修士論文 要旨

順序木のシンプルな木被覆表現

数理情報学専攻 48-206215　郡山 巧人
指導教員　定兼 邦彦 教授

1　導入

簡潔データ構造とは、データを限界近くまで圧縮して格納しつつ、効率的にクエリを処理できるデータ構造である。ウェブページや DNA 配列といった巨大データを扱う際に、データをコンパクトに表現することは、空間計算量の削減だけでなく処理速度の向上にもなる。

有限集合 S の 1 要素を表現するための情報理論的下限は $\lceil \lg |S| \rceil$ ビットであり、これに加えて $o(\lg |S|)$ ビットの領域のみを使用する表現を簡潔であるという。簡潔データ構造は、簡潔表現とクエリ処理のための $o(\lg |S|)$ ビットの簡潔索引からなり、Jacobson [3] によって提案された。

本研究では、順序木（子の順番を区別する根付き木）の簡潔データ構造を扱う。順序木の簡潔データ構造はよく研究されており、LOUDS 表現 [3]、BP 表現 [4]、DFUDS 表現 [1] が代表的である。後に、クエリの種類ごとに別々の索引が必要になり簡潔索引の大きさが膨れ上がってしまう問題を解決する手法として Navarro, Sadakane によって区間最大最小木 [6] が提案された。

順序木全体のある部分集合の 1 要素を表現するために順序木の簡潔表現を用いると冗長なことがある。本研究では葉の個数が固定された順序木の部分クラスを扱い、BP 表現に基づく簡潔表現を与えた。

順序木を簡潔に表現するための別のアプローチとしては，木を木被覆 [2] で表現するものがある．これは木をマイクロ木と呼ばれる部分グラフに分割し，各マイクロ木を任意の簡潔表現を用いて表現し，マイクロ木の間の接続関係を別の木構造で表現するというものである．なお，木の分割ではなく被覆となっている理由は，分割にしてしまうとマイクロ木の個数が非常に多くなってしまい，簡潔表現が得られないことがあるからである．このアプローチの利点は，順序木の部分クラスに対する簡潔データ構造を容易に与えられることである．木被覆に基づく木の簡潔データ構造の実用上の問題点は，マイクロ木の間の接続関係や，共有されたノードを単純かつコンパクトに表現する方法が非自明であることである．そのため，実装は存在せず，理論的には $o(n)$ ビットであるこれらの索引が実際にはどの程度の大きさになるかが分かっていないという問題点があった．

本研究では，3 種類の括弧からなる括弧列を用いた木被覆を単純かつコンパクトに表現する手法を提案した．順序木に対する各種クエリは区間最大最小木に基づく索引により括弧列による表現とほぼ同様のアルゴリズムで実現でき，索引のサイズも小さいままである．

木の部分クラスの一つである，非順序木について，簡潔表現を与えた．節点数 17 以下の非順序木を列挙し，それに対し符号を割り当てる表を作成した．非順序木を節点数 17 以下のマイクロ木に分割し，各マイクロ木をこの表を用いて符号化し，マイクロ木間の接続関係は括弧列で表現した．実験により，非順序木のサイズの情報理論的下限である約 $1.56n$ ビットに近いサイズの単純な表現を得られることを示した．

2　k-葉木

n 節点の順序木のうち k 個の葉を持つものを k-葉木と呼ぶ。k-葉木の総数は Narayana 数 [5] として知られ、情報理論的下限は $2n\mathrm{H}\left(\frac{k}{n}\right) - \Theta(\lg n)$ ビットである（H はエントロピー関数）。

本研究では k-葉木の BP 表現をビット列とみなし階差を取ることで疎な構造が現れることを見出し、疎なビットベクトルに対する簡潔データ構造 FID [8] を用いることで $2n\mathrm{H}\left(\frac{k}{n}\right) + \mathrm{O}(\frac{n \lg\lg n}{\lg n})$ ビットの表現をを与えた。簡潔索引に区間最大最小木 [6] を用いることで $\min\{k, n-k\} = \omega(\frac{n}{\lg n})$ を満たす k に対し簡潔データ構造を得ることができる。

3　修正木被覆

本研究では，木被覆を単純かつコンパクトに表現する手法を提案する．これは，3 種類の括弧 (), {}, [] を用いる括弧列 P による表現である．この括弧列 1 つが木被覆の構造についての全ての情報を含んでいる．通常の括弧 () と中括弧 {} は元の木に存在する節点を表し，大括弧 [] は木に追加したダミー節点を表す．よって，P から大括弧を削除し，中括弧を通常の括弧に変換すると元の木の括弧列表現が得られる．また，P の大括弧と中括弧のみを残した括弧列を考えると，それはマイクロ木の間の接続関係を表す木になっている．これにより，従来はポインタを用いて表現していたマイクロ木の間の接続関係をコンパクトに表現できる．

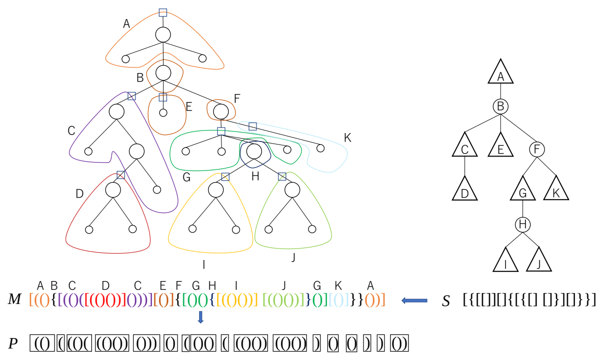

図 1. 修正木被覆の例

節点数	簡潔表現のビット数
10	11.53
100	171.4
1000	1835.97
10000	18542.12
100000	185539.02

表 1. 非順序木に対する実験結果

4 数値実験

非順序木とは、子の順番を区別しない根付き木のことであり、情報理論的下限は約 1.56n ビットである [7]。非順序木は子に適当な順番を付けることで順序木として表現できるので順序木の部分クラスとみなすことができる。

本研究では実際に非順序木に対し修正木被覆に基づく簡潔表現を作成し、その大きさを測定した。いくつかの節点数についてランダムな非順序木を 100 回生成し平均を取った結果を表 1 に示す。順序木の BP 表現よりも少ないビット数で表現できていることがわかる。

参考文献

[1] D. Benoit, E. D. Demaine, J. I. Munro, R. Raman, V. Raman, and S. R. Satti. Representing Trees of Higher Degree. *Algorithmica*, 43(4):275–292, 2005.

[2] M. He, J. I. Munro, and S. R. Satti. Succinct ordinal trees based on tree covering. 8(4):1–32, 2012.

[3] G. Jacobson. Space-efficient Static Trees and Graphs. In *Proc. IEEE FOCS*, pages 549–554, 1989.

[4] J. I. Munro and V. Raman. Succinct Representation of Balanced Parentheses and Static Trees. *SIAM Journal on Computing*, 31(3):762–776, 2001.

[5] T. V. Narayana. A partial order and its applications to probability theory. *Sankhyā*, 21:91–98, 1959.

[6] Gonzalo Navarro and Kunihiko Sadakane. Fully-Functional Static and Dynamic Succinct Trees. *ACM Transactions on Algorithms*, 10(3):Article No. 16, 2014.

[7] Richard Otter. The number of trees. *Annals of Mathematics*, 49(3):583–599, 1948.

[8] R. Raman, V. Raman, and S. R. Satti. Succinct indexable dictionaries with applications to encoding k-ary trees, prefix sums and multisets. 3(4), 2007.