

並列推論マシン PIM/i の命令セットの評価

加藤研児、佐藤正俊

kato@okilab.oki.co.jp

沖電気工業株式会社

1 はじめに

我々は、第五世代コンピュータ・プロジェクトの一環として並列推論マシン PIM/i [1] の研究開発を行っている。PIM/i は KL1 で記述された知識処理プログラムを高速に処理する高級言語マシンであり、KL1 とは並列実行を念頭において開発された記号処理向きの論理型言語である。高級言語マシンでは対象言語のセマンティックを直接アーキテクチャが支援する場合もあるが、PIM/i では実行時の最適化の可能性を多く含むような RISC アーキテクチャを設計し、コンパイラによる最適化を重視する方法を取った [2]。

プログラムがどのような機械語命令に展開されるかは、命令セットを評価する上で重要なことである。本稿では KL1 プログラムの実行に際してどのような処理が行われるか、及びどのような PIM/i の命令セットが実行されるかについて考察したので報告する。

2 命令セットと測定方法

PIM/i の命令は 40 ビット固定長であり、1 命令の中では 4 種類の操作（分岐、メモリ、演算、タグ）から最大 3 操作を指定して、複数操作が同時実行できるような構成を持つ。各操作は RISC 思想で設計され、全操作は理想的に 1 サイクルで実行できるように単純な操作に限定してある。また、パイプラインの段数は論理型言語における分岐頻度の高さを考慮して比較的少ない 3 段とし、分岐によるパイプラインの乱れを抑える構成となっている。さらに記号処理を円滑に行うため、データの一部をデータタイプを表すタグとして用い、タグ値による条件分岐なども可能とした。

実行命令の測定は、4 つのベンチマーク（PRIMES, QU, QK, BUP）をレジスタトランスファレベルのシミュレータ [3] を用いて 1 プロセッサで実行させ、その実行履歴ファイルを基にデータを収束することにより行った。また、KL1 の実行方法は Prolog における WAM 方式のように一旦抽象機械語命令である KL1b 命令にコンパイルし、さらに PIM/i の命令まで展開して、実行する方法を用いた [4]。PIM/i 命令への展開に際しては、コードサイズの増大を考慮し、KL1b 展開命令とランタイムライブラリに分けて展開し、KL1b 展開命令を共有メモリに配置し、ランタイムライブラリをローカルメモリに配置した。

*Evaluation of the PIM/i processor's instruction set.
Kenji KATO and Masatoshi SATO
Oki Electric Industry Co., Ltd.

表 1: KL1b 命令別の実行命令内訳 (%) と平均実行命令数

KL1b 命令	wait	r/w	enq	ctl	p/s	get	blt
演算	20	60	33	42	48	4	0
演算 + タグ	0	5	13	3	18	0	0
演算 + メモリ	3	5	33	13	16	0	0
メモリ操作	0	30	7	0	12	0	0
分岐	72	0	7	26	0	0	85
コール	4	0	0	16	3	26	4
引数設定	1	0	0	0	3	70	11
その他	0	0	7	0	0	0	0
実行命令数	3.2	3.0	15.0	5.8	2.0	3.6	1.2

r/w: read/write, enq: エンキュー操作, p/s: put/set,
ctl(control): 実行制御, blt(built.in): 組み込み述語

3 考察

3.1 KL1b 命令と PIM/i 命令

KL1 プログラムの実行における KL1b 命令レベルの処理が、PIM/i プロセッサにおいてどのように実行されたかを考察する。表 1 に、KL1b 命令と PIM/i 命令の対応を示した。これは 4 つのベンチマークでの測定の平均値である。なお、各ベンチマークでは用いるデータ構造や算術演算の種類により若干の差違はあったものの、ほぼ同じ様な分布を示した。

wait 系命令はデータタイプチェックが主な処理で、分岐命令の割合が多く含まれていた。get 系命令はユニフィケーションを行うサブルーチン呼び出しのみであるので引数設定の割合が高かった。read/write 系及び put/set 系の命令ではメモリアクセスが主な処理であり、サブルーチン呼び出しがほとんどなかった。なお、KL1b 命令は一部の命令を除くと 3 命令程度で実行できた。

3.2 KL1b 命令機能の実行率

KL1 プログラムの実行における処理の内容について考察する。図 1 は、KL1b 命令の機能に相当する実行命令数を、全命令数に対する割合で示している。(a) は、KL1b 命令別の実行命令数の割合を示し、(b) は、ランタイムライブラリ別の実行命令数の割合を示している。ライブラリの分類は、実行管理（コンテキスト切替え及びメモリ管理、PE 間通信等の処理）及びデレファレンス、ユニフィケーション、ゴールキュー操作、算術演算、GC 等で行っ

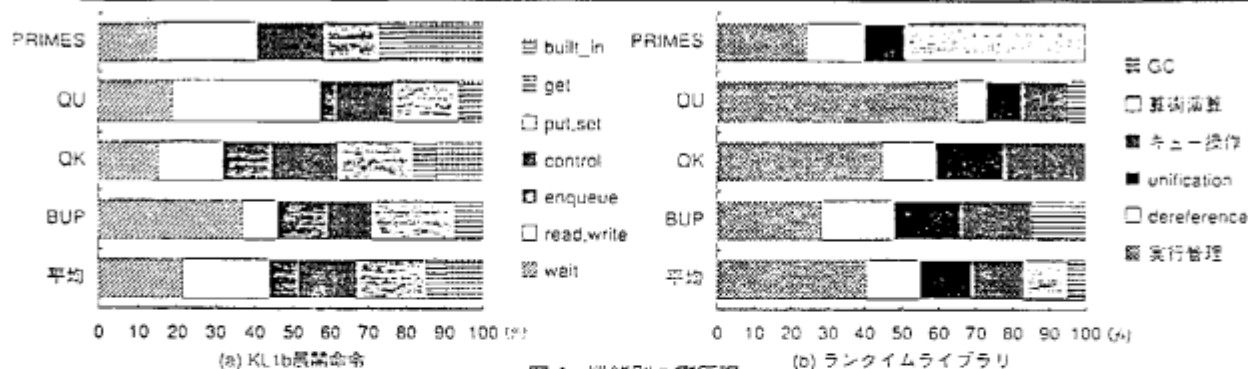


図 1: 機能別の実行率

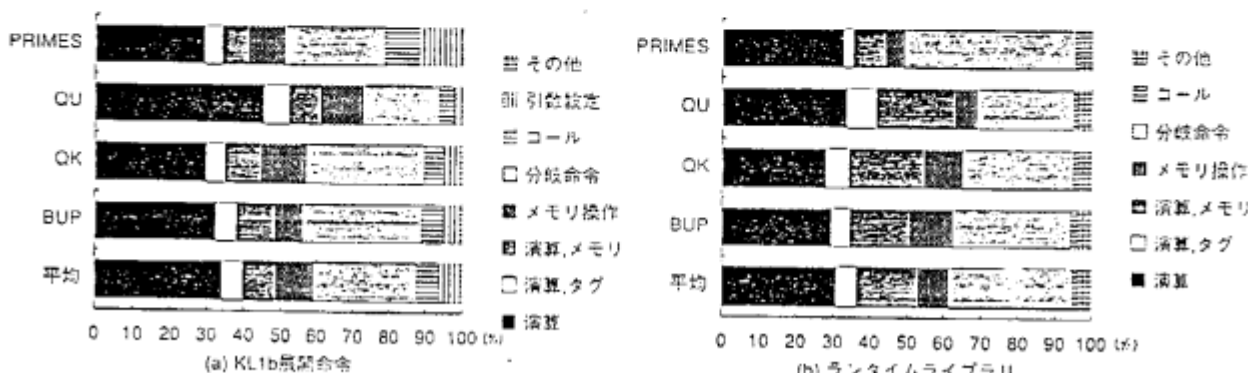


図 2: PIM/i 命令の実行率

た。なお、KL1b 展開命令での実行命令数とランタイムライブラリでの実行命令数の比率は、どのベンチマークでも 4:6 ~ 5:4 程度であった。

これによると、KL1b 命令レベルではメモリアクセス (read, write, put, set) やデータチェック (wait) などの、サブルーチンコールの無い、軽い KL1b 命令の実行が半数近くを占めた。一方ライブラリでは、実行管理やゴールキュー操作が主な処理であり、KL1 言語特有の機能であるデレファレンス及びユニフィケーションの処理も実行の 3 割程度を占めた。

3.3 PIM/i 命令の実行率

前節で考察した KL1 の処理が、PIM/i プロセッサにおいてどのように実行されるかを考察する。図 2 において実行命令の種類別の割合を、KL1b 展開命令の場合 (a) 及びランタイムライブラリの場合 (b) に分けて示す。

KL1b 展開命令では、演算の割合が全体の 3 割以上を占め、他操作との同時実行の場合を含めると 5 割程度が演算を伴う操作であった。また分岐命令は 3 割程度であり、サブルーチンコールと引数設定の割合は 1 割程度であった。

ランタイムライブラリの場合では、KL1b 展開命令に比較すると、演算が相対的に減少し、分岐命令やメモリ操作の割合が増加した。また、サブルーチン呼び出しと引数設定の割合が 5% 程度に減少した。なお、PRIME で分岐命令が格段に多いのは、PIM/i に乗除算命令がないためである。

4 おわりに

PIM/i の命令セットを抽象命令レベルとライブラリレベルに対応して考察した。4 つのベンチマークの実行結果によると、抽象命令の多くは簡単な操作であり、3 命令程度の PIM/i 命令に展開され、これら抽象命令の処理はメモリアクセスが多かった。また、並行実行機能である同期待ち (wait) は、主に分岐命令として実現され、言語特有の推論機能であるデレファレンスやユニフィケーションは、ランタイムライブラリの 3 割程度であり全体から見ると、少なかった。全体的には、分岐頻度が高く分岐命令の設計が重要であることが確認された。今後はさらに様々な見方から命令セットの評価をしていく予定である。

日頃から助言をいただいている ICOT 第 1 研究室の皆様、および沖電気の PIM 担当諸氏に感謝します。

参考文献

- [1] 大原他: 並列推論マシン PIM/i の概要、情報処理学会第 40 回全国大会、1990
- [2] 武田他: 並列推論マシン PIM/i の要素プロセッサのアーキテクチャ、情報処理学会第 40 回全国大会、1990
- [3] 吉田他: 並列推論マシン PIM/i の開発支援環境—シミュレーター、情報処理学会第 41 回全国大会、1990
- [4] 佐藤他: 並列推論マシン PIM/i の開発支援環境—翻訳系—、情報処理学会第 41 回全国大会予稿集、1990