アニーリング法による最適な型付きλ式の導出

5G−3

澤部 直太　東条 敏

情報処理振興事業協会 (IPA)

1 はじめに

多数部品から最適な構造物を構成する場合、最適な組み立て順序が存在する。しかし、部品の構造や制約条件などが複雑になると最適な組み立て順序の探索が困難になる。一方、複数プロセスの協調によって同時多発的に組み立てを行なえば、高速な構造生成が可能になる。

本研究では、型付きλ式で表現された複数部品から最適な構造を形成する一方式として、アニーリング法による崩壊・再構成機構を提案する。

2 型付きλ計算

2.1 型付きλ式のリダクションと単一化負荷

2つの型付きλ式 $e_1:\sigma_1, e_2\sigma_2$ をリダクションするとき、単一化可能であれば、

$$\sigma_1\theta = \sigma_2\theta$$

となる単一化子 θ が存在する。ここで、J.A.Robinsonの単一化アルゴリズム [1] を用いると、最汎単一化子 (most general unifier) θ' を求めることができる。

全ての単一化子に対して、その単一化の難しさを「負荷」として数値化しておく。2つの型付きλ式をリダクションしたとき、そこで求められた最汎単一化子の負荷を合計したものをそのリダクションの「単一化負荷」とする。

2.2 複数の型付きλ式からの構造生成

複数のλ式を含む集合の初期世代 Γ_0 を仮定し、λ式同士のリダクションによって世代が進むものとする。つまり、世代 Γ_n に含まれるλ式の集合から、2つのλ式を選択してリダクションしたものを世代 Γ_{n+1} とする。

任意のλ式同士をリダクションさせると、タイプエラーを起こす場合がある。このうち、一般的にはタイプエラーであるが、明示的な型変換を行なうことによりリダクション可能となる場合がある。

本研究では以下に挙げる2種類の型変換と、それに対応するコンビネータを導入した。

2.2.1 λ変数の順序の付け換え

次の型付きλ式を考える。

$$M = \lambda_{xy}.f(x,y) : \alpha \to \beta \to \gamma$$
$$N_1 = a : \alpha$$
$$N_2 = b : \beta$$

M と N_1 では

$$MN_1 = \lambda_y.f(a,y) : \beta \to \gamma$$

のように適用可能だが、N_2 は M に適用できない。ここで、$f(x,y)$ において x,y が順序によらず適用可能であると仮定すると、λ変数の順序を無視することができ、その順序を入れ換えても意味的には変わらない。

そこで、λ変数の順序の付け換えを行なうコンビネータ C [3] を導入する。C は一般には次のように定義される。

$$\mathbf{C}F = \lambda_x[\lambda_y[Fyx]]$$

本研究では C を拡張したコンビネータ $\mathbf{C_n}$ を以下のように定義する。

$$\mathbf{C}_1F = \lambda_x[\lambda_y[Fyx]]$$
$$\mathbf{C}_2F = \lambda_x[\lambda_y[\lambda_z[Fzyx]]]$$
$$\mathbf{C}_3F = \lambda_x[\lambda_y[\lambda_z[\lambda_w[Fwyzx]]]]$$
$$\vdots$$

$\mathbf{C_n}$ を用いることにより M, N_2 は次のように適用できる。

$$\mathbf{C}_1MN_2 = \lambda_x.f(x,b) : \alpha \to \gamma$$

2.2.2 λ変数を含むλ式の適用

初期世代 Γ_0 が次のλ式から構成されている場合を考える。

$$M = \lambda_{xy}.f(x,y) : \alpha \to \beta \to \gamma$$
$$N = \lambda_x.g(x) : \delta \to \beta$$
$$O = a : \alpha$$
$$P = d : \delta$$

これらのλ式がすべて結合した世代 Γ_n を作成するには

$$MO(NP)$$

という順序で構成される必要がある。ここで O と (NP) の順序は、前述のコンビネータ C により交換可能であるが、(NP) の順序を守らなければタイプエラーを起こしてしまう。そこで、N に P が適用される前に N を M に適用可能にするために、コンビネータ B[3] を導入する。B は、一般には次のように定義される。

$$\mathbf{B}FG = \lambda_x[F(Gx)]$$

本研究ではこれを拡張したコンビネータ $\mathbf{B'}$ を以下のように定義する。

$$\mathbf{B'}FG = \lambda_{x_G}[F(Gx)]$$
$$\mathbf{B'}FG = \lambda_{x_{GyG}}[F(Gxy)]$$
$$\mathbf{B'}FG = \lambda_{x_{GyGz_G}}[F(Gxyz)]$$
$$\vdots$$

各式において λ_{x_G} のようにλ変数に付けられた添字 G は、そのλ変数がλ式 G に適用されることを示している。この添字は後述するλ式の崩壊の際に活用される。

$\mathbf{B'}$ を用いると M, N は次のように適用できる。

Derive the optimal typed λ-formulae with annealing.
Naota SAWABE, Satoshi TOJO
Information-technology Promotion Agency.(IPA)
((株)三菱総合研究所より出向)

$\mathbf{B}'(\mathbf{C}_1 M)N = \lambda_{x_N}.\lambda_y.f(g(x_N), y) : \delta \to \alpha \to \gamma$

2.3 崩壊・再構成

リダクションによって構成したλ式を元のλ式に分離することを「崩壊」と呼ぶことにする。崩壊はリダクションの時に求めた最汎単一化子の情報にしたがって行なう。

一般に2つのλ式を結合したものは次のように崩壊させることができる。

$$\frac{MN = \lambda_y.f(a,y) : \beta \to \gamma}{M = \lambda_{xy}.f(x,y) : \alpha' \to \beta \to \gamma \qquad N = a : \alpha''} \; mgu(M,N) = u(\alpha', \alpha'')$$

本研究ではコンビネータ $\mathbf{B}'$ を導入したため、λ変数に添字が付けられる。そこで、実際には次のように崩壊が行なわれる。

$$\frac{MN(=\mathbf{B}'MN) = \lambda_{x_N}.\lambda_y.f(g(x_N),y) : \delta \to \beta \to \gamma}{M = \lambda_w.\lambda_y.f(w,y) : \alpha' \to \beta \to \gamma \qquad N = \lambda_{x_N}.g(x_N) : \delta \to \alpha''} \; mgu(M,N) = u(\alpha', \alpha'')$$

崩壊によって分離したλ式は、再構成を行なう対象となる。

3 アニーリング法による崩壊・再構成の制御

与えられたλ式の集合の各要素から最適な構造を構成するためには、構成、崩壊を制御する必要がある。ランダムに構成していったのでは局所最適な構造に陥ってしまうので、その構造を崩さなければならない。最汎単一化子ごとに定義した負荷が簡単なものであったとしても、もっとも負荷が少ない場合のみを選んで構成を進めても局所最適な状態に陥りうる。また、崩壊を起こしやすくすると、求めたい構造になかなか到達しない。

本研究では崩壊・再構成を制御するのにアニーリング法 [2] を用いる。つまり、構成の初期世代においては崩壊が起こりやすくし、負荷が最小ではないものが選ばれやすくしておく。世代が進むにつれて崩壊が起こりにくくし、負荷が最小のものが選ばれやすくする。

4 例題

複数単語から正しい文を構成することを考える。単語を型付きλ式で表現し、いくつかの単語を構成した結果として正しい文を作り出す。

例えば、表1に挙げた単語の集合から、

ケンは　ナオミに　本を　読m　aせる。

Ken-wa Naomi-ni hon-wo yom- aseru.

という文を構成した。この文は図1のような構造を持っている。構成の途中では局所最適と思われる文が生成されるが、最終的には正しい文が導き出された。

表 1: 単語の型付きλ式表現

Ken-wa	*Ken*:nom
Naomi-ni	*Naomi*:dat
hon-wo	*the_book*:acc
yom-	$\lambda_{xyz}.read(x,y,z) : agt \to cgt \to obj \to t$
aseru	$\lambda_{xy}.cause(x,y) : agt \to t \to t$

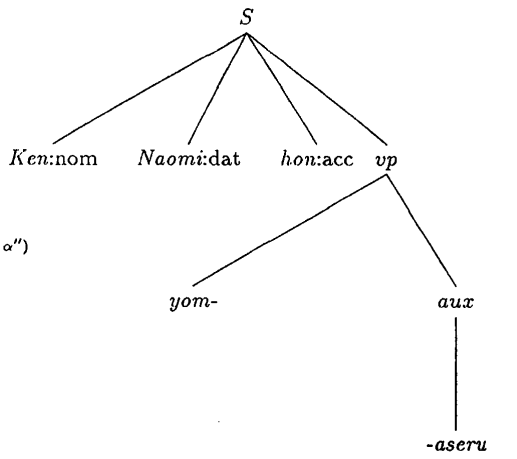

図 1: 文の木構造

5 今後の課題

例題として自然言語の文生成を挙げたが、今後の課題としてはより適切な応用分野に対する適用可能性を検討することが挙げられる。提案したアルゴリズムでは対象世界を型付きλ式で表現するが、次のような条件を満たしている分野への適用が有効であると考えている。

1. 数多くのλ式が結合することにより全体構造を作り出すこと。
2. λ式が単純でなく、種類も豊富なこと。
3. 全体構造が決定した後で、新しいλ式が追加され、それにより全体構造が大きく変化すること。

6 謝辞

本研究は、次世代産業基盤技術研究開発「新ソフトウェア構造化モデルの研究開発」の一環として情報処理振興事業協会 (IPA) が新エネルギー・産業技術総合開発機構から委託を受けて実施したものである。

参考文献

[1] Michael R. Genesereth and Nils J. Nilson.: "Logical Foundation of Artificial Intelligence" pp.66–69, Morgan Kaufmann(1987).

[2]

[3] Curry,Haskell B. and Robert Feys.: "Combinatory Logic,Vol.I" North Holland,Amsterdam. 深尾 毅: 最適化問題の確率化と熱力学, 計測と制御, 29-12, pp.1077–1085 (1990).