4L−3　順序保存符号によるデータ圧縮の実験結果

秋山奈美、中津楢男
愛知教育大学

1. はじめに

ﾃﾞｰﾀﾍﾞｰｽなど大量のﾃﾞｰﾀを記憶する場合、何等かのﾃﾞｰﾀ圧縮が必要とされる。瞬時に復号可能な静的符号化としては、ﾊﾌﾏﾝ符号化が最適であることが知られており、ﾃﾞｰﾀ圧縮の技法としての実用性も高い。しかし圧縮したﾃﾞｰﾀに対しある操作を施す場合には、原ﾃﾞｰﾀに復元する必要がある。だが、符号化の工夫によってはこうした復号操作が不用になる実用例も多く考えられる。ﾃﾞｰﾀﾍﾞｰｽでは２つの値の比較が本質的な操作と考えられる。この場合元のｱﾙﾌｧﾍﾞｯﾄ順序が符号語上でも保存されている順序保存符号化を用いれば、ﾃﾞｰﾀ同士の比較が符号語のままの比較操作で実現できる。順序保存符号の理論的考察は[1]で報告済みである。

本稿では、英語辞書ファイルを用いて、符号化しない場合、ハフマン符号化、順序保存符号化をした場合の各々について、圧縮率・検索時間を測定したのでその結果を報告する。

2. 順序保存符号とその応用

情報源をS、符号をｃとおき、瞬時に復号可能な符号だけを考える。また、符号語アルファベットは｛０、１｝とする。

［定義１］　ソースシンボルｘの符号語をｃ（ｘ）とする。ソースアルファベット上の全順序を≪、符号アルファベット上の全順序を＜とする（＜は２進文字列上の順序である）。∀ｘ，∀ｙ∈Ｓ，ｘ＜ｙ←→ｃ（ｘ）＜ｃ（ｙ）となる符号ｃを順序保存符号（order preserved code　以後ｏｐ符号と呼ぶ）と言う。

OP符号もﾊﾌﾏﾝ符号と同じく可変長符号であるため原則的には先頭から復号する必要がある。またOP符号の特徴である符号語間の比較を行うためには復号ﾌﾟﾛｾｽなしに符号語を認識できなければならない。固定形式ﾚｺｰﾄﾞではﾌｨｰﾙﾄﾞが固定長であるので符号語の認識は簡単である。可変形式ﾚｺｰﾄﾞの場合にはﾌｨｰﾙﾄﾞ長が可変であるため、ﾌｨｰﾙﾄﾞを区切るためにﾃﾞﾘﾐﾀ（区切り記号）を用いることにする。ところで、ﾃﾞﾘﾐﾀも１つの符号語であるため、ﾃﾞﾘﾐﾀと同じﾋﾞｯﾄ列が符号語の連接の部分列として現れる場合がある。このような部分列は、乱ｱｸｾｽを行う場合ﾃﾞﾘﾐﾀと見誤る可能性がある。そこで、符号語の連接の部分列としてﾃﾞﾘﾐﾀが現れず、どの部分から復号を始めてもﾃﾞﾘﾐﾀが正しく認識できるような符号をデリミタ用符号という。

3. 実験結果

利用したデータはapollo Domainワークステーションのスペルチェック用の英単語辞書（総単語数 42695：総文字数 374709：区切り ¥n）及びその索引ファイルである。この辞書の各文字の出現頻度を数え、その頻度をもとにハフマン符号化、最適OP符号化を行った。符号化による理論値は図１の通りである。

可変長符号化では単語が１バイトの途中から始まる場合もあるため、実際には

Experimental results of data compression by the order preserved coding
Nami AKIYAMA and Narao NAKATSU
Aichi University of Education

ASCII符号	ﾊﾌﾏﾝ符号	op符号	
374741	199955	204311	bytes

図１　各符号化によるファイルの大きさの理論値

単語はバイト境界で終了するよう必要なだけ０を補って符号化した。検索時間については次の２つの実験を行った。

１）辞書を　最適なデリミタ用ｏｐ符号で符号化したファイルopdict、及びデリミタ用ハフマン符号で符号化したファイルdhuffdictを作成し、圧縮しないままの辞書を含めた３つのファイルの大きさと、各々のファイルを２分探索で検索した場合の平均検索時間を測定した。

２）索引ﾌｧｲﾙを２分探索して、求める単語が存在する可能性のある番地の範囲を限定し（その単語が索引ﾌｧｲﾙにあればただちにその存在番地がわかる）、辞書ﾌｧｲﾙで次にその範囲を順アクセスで探索する。ハフマン符号の場合は復号しながら辞書を読みだすためデリミタ用ハフマン符号でなく普通のﾊﾌﾏﾝ符号を利用できる。

符号化後のファイルの大きさを図２に、検索時間結果を図３に示す。時間は、ランダムに選んだ20の英単語を検索するのに必要な時間を示している。実験用プログラムはapollo　DN4000でｃ言語を用いて開発した。

この結果よりop符号はアクセス時間の低下を招くことなくデータ圧縮を実現することが明らかになった。ハフマン符号とop符号を比較すれば、デリミタ用ハフマン符号とデリミタ用op符号で圧縮率はほぼ同程度、検索時間はデリミタ用op符号の方が若干速い（3%）。ﾊﾌﾏﾝ符号とデリミタ用op符号を比較すれば検索時間はデリミタ用op符号が8%程速かった。この時間差はﾊﾌﾏﾝ符号では復号操作が必要であるためと考えられる。この実験では必要な復号操作はそれほど大きな量でないが、θ結合演算の様にタップル同士の比較が大量に行われる場合や、順検索が頻繁に行われる場合にはこの差はより顕著になるものと考えられる。

より詳しい実験、日本語への応用等が残されている問題である。

ASCII符号	ﾃﾞﾘﾐﾀ用ﾊﾌﾏﾝ符号	ﾊﾌﾏﾝ符号	ﾃﾞﾘﾐﾀ用op符号	op符号	
374741	228492	200719	230643	207738	bytes

図２　各符号化によるファイルの大きさ

	ASCII符号	ﾃﾞﾘﾐﾀ用ﾊﾌﾏﾝ符号	ﾊﾌﾏﾝ符号	ﾃﾞﾘﾐﾀ用op符号
２分探索	１	１．１４	×	０．９８
索引探索	１	＊	１．０８	１．００

図３　検索時間の比（×は実行できず、＊は実験せず）

参考文献　[1]中津、「順序保存符号について」、信学技研報告IT89-56,1989.
[2]中津,「情報検索に適した符号：順序保存符号」,情報処理学会40回大会7H-1.