2T−6　入力文字情報を最大限に活用した複数文字列に対する文字列検索アルゴリズム

大曽根　匡
専修大学　経営学部　情報管理学科

1．はじめに

高速な文字列検索アルゴリズムとして、ＫＭＰ法やＡＣ法、ＢＭ法などがよく知られているが、特に、ＢＭ法は、パターンの末尾の文字から前方へ照合を進めることにより高速化を図った著名なアルゴリズムである。しかし、ＢＭ法では、パターンを右へシフトした際、それまでに入力した文字の情報を全て忘れてしまうようになっている。ＢＭ法を、複数パターンの同時検索に拡張したアルゴリズム（拡張ＢＭ法）も同様である。そこで、前回、単数パターンに対して、入力文字情報を最大限に活用したアルゴリズムを提案した[1]。今回はこれを拡張し、複数パターンの同時検索を可能とするアルゴリズムについて報告する。

2．提案アルゴリズム

提案するアルゴリズムでは、パターンを右へシフトした際に、それまで入力した文字情報を覚えておくようにした。例として、「ABCD」,「EFG」,「HI」の３つのパターンの同時検索を考えよう。まず、各パターンの最後尾の文字をテキストの２文字目に合わせておく。このときの照合状況を「**_?」としておく。ここで、＊は既照合文字、＿は未照合文字、？は次の照合文字を表している。そこで、「?」に対応する文字をテキストから入力する（テキストの２文字目）。それが「B」であれば、全パターンを２文字分右にシフトし、次の照合状況を「_B_?」とする。このようにして、パターンをシフトした後も、入力した文字「B」を覚えておくのである。

提案アルゴリズムでは、パターンから考えられる全ての照合状況をあらかじめ生成しておく。前述の例の場合の全ての照合状況を表１に示す。次に、[1]のアルゴリズムと同様に、状態ｉのときに文字ｃが入力されたら、次の状態が何になるかという情報が書かれた状態遷移テーブル T[i,c] と、次に何文字先の文字をテキストから入力すればよいかという情報の書かれたスキップテーブル S[i,c] を作成する。表２と３に、前述の例の場合の状態遷移テーブルとスキップテーブルを示す。そして、パターンの検索は、この２つのテーブルの参照を繰り返すことにより行う。その動作例を図１に示す。初期状態は０とし、テキストの２文字目から文字を入力し始める。この例の場合、提案法では、25 文字のテキストに対し 15 文字の入力で検索を終了している。すなわち、平均スキップ幅は 1.7 文字である。

3．性能実験

アルゴリズムの性能は、平均スキップ幅によって表現できる。そこで、提案法と拡張ＢＭ法、ＡＣ法の平均スキップ幅を定量的に比較するために、性能実験を行った。その実験結果の一部を以下に示す。、実験１では「101010」,「010101」,「110011」、実験２では「101010」,「000111」,「111110」のパターンを用いた。また、アルファベットの文字種は「1」と「0」の２文字とした。図２と図３は、テキスト上の「1」の出現確率に対する平均スキップ幅の変化を示している。これより、「1」の出現確率の減少に伴い、提案法の性能と拡張ＢＭ法やＡＣ法の性能との差がだんだん大きくなっていくことがわかる。

参考文献

【1】大曽根 他,”入力文字情報を最大限に活用した文字列検索アルゴリズムの提案,” 情報処理学会第46回全国大会論文集 (1987)

A String Searching Algorithm for Multiple Patterns
Tadashi OHSONE
Senshu University
2-1-1 Higashimita, Tama-ku, Kawasaki 214,Japan

表１．状態の定義

状態	照合状況
-3	**HI_ ?
-2	*EFG_ ?
-1	ABCD_ ?
0	**_ ?
1	*A_ ?
2	_ B_ ?
3	*E_ ?
4	*_ F?
5	**H?
6	**?I
7	A_ C?
8	_ B?D
9	*E?G
10	*?FG
11	A?CD
12	?BCD

？：次の照合位置
＊：既照合文字

表２．状態遷移テーブル

	A	B	C	D	E	F	G	H	I	#
-3	1	2			3	4		5	6	
-2	1	2			3	4		5	6	
-1	1	2			3	4		5	6	
0	1	2			3	4		5	6	
1	1	2	7		3	4		5	6	
2	1	2		8	3	4		5	6	
3	1	2			3	4	9	5	6	
4	1				3		10	5		
5	1				3			5	-3	
6								-3		
7	1			11	3			5		
8			12							
9						-2				
10					-2					
11		-1								
12	-1									

空白は状態０

表３．スキップテーブル

	A	B	C	D	E	F	G	H	I	#
-3	2	2	2	2	2	1	2	1	-1	2
-2	2	2	2	2	2	1	2	1	-1	2
-1	2	2	2	2	2	1	2	1	-1	2
0	2	2	2	2	2	1	2	1	-1	2
1	2	2	1	2	2	1	2	1	-1	2
2	2	2	2	-1	2	1	2	1	-1	2
3	2	2	2	2	2	1	-1	1	-1	2
4	2	2	2	2	2	2	-2	1	2	2
5	2	2	2	2	2	2	2	1	2	2
6	3	3	3	3	3	3	3	3	3	3
7	2	2	2	-2	2	2	2	1	2	2
8	3	3	-2	3	3	3	3	3	3	3
9	3	3	3	3	3	3	3	3	3	3
10	4	4	4	4	4	4	4	4	4	4
11	4	4	4	4	4	4	4	4	4	4
12	5	5	5	5	5	5	5	5	5	5

＃：その他の文字

入力順	①	②	③	④	⑦	⑤	⑥	⑧	⑨	⑩	⑪	⑫	⑭	⑬	⑮
アドレス	2	4	6	8	9	10	11	13	15	16	18	20	21	22	24
	A	C	D	A	E	C	D	G	H	A	D	G	G	I	C
次状態	2	0	0	1	0	7	11	0	5	1	0	0	0	6	0
スキップ幅	2	2	2	2	4	1	-2	2	1	2	2	2	3	-1	2

図１．提案法の動作例

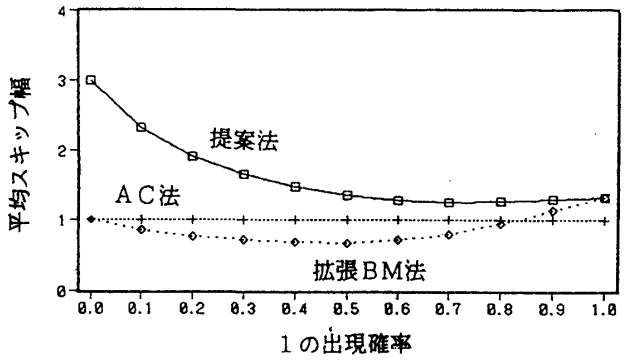

図２．実験１に対する平均スキップ幅

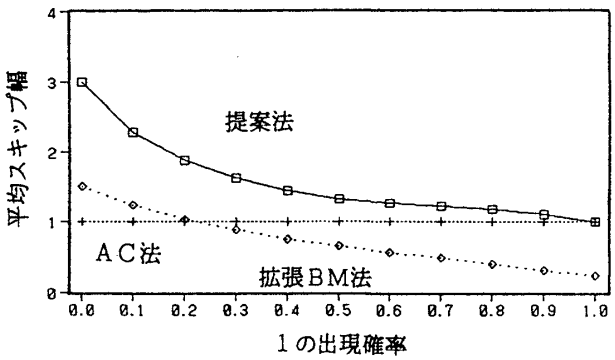

図３．実験２に対する平均スキップ幅