論文の内容の要旨

論文題目　Generation of Efficient Algorithms for Maximum Marking Problems
(和訳　最大マーク付け問題の効率的解法の自動生成)

氏名　篠埜 功

本論文では, 最大マーク付け問題という最適化問題の効率の良いアルゴリズムの自動生成を行う. 最大マーク付け問題とは, 「要素に重みの与えられたあるデータ x が与えられたときに, 性質 p を満たすマーク付けの中で与えられた重み関数 w の値を最大にするものを求める」という問題である. この問題は, ナップサック問題, データマイニングの分野における最適連想規則問題等, 現実の問題を含む, 広範囲の問題を例として含んでおり, 効率の良いアルゴリズムを求める手法を与えることは重要である.

効率の良いプログラムを正しさの明らかな仕様から導出することによって, 正しくかつ効率の良いアルゴリズムを得るというプログラム導出は, プログラム構築の一つの有用な方法である. プログラム導出を用いると, 仕様の変更に柔軟に対応できるという利点がある. 一般にはプログラマが考えて効率の良いプログラムを作成するが, その場合, 仕様が変更されると, プログラムを正しく修正することは容易でない. それに対し, プログラム導出を用いると, 仕様を変更するだけで, 導出を同様に行えば, 効率の良いプログラムを得ることができる.

最大マーク付け問題は, 再帰的グラフ (decomposable グラフ) という, 制限されたグラフ上で考えられていた, 最大部分 (頂点または枝) 集合問題を, マークを付ける問題として一般化し, 再帰データ上の問題として定式化しなおしたものである. 再帰的グラフを入力とする最大部分集合問題には, 性質 p が正則な述語であれば, 線形時間アルゴリズムが存在することが Bern らによって発見された. さらに, 正則な述語を, 基本的な述語と, $\forall$, $\exists$, $\wedge$, $\vee$, $\neg$ を用いて作り上げられる単項二階論理式によって記述し, これから, 線形時間アルゴリズムを機械的に導出する方法が Borie らによって考案された. この論理式の記述力は高く, 頂点被

覆問題, 巡回セールスマン問題等, 広範囲のグラフ上の問題を記述でき, 理論的には興味深いが, 素朴に線形時間アルゴリズムを導出するため, 非常にサイズの大きなテーブルが生成され, 計算量の定数係数が非常に大きくなり, 全く実用にならないという問題点があった. 我々の提案する方法は, この問題点を解決し, 係数の小さい線形時間アルゴリズムを機械的に導出する. 具体的には, 例えば, 最大部分列和 (maximum segment sum) 問題という問題においては, 上記の Borie らの方法によるアルゴリズムと我々の提示する方法によるアルゴリズムを同一の基準で計算量評価すると, 計算量の係数が, Borie らの方法では $(4^{(2^{(2^{619})})})^2$ となるのに対し, 我々の提示する方法では 8^2 となる.

我々のアプローチのポイントは, 性質 p を再帰関数で記述するところにあり, これにより, 再帰関数上でのプログラム変換を行うことができるようになる. 本論文において提案する主要な定理は, 「性質 p が, 有限の値域を持つ, mutumorphisms という再帰関数群で表されれば, 最大マーク付け問題を解く線形時間アルゴリズムが存在し, それを機械的に導出できる」という最適化定理である. 有限の値域を持つ, mutumorphisms という関数群で性質 p を表すという条件は非常に緩い条件であり, 柔軟に広範囲の性質を記述できる. 導出されるアルゴリズムはテーブルを用いるものであるが, そのサイズが多くの場合において十分小さく, 実用的なアルゴリズムとなる. テーブルサイズは mutumorphisms 中のそれぞれの関数の値域のサイズで決まるが, 多くの性質は, 少数の述語 (真偽値を返す関数) からなる mutumorphisms によって記述可能であるので, テーブルサイズが小さなアルゴリズムとなる.

例として, リスト上の, 最大独立部分列和問題を考えてみる. 最大独立部分列和問題とは, 重みを持つ要素からなるリストを入力として受け取り, どの 2 つの要素も隣接していないような要素集合のうち, 重み和が最大のものを求めるという問題である. 性質 p は, マークの付けられた要素中には隣接している要素はないという性質であり, これをリスト上の再帰関数として記述すればよい.

$$
\begin{array}{lcl}
p\ [x] & = & \mathit{True} \\
p\ (x : xs) & = & \textbf{if}\ \mathit{marked}\ x\ \textbf{then}\ p_1\ xs\ \textbf{else}\ p\ xs \\
p_1\ [x] & = & \mathit{not}\ (\mathit{marked}\ x) \\
p_1\ (x : xs) & = & \mathit{not}\ (\mathit{marked}\ x)\ \wedge\ p\ xs
\end{array}
$$

関数 p, p_1 はリスト上の有限 mutumorphisms を成しており, 最適化定理により, 線形時間アルゴリズムが機械的に得られ, そのテーブルサイズは 8 である. 一方, この性質 p を単項二階論理式で書くと,

$$
\begin{array}{l}
p = \forall v_1\ \forall v_2\ \neg\ \mathit{Adj}\ (v_1, v_2) \\
\mathit{Adj}\ (v_1, v_2) = \exists e_1\ (\mathit{Inc}\ (v_1, e_1)\ \wedge\ \mathit{Inc}\ (v_2, e_1))\ \wedge\ \neg\ (v_1 = v_2)
\end{array}
$$

のように記述されるが, テーブルサイズは $2^{(2^{142})}$ 以上となる.

提案した最適化定理は, 広範囲に渡るさまざまな最適化問題に適用することができ, 簡潔な仕様から, 実用的線形時間アルゴリズムが機械的に得られる. 適用例は, ナップサック問題, 段落構成問題, データマイニングにおける最適連想規則問題など多岐にわたる. 最適連

想規則問題に対しては, 既存のアルゴリズムと効率は同等であるが, 新たなアルゴリズムが導出された.

また, 本研究で提案する手法を, 実際に仕様から線形時間アルゴリズムを生成する自動生成システムとして実現する. MAG 等の既存の変換システムを用いても実現は可能であるが, 戦略の記述ができない等不自由な点があるため, 新たに自動生成システムを作成する. このシステムは, Calculation Carrying Program という既存の研究に基づいて作成されている.

Bird, de Moor により, あるクラスの最適化問題 (最大マーク付け問題を含む) に対し, 関係を用いた仕様記述から効率の良いアルゴリズムを導出する, Thinning Theory という理論が提案されている. これは, 非常に広範囲の問題を扱えるが, 導出に人間の洞察力が必要である. 本研究は, 最大マーク付け問題という問題クラスに注目し, 機械的に導出する手法を与えている. 我々の方法では, 性質 p が有限の値域を持つ関数からなる mutumorphisms で記述できさえすれば, あとは機械的に線形時間アルゴリズムに変換することができる.

我々の方法の適用範囲は広いが, 入力データが再帰的グラフで書けない場合には適用できない. 例えば, 2 次元格子は, いくらでも大きな木幅をもつものが存在し, 再帰的グラフでは表すことができないので, 最大部分列和問題の 2 次元版の問題には適用できない. 再帰的グラフは確かに制限されたグラフではあるが, 応用可能範囲は広い. 一例として, 構造的プログラムの線形時間制御フロー解析が考えられる. 構造的プログラムの制御フローグラフは再帰的グラフで表されることが知られており, 最大マーク付け問題で定式化できる問題であれば, 我々の方法が適用可能である.