D-024

短縮した遷移パターンによるダブル配列構築法の提案
Proposal to Construction Method of Double Array with Short Transition Pattern

瀬社家 奨 † Sho Seshake
望月 久稔 † Hisatoshi Mochizuki

1. はじめに

高速な検索手法であるトライの実装方法としてダブル配列がある [1]．ダブル配列は，トライ木における節点から遷移可能な遷移種の集合，即ち遷移パターンを組み合わせて構築するため，トライの検索性能をコンパクトに実現する．ダブル配列は遷移パターンを組み合わせる位置，即ち基底値を管理する配列 BASE と，遷移元の節点を確認する配列 CHECK からなる．節点 r から遷移種 a による節点 t への遷移が存在する時，$BASE[r]+a=t, CHECK[t]=r$ を満たす．

基底値を算出する処理を XCheck と呼ぶ [1]．XCheck は，遷移パターンとダブル配列それぞれを構成する要素を比較し，パターンを構成する遷移種の遷移先すべてが未使用要素である基底値を算出する．XCheck はダブル配列上の全要素を比較対象要素とするため，ダブル配列の構築において膨大な時間計算量を要する．XCheck を高速化する先行研究として未使用要素をリスト構造で管理する手法があり，森田ら [2] は単方向，大野ら [3]，矢田ら [4]，中村ら [5] は双方向のリスト構造を用いて，比較対象要素数を抑制している．

本稿は，XCheck の比較対象である未使用要素の分類と，比較する遷移パターンにおける遷移種のコードの最大と最小の幅を狭めることにより，XCheck の比較回数削減を図る．

2. 未使用要素の分類と遷移パターン長の短縮

ダブル配列の構築を高速化するためには，XCheck の時間計算量を抑制する必要がある．本章では XCheck の比較回数を削減するために，XCheck において比較対象である未使用要素を周辺要素の使用状況に応じて分類する手法を説明する．次に，XCheck で比較する遷移パターンにおいて遷移種のコードの最大と最小の幅を遷移のパターン長と定義し，遷移種を回転することでパターン長を短縮する手法を説明する．

まず，比較対象要素の候補を限定するために，未使用要素を周辺要素の使用状況に応じて分類する [6]．未使用要素 r の未使用状況を，r の右方向に存在する複数個の要素の使用状況から求め，r をその未使用状況に対応するリスト構造に追加する．XCheck 時は遷移パターンが必要とする未使用状況のリスト構造を比較対象要素の候補とする．

次に，パターン長の短縮法を説明する．遷移パターンを構成する遷移種のコードのうち，最大のコードを max，最小のコードを min とし，パターン長を式 (1) に定義する．

$$\text{パターン長} = max - min + 1 \qquad (1)$$

遷移パターンの遷移種はコード上で連続するとは限らないため，遷移パターン中に隙間が生じ得る．本手法は，遷移種のコードをリングバッファ状に循環し，遷移種を回転することで遷移パターン内の隙間を端に移動し，パターン長を短縮する．コードの循環は短縮前のパターン長の範囲で行う．

例として遷移パターン $PAT=\{a,c,d\}$ の短縮を図 1 に示す．図中の a,c,d は遷移種，数値はコードであり，二重円は PAT の先頭の遷移種，四角枠は max，一重円は min を表す．パターン長を l，遷移種のコードを $\{a,b,c,\cdots\}=\{1,2,3,\cdots\}$ とする．図 1 の (i) において，$max=4, min=1$ から式 (1) より，$l=4-1+1=4$ である．まず，(i) の遷移種を 1 つ回転すると (ii) となり，$max=4, min=1$ より，$l=4$ である．次に，(ii) の遷移種を 1 つ回転すると (iii) となり，$max=3, min=1$ より，$l=3$ である．以上より，PAT の遷移種を 2 つ回転してパターン長を 3 に短縮する．

提案手法では，元々小さいパターン長を短縮してもその割合は小さく，効果よりも短縮処理の負荷の方が大きくなる．そこで閾値を設け，パターン長が閾値を超える遷移パターンにのみ短縮処理を行う．

また，提案手法では短縮した遷移パターンを用いてダブル配列を構築するため，本来の遷移種のコードでは遷移できない．そこで，遷移種に遷移パターン短縮と同様の処理を行うことで，遷移種のコードを変換する．

例として，遷移種 $c=3$ に遷移パターン $PAT=\{a,c,d\}$ の短縮と同様の処理を行う．PAT の短縮処理と同様に c を 2 つ回転すると $c=1$ となる．

3. 未使用要素分類と遷移パターン長短縮によるダブル配列構築法の評価

XCheck の比較回数削減を図った提案手法によるダブル配列の構築時間と記憶領域，検索時間を評価する．実験環境として Intel(R)Core(TM)i7 920 @ 2.67GHz，CentOS6.6 32bit，キーセットとして英語版 Wikipedia[7] のタイトル集合を使用する．文字コードは UTF-8 とする．タイトル集合からランダムに抽出

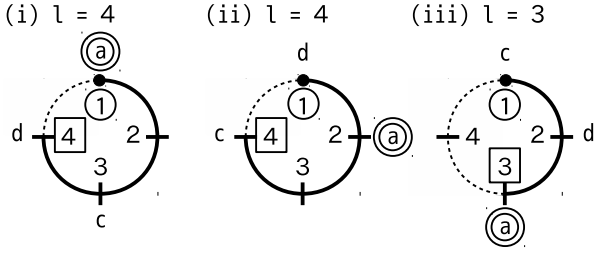

図 1: 遷移パターン $\{a,c,d\}$ の短縮過程

† 大阪教育大学，Osaka Kyoiku University

した 10 万〜100 万件を実験キーセットとする．未使用要素分類時に使用状況を調べる周辺要素の数を 2 とし，パターン長を短縮する閾値を 96 とする．比較手法として，中村らの手法 [5] を使用する．

実験キーセットを登録する際の構築時間を図 2 に，XCheck における比較回数を図 3 に示す．図 2 より，提案手法は比較手法に対して構築時間を 60〜90 %削減した．図 3 より，提案手法は比較手法に対して比較回数を 99 %削減した．ダブル配列の未使用要素を周囲の使用状況に応じて分類することで，XCheck における比較対象要素を限定し，パターン長を短縮することで，XCheck における未使用要素と遷移パターンの比較回数を削減し，構築時間を削減した．

記憶領域の効率を評価するにあたり，占有率 = (使用要素数)/(使用要素の最大 $index$) を定義する．占有率が高いほどダブル配列の記憶領域の効率が良い．提案手法と比較手法において，実験キーセットを登録した際の使用要素の最大 index と占有率を表 1 に示す．比較手法に対して提案手法の使用要素の最大 index は 40 %大きくなり，占有率は 30 %低下した．提案手法は，遷移パターンを短縮することにより，複数種の遷移パターンを短縮後の遷移パターンに集約する．XCheck 時，遷移パターンが必要とする未使用状況のリスト構造に偏りが生じるため，占有率が悪化した．

提案手法と比較手法において，実験キーセットを登録後，タイトル集合からランダムに抽出した 10 万件を検索する実験を行ったところ，遷移種に遷移パターンの短縮処理を再現するため，提案手法の検索時間は比較手法に対して 10〜40 %増加した．

4. おわりに

本稿は，基底値算出の高速化を図る手法として，未使用要素を周辺要素の使用状況に応じて分類する手法と，遷移パターンの遷移種を回転することでパターン長を短縮する手法を提案した．実験により，提案手法は基底値算出の比較回数を削減した．

今後の課題として，構築時間に対するより詳細な分析の他，短縮した遷移パターンや，未使用要素分類と遷移パターン短縮の関連性の分析などが挙げられる．

表 1: 最大 index と占有率

登録件数	提案手法		比較手法	
	最大 index	占有率	最大 index	占有率
200000	422783	0.698	298686	0.988
400000	849628	0.697	597949	0.991
600000	1278672	0.696	898346	0.991
800000	1709259	0.696	1200017	0.992
1000000	2147391	0.694	1501761	0.992

参考文献

[1] Aoe,J.：An Efficient Digital Search Algorithm by Using a Double-Array Structure, IEEE Transactions on Software Engineering, Vol.15, No.9, pp.1066-1077, 1989.

[2] 森田和宏，泓田正雄，大野将樹，青江順一：ダブル配列における動的更新の効率化アルゴリズム，情報処理学会論文誌，Vol.42，No.9，pp.2229-2238，2001.

[3] 大野将樹，森田和宏，泓田正雄，青江順一：ダブル配列による自然言語辞書の高速更新法，言語処理学会第 11 回年次大会発表論文集，pp.745-748，2005.

[4] 矢田晋，大野将樹，森田和宏，泓田正雄，吉成友子，青江順一：接頭辞ダブル配列における空間効率を低下させないキー削除法，情報処理学会論文誌，Vol.47，No.6，pp.1894-1902，2006.

[5] 中村康正，望月久稔:圧縮ディジタル探索における辞書情報更新の高速化手法，情報処理学会論文誌データベース，Vol.47，No.SIG 13(TOD 31)，pp.16-27，2006.

[6] 村山智也，望月久稔：ダブル配列構築の高速化を目的とした節点から遷移可能な遷移種の集合に基づく未使用要素の管理法，第 8 回データ工学と情報マネジメントに関するフォーラム (DEIM2016) 最終論文集，C6-4，2016.

[7] Wikipedia：https://www.wikipedia.org/, 2016/4/6 参照.

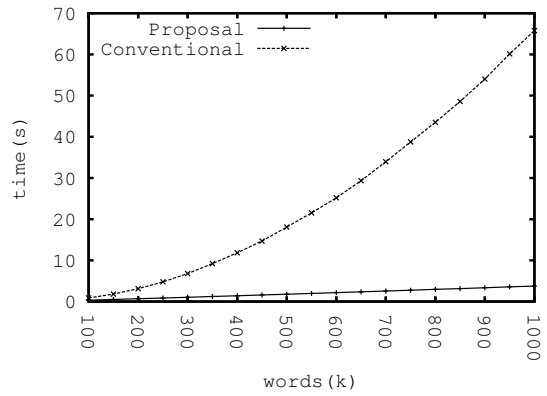

図 2: 構築時間

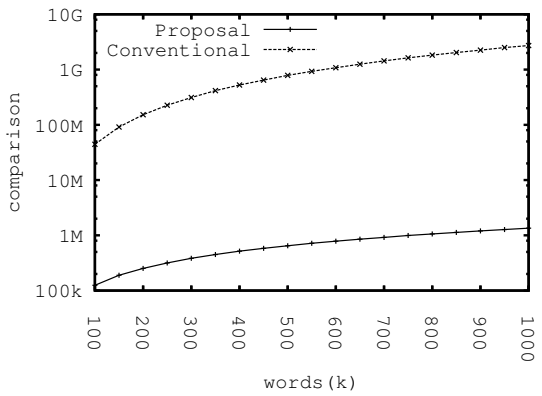

図 3: XCheck の比較回数 (片対数グラフ)