



D. E. Knuth, J. H. Morris, V. R. Pratt: Fast Pattern Matching in Strings

SIAM Journal on Computing, Vol.6, No.2, pp.323-350 (1977)

テキストの中から与えられたパターンを見つけ出すという、いわゆる文字列探索のアルゴリズムを調べようとアルゴリズムの教科書をひもとくと、必ずといってよいほど、力まかせの方法、Knuth, Morris, Prattによる方法、BoyerとMooreによる方法¹⁾の3つが紹介されている。本論文は2番目の方法—著者の頭文字をとってKMPアルゴリズムと省略される—について述べている。

文字列探索が必要とされる代表的な例に、テキストエディタの探索コマンドがある。実際のテキストエディタの探索コマンドでは、探索する文字列がきわめて短い場合には力まかせの方法が用いられることがある以外は、第3の方法—これも著者の頭文字からBMアルゴリズムと省略される—が利用される場合が多いようではあるが、KMPアルゴリズムはBMアルゴリズムと並び、アイデア1つで計算量が改善されるというアルゴリズムの面白さを、我々に十分教えてくれる。

本論文は、KMPアルゴリズムの動作を例を用いて解説することから始まり、具体的なアルゴリズムの提示、アルゴリズムの改良と拡張、計算量に関する理論的な考察と続く。アルゴリズムを提案する論文としてはここまでで十分なのだが、本論文はさらに、長さが偶数の回文文字列の話、KMPアルゴリズム発明の歴史、ついには後書きと称してBMアルゴリズムに関する考察にまで話が及ぶという、特異とも思える構成となっている。

ここで述べられている歴史によると、KMPアルゴリズムは、元々はMorris、および、Knuth, Prattチームにより独立に発明された。Morrisは、1969年にCDC 6400という計算機上のテキストエディタを作成していた際、有限オートマトン理論を利用して、このアルゴリズムの原形を考えた。またこれとは別に、Knuthは1970年に、長さが偶数の回文文字列の2方向プッシュダウンオートマトンによる認識の理論の調査をしているうちに、その背後の文字列探索アルゴリズムの存在に気づき、本アルゴリズムの発明に至ったという。これは、Knuthにとっ

て、オートマトン理論が現実のプログラミングの問題の解決に役に立った初めての経験であったらしい。Robert Sedgewickは、その有名なアルゴリズムの教科書²⁾の文字列探索の章の中で、このことを「理論的な結果が思いがけず直接実用と結びついた稀な実例である」と指摘しているが、実際この経験は、Knuthにとってきわめて印象深いものであったに違いない。そう考えると、上で述べたような本論文の特異とも思える構成も、著者による思いの発露と捉えて興味深い。ちなみに、本論文が掲載されているSIAM Journal on Computingの同じ号には、Sedgewickの論文も掲載されている。

Morrisがこのアルゴリズムを思いついた背景には、CDC 6400でのバッファリング処理が複雑であるため、テキストの「後戻り」を必要としない方法が求められていたことがあった。最近のように主記憶をふんだんに使うことができなかったという不自由さが、新しいアルゴリズムの発明に結びついたといえよう。

本論文のKMPアルゴリズムの改良の章には、番兵を用いた高速化の話も述べられている。力ずくに記述されたプログラムが多い現在では、番兵は以前ほどは使われなくなってしまったが、本論文が発表された1970年代後半では、番兵は美しく高速なプログラムを記述するための重宝な技術であった。このようなところにも、時代の流れを感じさせるものがある。

KMPアルゴリズムは、素朴なアルゴリズムから賢いアルゴリズムをプログラム変換によって導出する例題としてもよく用いられる。このような題材としてもKMPアルゴリズムが役に立つことになるとは、著者たちは想定していなかったであろう。

参考文献

- 1) Boyer, R. S. and Moore, J. S.: A Fast String Searching Algorithm, Commun. ACM, Vol.20, No.10, pp.762-772 (1977).
- 2) Sedgewick, R.: Algorithm in C++, Addison Wesley (1992).

(平成16年6月18日受付)

岩崎英哉 / 電気通信大学情報工学科
iwasaki@cs.uec.ac.jp