

速度 人類の永遠の追求

相 利民
Limin XIANG

九州産業大学情報科学部社会情報システム学科
Department of Social Information Systems, Faculty of Information Science, Kyushu Sangyo University
xiang@is.kyusan-u.ac.jp, <http://www.is.kyusan-u.ac.jp/~xiang/>

1. はじめに

ある角度から見ると、人類社会の発展は人類が速度を追求する過程である。人間の寿命は有限の時間であるので、速度が上がると、相対的に人間の寿命は長くなる。例えば、福岡から京都へ行くのに、昔は数日間が掛かったが、今は数時間しかが掛からず、将来は更に短くなると予想される。即ち、人間の寿命は昔より相対的に数十倍から数百倍に長くなっている。このように人類は生産の速度や移動の速度や計算の速度などを追求し、生産の道具や飛行機やコンピュータなどを発明した。

情報社会になる現在、明らかに、情報処理の速度がますます要求される。例えば、インターネットから資料を調べるとき、数分間でも待ちたくない。情報処理の速度を上げる方法は、計算機の CPU の速度を上げる方法以外にも考えられる。その中で、

並列計算の利用、および
効率的なアルゴリズムの採用

という方法もある。並列計算と効率的なアルゴリズムは当研究室の二つの研究分野である。次にそれぞれを紹介する。最後に、当研究室の情報科学基礎演習を紹介する。

2. 並列計算

高性能計算 (HPC) を必要とする領域は、中長期の天気予測や遺伝子の解明などきわめて多い。高性能計算のかぎの技術は並列計算である。一方、一つの集積回路のトランジスタの個数は Moore の法則 (即ち、18ヶ月毎に倍になる) により 1971 年の 2,250 から 2000 年の 42,000,000 に増えて、Moore の法則は新しい世紀に入っても有効である (<http://www.intel.com/research/silicon/mooreslaw.htm>)。すなわち、応用の要求とハードウェア技術の発展が並列計算に関する研究の 2 つの原動力である。

文献中の並列計算アルゴリズムはほとんど PRAM (Parallel Random Access Machine) に属する並列計算モデルで書くアルゴリズムである。PRAM は並列計算の最も知られたモデルである。BSR (Broadcasting with Selective Reduction) モデルはカナダの学者 S.G. Akl と G.R. Guenther が 1989 年に最初に提案した PRAM に属する並列計

算モデルである。このモデルは CRCW (Concurrent Read Concurrent Write) より強力である。1994 年には Akl と Stojmenovic が BSR における BROADCAST 命令の選択基準を 1 個から k 個に拡張して BSR k を提案した。BSR k では k 個の選択基準を組合せるのは AND 操作だけである。

BSR を多くの応用に対してさらに強力にするために、我々が BSR に基づいて 1999 年 8 月に並列・分散計算分野の代表的な国際学術誌 IEEE Transactions on Parallel and Distributed Systems [1] で BSR+モデルを提案し、BSR k の実現法より効率的な BSR+実現法を示した。BSR + モデルは BROADCAST 命令において選択の一般形、特に任意の選択基準を任意の論理演算で結合することを許すようにする。結局計算能力はすべての PRAM モデルの中で BSR+が最も強力である。

BSR 上では様々な応用に対して定数時間解を与えてきたが、BSR が提案されたときに提起された、BSR 上でのいくつかの理論的問題がまだ解決されていない。そのうちで (1)BSR 上の任意の問題には自明な定数時間を除いて下限が知られていない、(2)BSR 上での非定数時間で、CRCW の下限よりは良い改良があるか (3) BSR が定数時間性能を達成するような問題をいかに特徴付けるか、という三つがある。我々は 2001 年 8 月の IEEE Transactions on Parallel and Distributed Systems [2] で、これら三つの問題を解いている。問題 (1) に対して BSR 上の下限を最適逐次解を持つ任意の計算問題に対して示している。問題 (2) に対しては能率のよいソーティングアルゴリズムで答えている。問題 (3) に対しては必要条件をサイズ n と、使用可能なプロセッサの数 p とが異なる場合に、BSR 性能を論じるために BSR に対する Work-Time スケジュール作成原理と最適性という概念を導入している。

並列計算モデルは現存または未来の並列計算機の抽象である。BSR モデルは後者に属する。並列計算モデルの重要な役割の一つは問題を解決する並列アルゴリズムを容易に書くことができることにある。色々な問題 (例えば、ソーティング問題やデータ構造に関する問題など) に対する時間最適な BSR (BSR +) の並列計算アルゴリズムが発表された。しかし、BSR (BSR +) の並列計算アルゴリズムが実行できる並列計算機はまだ存在していな

い。一方, LAN の普及と PC の低価格化によって研究室の範囲で数台の PC を用いて BSR(BSR+) の並列計算アルゴリズムが実行できる環境を作ることが可能になってきた。『強力な並列計算モデル BSR+ のアルゴリズム実行環境の実現に関する研究』は平成 14 年度科学研究費補助金に採択された 2 年間の本研究室の研究課題である。

3. 効率的なアルゴリズム

情報社会が進み, 計算機が社会の隅々まで行き渡るようになった。計算機を動かすにはプログラムが必要である。プログラムを書くにはアルゴリズムの設計が不可欠である。アルゴリズムの効率は対象となるオブジェクトの表現と密接に関連している。当研究室では組合せオブジェクトを中心として, 効率的なアルゴリズムを研究している。

代表的なデータ構造である二分木の生成問題に対して, 我々は文法で表現する方法とその表現を用いた効率的なアルゴリズムを The Computer Journal 誌に発表した [3]。二分木の表現法に対して伝統的な整数系列による方法に対して, 二分木に対しては文法 GBT に基づいて, 完全二分木に対しては文法 GFBT に基づいて, 文字系列 (語) で表す方法を与えている。GBT(GFBT) から導かれる語の性質を議論し, GBT(GFBT) の語を辞書式順はじめいくつかの方法で数え上げるアルゴリズムを提示しこれを解析している。GBT の語を数え上げるアルゴリズムを変更して二進木をコンピュータ上の表現で数え上げるアルゴリズムを得た。木当たりの平均時間は $O(1)$ である。

文法で表現する方法を k 分木に拡張して, 同誌に発表した [4]。通常の k -進木に対する系列を能率的に生成するために本論文では再帰的アルゴリズム GenWordsR と非再帰的アルゴリズム GenWordsNR を提示している。両者を, 文献にみられる従来の再帰的アルゴリズムと非再帰的アルゴリズムのいくつかと比較している。 k -分木を生成するための再帰的アルゴリズムの複雑さを測る尺度として再帰呼び出しの平均回数を用いると従来の再帰的アルゴリズムでは k -進木に対して $O(k)$ 回呼び出しが最善の結果である。これに対して GenWordsR は k -分木に対して $O(1/k)$ 回呼び出しを必要とする。 K -分木を生成するための非再帰的アルゴリズムの時間複雑度を測る尺度として平均比較回数を用いると GenWordsNR は従来の非再帰的アルゴリズムを凌駕する。 k が増加するにつれて GenWordsNR が行う比較回数 (平均比較回数) は従来の最善非再帰的アルゴリズムの回数の 66% に近づく。

2 分木や k 分木や順列や組合せや巡回置換などのいろいろな組合せオブジェクトがある。組合せオブジェクトの生成問題に対して, 上記の 2 分木と k 分木の文法で表現する方法と効率的な生成アルゴリズムを提案した後, 我々は組合せの生成問題に対して, 組合せ正準表現の新しい暗号木で組合せリストを表現する方法と効率的な $O(1)$ 時

間生成アルゴリズムを 2001 年 8 月 The Computer Journal 誌に発表した [5]。Eades と McKay のアルゴリズムでは r 個から n 個を取る組み合わせ $C(r,n)$ を正準表現で SMCP をもって列挙できるが最悪の場合に組み合わせ一つ当たり $O(n)$ 時間を要してしまう。Takaoka のアルゴリズムでは $C(r,n)$ を正準表現で生成するための $O(1)$ 時間アルゴリズムでは組み合わせを強最小変更性 SMCP (Strong Minimal Change Property) で列挙できない。我々は $C(r,n)$ を正準表現で列挙する SMCP をもったアルゴリズムのオーダーを論じ, ループのないアルゴリズムを与えている。また, 長さ $2n$ の well-formed parenthesis string を列挙するためにループのないアルゴリズムも与えている。二つのアルゴリズムは以前のものに比べて約 $1/2$ に時間を短縮する組合せの効率的な生成アルゴリズムである。さらに, 我々のアルゴリズムは異なったオーダーのオブジェクトを生成するために容易に変更可能である。

効率的なアルゴリズムの追求はアルゴリズムに関する研究が求める重要な目標の一つである。効率的なアルゴリズムがなければ, 巨大データベースや画像処理や生命分子計算などの計算システムの実現とその性能の向上は不可能である。

4. 情報科学基礎演習の紹介

アルゴリズム設計は情報処理の重要な基礎です。アルゴリズム設計の原理と方法や関連する後続講義 (例えば, プログラミング基礎, データ構造とアルゴリズム, データベースなど) を理解しやすいように, アルゴリズム設計・プログラミングに関する基礎演習を行います (プログラミング言語 Delphi Pascal のコンソールアプリケーションに基づきます)。

- 第 1 回 . (ガイダンス)
- 第 2 回 . コンピュータと簡単に対話
- 第 3 回 . 整数の比較
- 第 4 回 . 1 位の十進数から二進数への転換
- 第 5 回 . 素数の判定, 列挙
- 第 6 回 . 最大公約数を求める
- 第 7 回 . 名簿検索 (逐次探索)
- 第 8 回 . 名簿検索 (2 分探索)
- 第 9 回 . 整列 (ソート)
- 第 10 回 . 再帰アルゴリズムで最大公約数を求める
- 第 11 回 . 部分集合生成アルゴリズム
- 第 12 回 . 簡単なゲーム

◇ 参 考 文 献 ◇

- [1] Limin XIANG, Kazuo USHIJIMA, Selim G. AKL, and Ivan STOJMENOVIC, An Efficient Implementation for the BROADCAST Instruction of BSR+, IEEE Transactions on Parallel and Distributed Systems, vol.10, no.8, 1999, pp.852-863.
- [2] Limin XIANG and Kazuo USHIJIMA, On time bounds, work-time scheduling principle and optimality for BSR, IEEE Transactions

- on Parallel and Distributed Systems, vol.12, no.9, 2001, pp.912-921.
- [3] Limin XIANG, Kazuo USHIJIMA, and Changjie TANG, Grammar-oriented enumeration of binary trees, The Computer Journal, vol.40, no.5, 1997, pp.278-291.
- [4] Limin XIANG, Kazuo USHIJIMA, and Selim G. AKL, Generating regular k-ary trees efficiently, The Computer Journal, vol.43, no.4, 2000, pp.290-300.
- [5] Limin XIANG and Kazuo USHIJIMA, On $O(1)$ time algorithms for combinatorial generation, The Computer Journal, vol.44, no.4, 2001, pp.292-302.