

関数型プログラムの束縛評価グラフ生成ツール

5 A E - 6

合田和正 程京徳 牛島和夫

九州大学大学院 システム情報科学研究科 情報工学専攻

1. はじめに

関数型プログラムは、記述が簡潔で、ハードウェアの仕様記述や定理自動証明に代表される重要な応用があるにもかかわらず、命令型プログラムと比較してあまり普及してこなかった。その理由の一つは、関数型プログラムの開発支援環境があまり整備されていないことである。

現在、関数型プログラム開発活動を支援するツールはいくつか存在するけれども、複雑さ計測^[1]といった個々のプログラム開発活動に特化されており、プログラム開発活動を統合的に支援する環境はまだない。

本研究は、関数型プログラムの開発活動をソフトウェア工学の立場から支援する環境の構築を目指している。そのような環境を構築する際には、プログラムを抽象的に表現するモデルが重要となる。そのモデルの一つとして、著者らが提案した束縛評価グラフ^[2]がある。本発表では、関数型プログラムからその束縛評価グラフを生成するツールについて述べる。

2. 束縛評価グラフ

束縛評価グラフ^[2]は、節点と枝を分類した有向グラフであり、関数型プログラムの抽象的な表現モデルである。節点は、プログラムの宣言、照合規則、部分式の集合に対応し、データに関する情報を持つ。節点におけるデータに関する情報とは、評価される式、束縛されるパターン、関数を示す式、引数を示す式、照合規則で照合されるパターン照合規則でパターンに対応する式である。節点間の関係を表現する順序弧は、関数適用や局所宣言のように環境の変化を伴う関係を表現する環境順序弧、ある式とその部分式との関係を表現する構造順序弧、その他の関係を表現する要素順

序弧から構成される。関数型プログラムでは、環境、構造のそれぞれに階層を考慮することができる。これを深度と呼ぶ。Standard ML^{[3],[4]}プログラムの例を図1に、その束縛評価グラフを図2に示す。

```

val a = 5;
val b = a;
val c = fn x => b;
val d = fn 0 => 7
      | y => c y;
val e = d 3;
    
```

図 1: プログラムの例

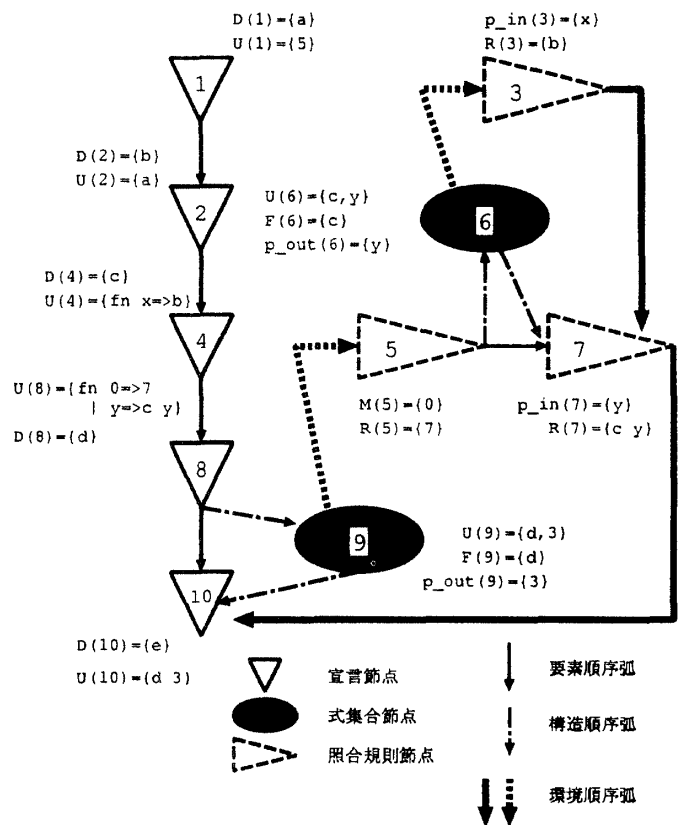


図 2: プログラムの束縛評価グラフ

A generator of binding evaluation graph for functional programs
 Kazumasa Gouda, Jingde Cheng, and Kazuo Ushijima
 Dept. of Comp. Sci. and Comm. Eng.,
 Kyushu University

3. 束縛評価グラフ生成ツール

束縛評価グラフ生成ツールは、プログラムを入力として束縛評価グラフを生成する。現在は、Standard ML プログラムを対象としている。

このツールは、yacc.lex と C で実現し、Standard ML の構文規則にしたがって入力プログラムを解析する。出力する束縛評価グラフの書式は以下のようになっている。

節点番号 節点の種類 行番号 データに関する情報
 弧番号 弧の種類 先行節点番号, 後続節点番号

図1のプログラムを入力とした束縛評価グラフ生成ツールの出力例を図3に示す。

```

1 Dec L. 2 D= {a}, U= {5}
2 Dec L. 3 D= {b}, U= {a}
3 MR L. 4 p_in= {x}, R= {b}
4 Dec L. 4 D= {c}, U= {fn x => b}
5 MR L. 5 M= {0}, R= {7}
6 ExS L. 6 F= {c}, p_out= {y}
A1: S-in 5, 6
A2: EnC 6, 3
7 MR L. 6 p_in= {y}, R= {c y}
A3: EnD 3, 7
A4: S-out 6, 7
A5: Ele 5, 7
8 Dec L. 6 D= {d}, U= {fn 0 => 7 | y => c y}
9 ExS L. 7 F= {d}, p_out= {3}
A6: S-in 8, 9
A7: EnC 9, 5
10 Dec L. 7 D= {e}, U= {d 3}
A8: EnD 5, 10
A8: EnD 7, 10
A9: S-out 9, 10
A10: Ele 8, 10
A11: Ele 4, 8
A12: Ele 2, 4
A13: Ele 1, 2

```

図3: 束縛評価グラフ生成ツールの出力

束縛評価グラフ生成ツールは、節点と弧を Standard ML の構文規則にしたがって、以下のようにしてプログラムから束縛評価グラフを生成する。

- i) プログラム要素のうち、宣言、照合規則、関数適用の部分式(式集合)を、節点として出力する。
- ii) 同じ深度の宣言節点を要素順序弧で連結する。
- iii) 関数適用の式の同じ深度の先行節点から式集合節点に構造入口順序弧で連結する。
- iv) 関数適用の式に対応する節点に式集合から構造出口順序弧で連結する。

関数型プログラムの基本要素は、式およびパターンである。そのため、宣言、照合規則、式集合も、これら

から構成されている。具体的には、宣言および照合規則は、式とパタンの組から構成される。式集合は、いくつかの式から構成される。実際に節点および順序弧を生成する際の手順は、以下の通りである。まず、プログラムから式とパターンとを抽出する。続いて文法にしたがって、宣言、照合規則、式集合のどれかの節点を生成する。さらに、これらの節点間の関係も文法にしたがって、順序弧を生成する。

4. 応用

このようにして得られた束縛評価グラフは、関数型プログラムの抽象的表現として、デバッグ、テストをはじめとするプログラム開発活動に応用することができる。具体的には、デバッガ、スライサ、テストケース生成ツール、複雑さ計測ツールといった各種のソフトウェア開発支援ツールへの共通の入力として用いることができる。つまり、各種のツールの入力や出力を束縛評価グラフかその部分グラフにすることによって、どんな関数型言語で書かれたプログラムでもいったん束縛評価グラフを生成することができれば、それら各種ツールをそのまま利用できることになる。

デバッガやスライサで束縛評価グラフを用いると、ある変数に関する部分プログラム(スライス)の計算が可能である。束縛評価グラフでは、関数への入力は必ず照合規則のパターンとして表現されるので、この性質を利用してテストケース生成ツールを作成することができる。

また、これらのツールの入力だけでなく出力も束縛評価グラフあるいはその部分グラフに定めて、共通の書式でやりとりすれば、利用の幅が大きく広がる。

参考文献

- [1] K. van den Berg, "Software Measurement and Functional Programming," Ph.D thesis, University of Twente Enschede, 1995.
- [2] 合田 和正, 程 京徳, 牛島 和夫, "束縛評価グラフとその応用," ソフトウェア・シンポジウム'97, pp. 222-230, 1997.
- [3] R. Milner, M. Tofte, and R. Harper, "The Definition of Standard ML," The MIT Press, 1990.
- [4] R. Milner, M. Tofte, "Commentary on Standard ML," The MIT Press, 1991.