中から残り $k-d(v)$ 個の頂点を見つければよい（ただし，$d(v)$ は $v$ の次数と呼ばれ，$d(v)=|N(v)|$ として定義される）．この方針に従ってアルゴリズム $A_2(G, k)$ を考えると「$A_2(G, k)$ の出力は $A_2(G-v, k-1)$ と $A_2(G-(N(v)\cup\{v\}), k-d(v))$ の出力のいずれかが Yes ならば Yes であり，そうでなければ No である」という再帰的なアルゴリズムが設計できる．この場合，$d(v)$ ができる限り大きくなるように $v$ を選ぶと探索木のサイズが小さくなるが，普通 $d(v)$ の大きさ（小ささ）に制限はない．しかし，最大次数が2以下であるグラフはすべての連結成分がパスか閉路になっているので，そこでは最小頂点被覆を簡単に見つけることができる．つまり，「最大次数が2以下」となったら，再帰呼び出しを止めるのである．こうすることで，再帰呼び出しをする間は $d(v)\geq 3$ となる頂点 $v$ の存在が保証され，そのような頂点を選ぶことで，探索木のノード数がおよそ $1.4656^k$ となることが示される．図1右が分枝の様子を示している．

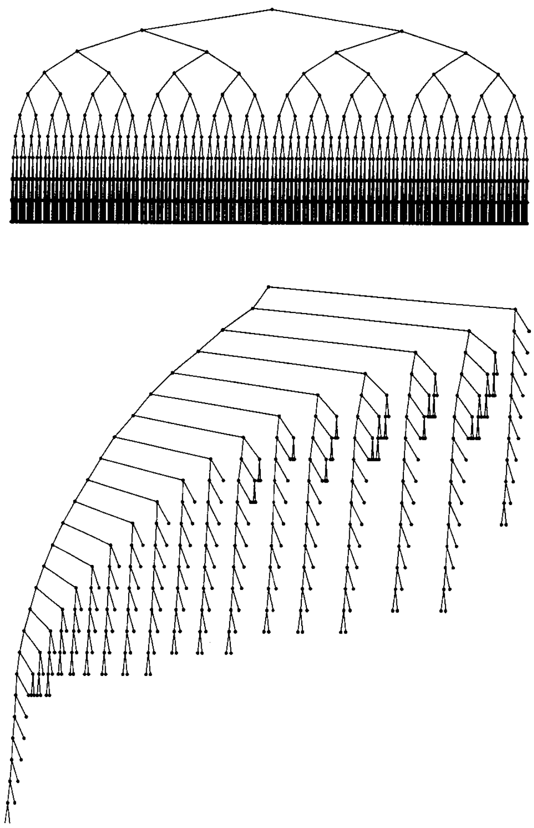

図2　2つの異なるアルゴリズムの探索木．上が $A_1$ に対する探索木．下が $A_2$ に対する探索木．共に頂点数200のグラフにアルゴリズムを適用したもので，上は $k=10$，下は $k=25$ として実行した例

では，実例を見てみよう．図2では，2つのアルゴリズム $A_1$ と $A_2$ を頂点数200のあるグラフ上で実行したときの探索木である．アルゴリズムの記述から想像される通り $A_1$ ではバランスした木が，$A_2$ ではアンバランスな木が得られる．このアンバランスさによって，右部分木の深さが再帰的に小さくなり，全体として高速化されている様子がよく分かる．

## 3. 技法II：カーネル化

問題解決に対する一般的な方法として他のものに「前処理」がある．固定パラメータ・アルゴリズムの

分野では，多項式時間の前処理によって得られる構造の大きさがパラメータにしか依存しない場合，その得られた構造を問題のカーネル（kernel）と呼んでいる．いったんカーネルが得られれば，その上でしらみ潰し探索を行っても，計算量の指数的な部分はパラメータにしか依存しなくなり，全体で固定パラメータ・アルゴリズムとなるのである．

では，次のような前処理を考えてみる．与えられたグラフ $G=(V,E)$ に大きさ $k$ の頂点被覆 $C$ があるかどうか判定したい．もし次数が $k$ よりも大きな頂点 $v$ があれば，それは $C$ に含まれなければならない．なぜなら，そうでないとすると，$v$ に接続する辺を被覆するために $v$ に隣接する頂点をすべて $C$ に含めなければならないが，そうすると $C$ の頂点数が $k$ を超えてしまう．このようにして，次数が $k$ よりも大きい頂点を随時 $C$ に含めていき，それらを除去していこう．これを繰り返すことで得られたグラフを $G'=(V',E')$ としよう．このグラフにおける任意の頂点の次数は $k$ 以下である．よって，$k$ 個頂点を集めてきても高々 $k^2$ 個の辺しか被覆できない．したがって，$|E'|>k^2$ ならば答えは No である．そうでないときには $|E'|\leq k^2$ であり，つまり，グラフ $G'$ の大きさはパラメータ $k$ にしか依存しない．これが求めていたカーネルであり，その大きさは $O(k^2)$ である．

実例を見てみよう．図 3 の上側のグラフに対してこのカーネル化を施した例が図 3 の下側のグラフである．かなり疎なグラフが得られていることが分かる．

得られるカーネルの大きさが小さければ小さいほど，アルゴリズムの効率がよくなる．上で与えたカーネルの大きさは $O(k^2)$ であるが，実は大きさが $2k$ のカーネルも知られている．これは頂点被覆問題の線形計画緩和から得られるものであり，大変興味深い．

ここまで議論した 2 つの手法を組み合わせて使うこともちろんできる．それ以外にも様々な技法が存在するので，詳細は参考文献に譲る．ちなみに，頂点被覆問題に対する現在最速の固定パラメータ・アルゴリズムは Chen, Kanj and Xia [1] によるもので，その計算量は $O(1.2738^k+kn)$ である．固定パラメータ・アルゴリズムの世界は日進月歩であり，今後どのような進化を遂げるのか想像がつかない．本稿では参考文献に登場しない「実物」を図として示してみた．理解の助けになれば幸いである．

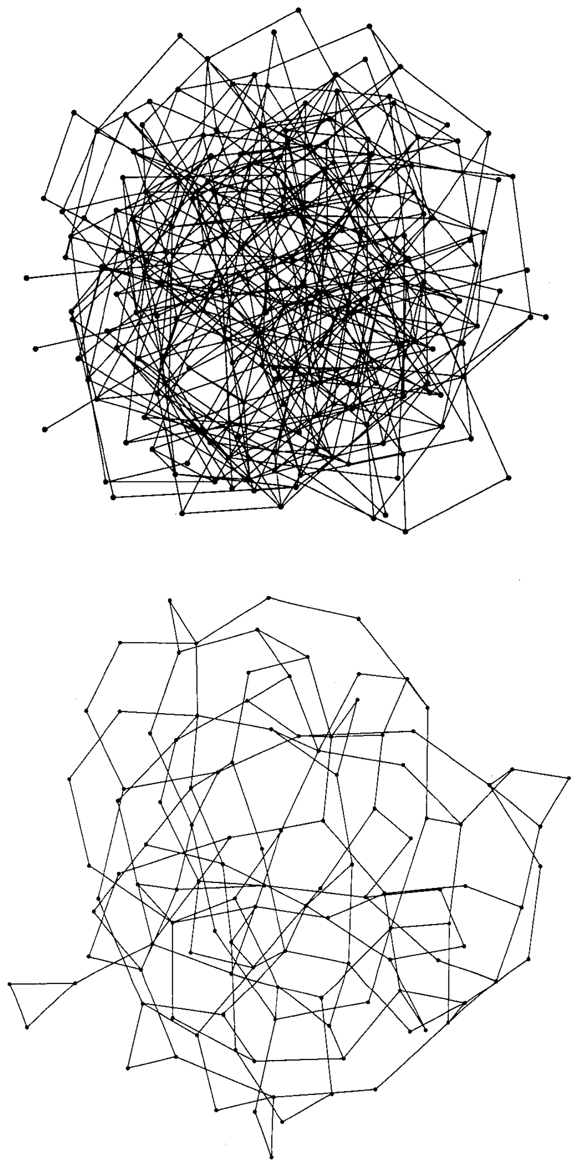

**図 3**　上側のグラフにカーネル化を適用した例．結果が下側のグラフである

## 参考文献


[1] J. Chen, I. A. Kanj and G. Xia: Improved parameterized upper bounds for vertex cover. Proc. 31st MFCS (2006) 238-249.

[2] R. Downey and M. Fellows: Parameterized Complexity. Springer, 1999.

[3] J. Flum and M. Grohe: Parameterized Complexity Theory. Springer, 2006.

[4] R. Niedermeier: Invitation to Fixed-Parameter Algorithms. Oxford University Press, 2006.

[5] 岡本吉央：離散最適化問題に対する高速な厳密アルゴリズム．オペレーションズ・リサーチ 50, 2005, pp. 763-769.