

手続き間並列化コンパイラ WPP の試作

1 E - 2

— 定数伝播とクローニングの評価 —*

佐藤 茂久† 蓮見 勝久§ 飯塚 孝好† 菊池 純男†

†新情報処理開発機構 §日立ソフトウェアエンジニアリング (株) †(株) 日立製作所

1 はじめに

我々は、共有メモリマルチプロセッサ向けの手続き間並列化コンパイラ WPP (Whole Program Parallelizer) を開発中である [1]。出来るだけ粒度の大きな並列性を検出するために、手続き間並列化を行うことを特徴とする。ここでは手続き間スカラデータフロー解析の一部として行う手続き間定数伝播と、その過程で行う手続きのクローニングについて述べる。

2 手続き間定数伝播とクローニング

WPP の手続き間定数伝播は、各手続きの入口での仮引数・外部変数の値と、値を返す手続きの呼び出し毎の返却値を解析する。さらに、手続き入口での変数の値に応じて手続きのクローニング [2] も行う。

手続き間定数伝播の目的はできるだけ多くの変数の値を確定させることであるが、並列化コンパイラでは特に、配列の寸法・添字、ループの上下限・増分・反復回数、条件分岐の条件式などに使用されている変数の値を確定して、より効果的な並列化を行いやすくすることが重要である。クローニングも、単に定数を増やすのではなく、並列化を促進する場合に行う。

手続き間定数伝播は以下のように実施する。まずコールグラフ上のボトムアップ伝播によって、各手続きの変数の値への作用を要約する。次に、コールグラフ上のトップダウン伝播によって、各手続きの入口と出口での変数の値を求めていく。トップダウン伝播の過程でコールサイトによって値の異なる変数のある場合、手続きのクローニングを行い、その値毎に異なる手続きを呼び出すようにプログラムを変換することも出来る。

図 1(a) のプログラムを例に説明する。

まずボトムアップ伝播では、手続きの実行前の変数の値から実行後の値を求める関数を求める。このような関数はジャンプ関数と呼ばれる [3]。例では手続き SUB2 の作用は $f_{SUB2}(M) := J$ という関数に要約される。この関数は、SUB2 の実行後の M の値は、SUB2 の実行前の J の値に等しいことを表す。

<pre> SUBROUTINE SUB1() COMMON //I,J J = 1 CALL SUB2(K) J = 2 CALL SUB2(L) I = J + K + L END SUBROUTINE SUB2(M) COMMON //I,J M = J END </pre>	<pre> SUBROUTINE SUB1() COMMON //I,J J = 1 CALL SUB2_1(K) J = 2 CALL SUB2_2(L) I = 2 + 1 + 2 END SUBROUTINE SUB2_1(M) M = 1 END SUBROUTINE SUB2_2(M) M = 2 END </pre>
--	---

(a) 最適化前

(b) 最適化後

図 1: FORTRAN 77 プログラムの例

次にトップダウン伝播では、SUB1 の各コールサイトについてその時点での変数の値を呼び出し先に伝播する。第一の SUB2 の呼び出しでは、変数 J の値が 1 であることを SUB2 に伝播する。呼び出し後の変数の値は SUB2 のジャンプ関数 f_{SUB2} と呼び出し前の変数の値から求められる。まず、J の値が 1 とわかっているため、 $f_{SUB2}(M) := 1$ となり、仮引数 M に 1 が代入されることがわかる。このことを呼び出し元に伝播するとき、仮引数 M を対応する実引数 K に置き換えて、第一の呼び出し後に K が 1 であることが検出される。また、J の値が変更されないため、第二の SUB2 の呼び出しの前では J の値が 1 であることもわかる。同様にして、第二の呼び出しの後では L, J の値が共に 2 であることが検出される。

次に SUB2 の処理を行う際には、SUB1 からの二つの呼び出しで仮引数 M の値が 1 と 2 であることが検出されている。そこで M を 1 と 2 に特化したクローン手続きを作成し、SUB1 からの呼び出し先をそれぞれのクローン手続きに変更する。図 1(b) に手続き間定数伝播とクローニングの実施後のコードを示す。

手続き間定数伝播の精度は、伝播対象とする変数の種類と、ジャンプ関数で表現できる代入文の範囲に依存する。現在 WPP では、整数型・実数型 (複素数も含む) のスカラ変数と、添字が定数である配列要素について定数伝播を行うことが出来る。また、ジャンプ関数は定義文の右辺が単一の変数または定数の場合実装済みであり、より複雑な式も今後対応していく。

*Prototyping of Interprocedural Parallelizing Compiler WPP — Evaluation of Constant Propagation with Procedure Cloning — by Shigehisa Satoh (RWCP), Katsuhisa Hasumi (Hitachi Software Engineering Co., Ltd.), Takayoshi Iitsuka (Hitachi, Ltd.), Sumio Kikuchi (RWCP).

表 1: SPECfp95 で検出された手続き間定数

プログラム	クローニング	手続き	仮引数 (定数 / 総数)		外部変数 (定数)		定数の用途			
			整数型	実数型	整数型	実数型	配列	ループ	分岐	その他
swim	無 / 有	6	0/0	0/0	0	0	0	0	0	0
su2cor	無	25	6/29	0/6	0	0	3	1	3	0
	有	55	58/90	0/42	0	0	46	10	55	0
hydro2d	無	40	7/12	0/2	0	5	9	2	10	3
	有	44	14/19	0/4	1	5	16	6	16	3
mgrid	無	13	2/27	0/5	0	0	2	1	0	0
	有	14	5/29	0/5	0	0	5	2	2	0
applu	無 / 有	16	8/27	0/3	3	62	11	1	0	78
turb3d	無	23	15/45	0/11	28	0	29	13	7	7
	有	57	105/199	0/16	28	0	108	60	74	18
apsi	無	87	2/287	2/237	0	0	2	1	1	1
	有	91	7/322	11/267	0	0	3	1	10	5
fpppp	無	13	6/40	0/4	0	4	2	0	5	7
	有	15	8/43	0/6	0	4	4	2	13	13
wave5	無	88	7/102	0/35	16	5	18	8	23	3
	有	116	14/124	1/40	22	14	42	24	76	18

3 評価

SPECfp95 ベンチマークの FORTRAN 77 プログラムで、手続きが複数あるものについての手続き間定数伝播とクローニングの結果を表 1 に示す。

ここではクローニングを行った場合と行わなかった場合についてそれぞれ、手続き数、各手続きの入口で定数となる変数の数、定数となった変数のその手続き内での用途を示す。手続き数はプログラム中で実行され得る手続きの総数で、クローニングを行う場合はクローン手続きを含む。ただし、二次入口のある手続きは入口毎に別の手続きとして扱う。定数の数は、仮引数と外部変数、整数と実数(複素数も含む)に分けて数えた。仮引数の欄では '/' の右側が仮引数の総数、左側がそのうちの定数となる変数の数である。外部変数(COMMON)の欄は定数となるものの数を手続き毎に重複して数えたときの総数である。これらはいずれも手続き中で露出使用されるもののみ数えている。定数の用途は次のように分類し、重複して数えた。

- 配列: 配列の寸法・上下限、配列の添字等で使用。
- ループ: ループの上下限・増分・反復回数等で使用。
- 分岐: ループ以外の条件分岐の条件式で使用。
- その他: 上記のいずれの使用もない。

なお、定数となる変数の値が手続き内で他の変数に代入される場合は、代入先の変数の使用法も含めた。

SPECfp95 では swim 以外のプログラムで手続き間定数が検出され、クローニングによってより多くの定数が検出された。例えば su2cor と turb3d では、コールサイトによって実引数の値が異なる手続きがあるため、その値毎にクローン手続きが作成される。これらの引数は配列の添字や条件分岐の条件式で使用さ

れるため、その値によって手続き内での配列の参照領域が異なり、並列化への影響が大きい。なお、実数型の変数は配列やループでは使用されないため、クローニングによって増加した定数の多くは整数型である。

さらに、変数の用途を考慮せずに定数を最大限検出できるようにクローニングを行うと、applu, turb3d, fpppp, wave5 でそれぞれ 3, 8, 3, 13 個のクローン手続きが作成される。しかしこれらのクローニングはいずれもループ並列化には貢献しない。また、用途を考慮したクローニングでも、その全てが並列化に貢献するわけではない。

今回は変数の用途から並列化への効果の有無を予測して手続き数の増大に対処したが、それでも turb3d のように手続き数が倍以上に増えてしまうこともある。そこで、今後は並列化を実施する際に並列化に影響がないことがわかったクローニングを取り消せるようにする予定である。

4 まとめ

WPP における手続き間定数伝播とその過程で行う手続きのクローニングの概要と、SPECfp95 ベンチマークにおける効果について報告した。今後は解析精度の向上やより効果的なクローニングを行っていく。

参考文献

- [1] 飯塚他, 手続き間並列化コンパイラ WPP の試作 — 現状と今後の課題 —, 情報処理学会 第 56 回全国大会, 1998.
- [2] K.D.Cooper, M.W.Hall, and K.Kennedy, A Methodology for Procedure Cloning, in *Computer Languages*, Vol. 19, No. 2, 1993.
- [3] D.Grove and L.Torczon, Interprocedural Constant Propagation: A Study of Jump Function Implementation, in *Proceedings of PLDI '93*, June 1993.