1Q−02

BM法を用いたハードウェア方式による文字列検索アルゴリズムの提案

大曽根　匡
専修大学 経営学部 情報管理学科

1．はじめに

ハードウェアを用いた文字列検索アルゴリズムとしては，素朴なアルゴリズムを並列化したセルラ・アレイ方式や連想メモリ法，そして，有限オートマトン法をハードウェア化した方式などが提案されている．これらのハードウェア方式は，いずれも１文字ずつ文字を検索エンジンに入力していくので，テキスト長をnとすると，計算量は$o(n)$となる．一方，高速なソフトウェア・アルゴリズムとして知られているBM法の計算量は，パターン長をmとすると，$o(n/m)$である．しかし，この高速なBM法を利用したハードウェア方式による文字列検索アルゴリズムは，筆者の知る限りでは提案されていない．そこで，本論文では，ハードウェアを用いてBM法を並列化した方式のアルゴリズムを提案する．

2．提案アルゴリズム

本論文では説明を簡単にするためにパターン長を４文字に限定する．しかし，この限定は本質的ではなく，少しの工夫で任意の長さのパターンに適用することができる．

アルゴリズムの発想の基本は，ハードウェアを用いてBM法を並列化することである．そのアルゴリズムの概要を下記に示す．

（1）*RAM* からテキストを図 2.1(a)のように４文字飛びに並列に４文字分入力する．この入力の方法を*縦読み*と呼ぶことにする．

（2）縦読みされた４文字の情報を使って，BM法の考え方に基づいて，その付近に指定されたパターンの存在する可能性があるかどうかを判断する．もし可能性があれば，可能性のある位置から図 2.1(b)のようにテキストを４文字分*横読み*し，パターンと一致するかどうかを判定する．

（3）もし，可能性のない場合は，前に縦読みした位置より 16 文字先から再び４文字分縦読みを行う（図 2.1(c)参照）．

この手続きをテキストが終了するまで繰り返す．

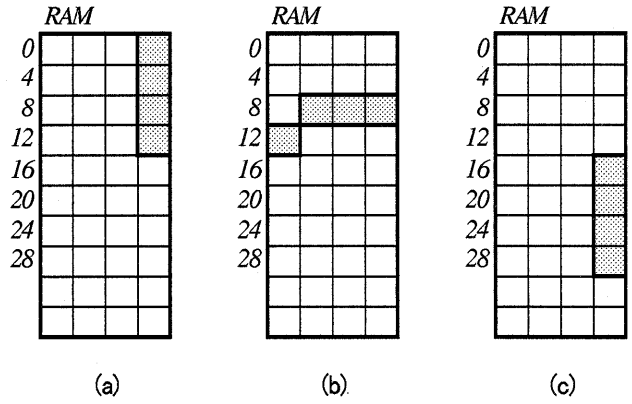

図 2.1　アルゴリズムの概要

以上が本論文で提案する並列化したBM法の概要である．

このアルゴリズムでは，*VH-RAM* という縦読み／横読みのを可能とする特別な機構をもった *RAM* を導入する．しかし，前処理を行うことにより，通常の *RAM* を用いて実現することも可能である．

A Parallel String Seraching Algorithm Using BM Method.
Tadashi Osone
Department of Information Management,
Senshu University

3．ハードウェア構成と動作例

前節で示したアルゴリズムを実現するためのハードウェア構成を図3.1に示す．

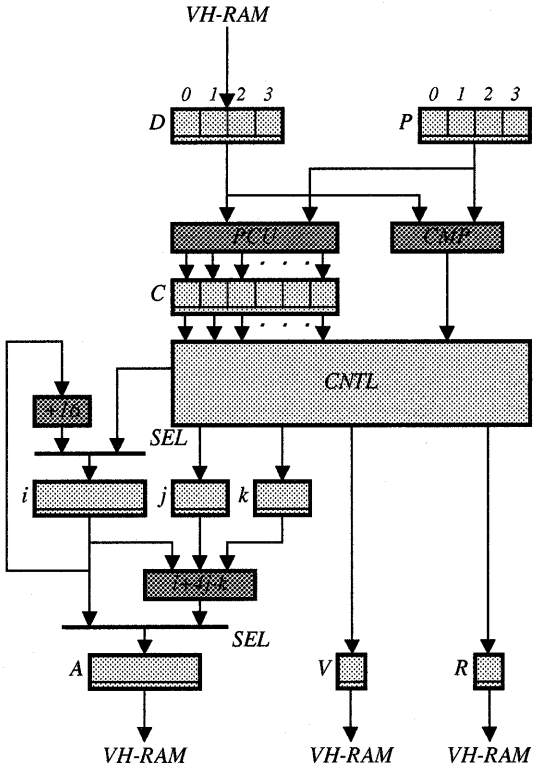

図 3.1 ハードウェア構成

動作例として，パターンが「ＡＢＣＤ」，テキストが図 4.1 に示す４８文字のテキストの場合について考える．この例の場合，あらかじめパターンレジスタ P にパターン「ＡＢＣＤ」をセットしておく．また，初期縦読みアドレスを３にセットする．

（１）時刻１：縦読みアドレス３から最初の縦読みデータ「ＥＦＧＨ」をフェッチし，データレジスタ D に取り込む．そして，並列比較ユニット PCU を用いて P と D の各文字の比較を行う．この場合，両者の文字同士で一致する文字はないので，すべての比較結果ラッチ $C[j,k]$ は０となり，横読みを行わないことを判断する．そして，縦読みアドレスを＋１６更新し１９とする．

（２）時刻２：縦読みアドレス１９から２回目の縦読みを行い，データ「ＥＢＦＢ」を D に取り込む．このとき，D の１文字目Ｂと P の１文字目Ｂが一致するのでラッチ $C[1,1]$ が１となり，また，D の３文字目Ｂと P の１文字目Ｂが一致するので $C[3,1]$ も１となる．そして，横読みアドレスを２２と３０を算出する．

（３）時刻３：アドレス２２からテキスト「ＡＢＣＤ」を横読みし，パターンを検出する．

（４）時刻４：アドレス３０からテキスト「ＡＢＡＢ」を横読みし，パターンの未検出を確認する．これで横読みがすべて終わったので，縦読みアドレスを３５に更新し，縦読みを再開する．

T

0	A	B	C	E
4	F	A	B	F
8	C	D	A	G
12	B	C	D	H
16	C	A	B	E
20	A	B	A	B
24	C	D	A	F
28	A	B	A	B
32	A	B	C	A
36	B	C	D	C
40	E	A	B	C
44	D	B	D	E

図 4.1 テキストの例

（５）時刻５：縦読みアドレス３５からデータ「ＡＣＣＥ」をデータレジスタ D に読み込む．このとき，ラッチ $C[0,0]$ と $C[1,2]$，$C[2,2]$ が１となる．そして，それらに対応する横読みアドレス３５，３７，４１を算出する．

以下同様にして，最終的には，縦読み３回，横読み５回の計８回のデータ読み出しで４８文字のテキストの検索を終了することができる．

このアルゴリズムを用いると，最良の場合，１回のテキストの縦読みで 16 文字分のテキストの検索をすることができため，ソフトウェアによるＢＭ法における最良の場合よりも４倍高速であり，従来のハードウェア方式に比べ１６倍高速となる．

参考文献

【1】Knuth, D.E. Morris, J.H. and Pratt, V.R., "Fast Pattern Matching in Strings," ***SIAM J. Comput.***, Vol.6, No.2, pp.323-350 (1977).

【2】Boyer, R.S. and Moore, J.S., "A Fast String Searching Algorithm," ***Comm. ACM***, Vol.20, No.10, pp.762-772 (1977).