

Lisp マシンにおけるキャッシュメモリの有効性

1P-2

川村浩正

渡辺和文

NTT ヒューマンインタフェース研究所

1. はじめに

キャッシュメモリの有効性は、メモリアクセスの局所性に密接に結びついている。Lisp マシンでは、命令/データが2進木リスト構造で表現され、さらにメモリ管理が動的であるため、メモリアクセスの局所性が期待できない。しかし、Lisp マシンの高速化のためには、メモリアクセス時間を実効的に短縮するキャッシュメモリが重要となる。そこで、Lisp マシン ELIS[1] を用いてアドレスをトレースし、キャッシュメモリの有効性を測定したので報告する [2]。

2. 測定方法

測定は、2台の ELIS を使い、1台を被測定用、もう1台をアドレスデータ格納用とした。実プログラム走行中の被測定用 ELIS から、アドレス信号、メモリ制御信号(読み書き制御等)を取り出し、インタフェースを介してデータ格納用 ELIS のメモリに書き込む。データ数は、格納用に使用した ELIS のメモリ容量の関係で約 250 万まで可能である。測定によって得られたデータを用い、そのアクセス分布の解析、およびキャッシュのシミュレーションを行なった。キャッシュのマッピング方式は、ダイレクトマッピングとし、書き込み方式は、ライトスルーとした。書き込みは全てミスヒットとしてカウントした。

使用したベンチマークプログラムは以下の2つである。

(1) TPU

約 400 行の規模を持つ定理証明プログラム [3]。

(2) Dderiv

リスト処理を行なう再帰定義されたプログラム [4]。トレースメモリがオーバーフローしないように、本来の実行ループの 1000 分の 1 を測定対象とした。

3. 結果

3.1 メモリアクセスの領域別分類

表 1 は TPU3 のトレースデータから求めたメモリ

表 1. TPU3 におけるアクセスの割合

領域	インタプリタ	コンパイル後
シンボル	3%	2%
セル	85%	60%
ベクタ	12%	30%
命令	0%	8%
全アクセス数	約 39 万回	約 12 万回

アクセスの割合を示す。

最もアクセス回数が多いのは、セル領域で、その次にベクタ領域、シンボル領域が続く。セル領域は、2進木リスト構造データを、ベクタ領域は、関数に付随する情報を、シンボル領域は、シンボルをそれぞれ格納するために使用される。セル領域におけるアクセス割合は他の領域と比較して大きいので、セル領域での局所性がキャッシュメモリの性能に与える影響は大きい。

3.2 キャッシュ容量、ブロックサイズの影響

図 1 は、ブロックサイズを 8cell(8byte/cell) に固定したときの、キャッシュ容量とヒット率の関係である。プログラムにかかわらず、インタプリタで実行した場合もコンパイル後に実行した場合もキャッシュ容量が 4kcell を越えると、ヒット率は、ほぼ飽和してしまふことがわかる。

図 2 は、キャッシュ容量を 4kcell に固定したときのブロックサイズとヒット率の関係である。ブロックサイズが、8~32cell の範囲で最大値を持つ。

3.3 コンパイルによる影響

表 1 から、コンパイル後に実行した場合の方が、セル領域へのアクセス割合が減少することがわかる。この理由は、次の通りである。インタプリタで実行するときは、セル領域上の 2進木リスト構造を持つ S 式を取り出しながら解釈実行していく。そのため、関数本体とデータへのアクセスには、共にリストたどりが必要となる。しかし、コンパイルすることで関数本体はバイトコード列に展開され、命令領域に移る。した

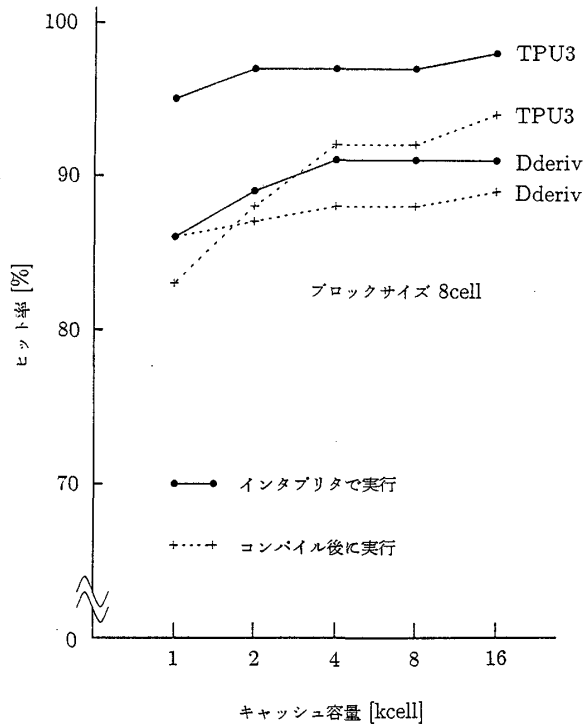


図1 キャッシュ容量とヒット率の関係

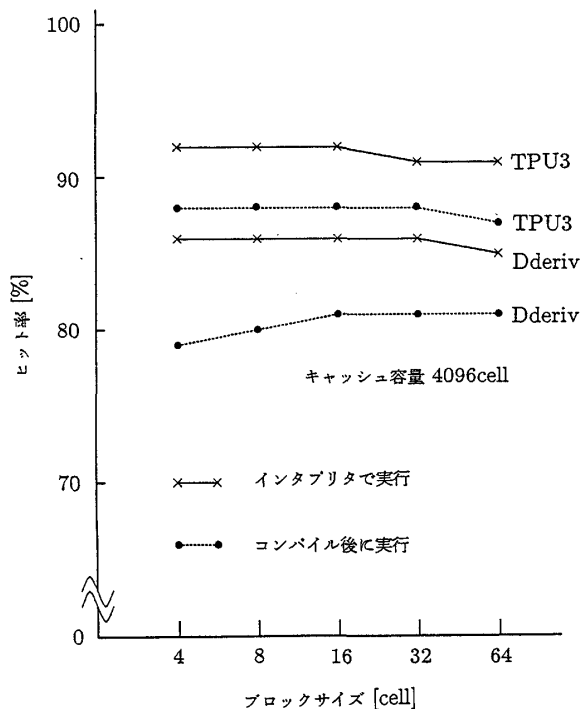


図2 ブロックサイズとヒット率の関係

がって、セル領域のアクセス回数は減少する。

図1、図2から、キャッシュのヒット率が約5%減少することがわかる。これは、ライトスルー方式では、書き込みの割合が増えることが、ヒット率の低下につながるためである。ELISでは、インタプリタで実行した場合、1命令取り出すために1回のメモリアクセスを必要とするが、コンパイル後では、1回のメモリアクセスで最大8個の命令が取り出せる。このため、読み出しのためのアクセス回数が減少する。一方、書き込みアクセスはデータの書き出しであるので、その数は変化しない。したがって、書き込みの割合は増加する。

4. 終わりに

Lisp マシン ELIS を用い、ベンチマークプログラム走行時のアドレスをトレースした。そして、このトレースデータを用い、キャッシュのシミュレーションを行なった。その結果、以下のようなことがわかった。

- (1) キャッシュ容量が4kcell以上になると、ヒット率は飽和する。
- (2) ブロックサイズが8～32cellでヒット率は最大値をもつ。
- (3) インタプリタで実行した場合の方が、コンパイル後に実行したときよりヒット率は高くなる。

今後は、データ格納用メモリの容量を大きくし、大きな応用プログラム(エキスパートシステム等)の実行時や、マルチプロセス時における多くのデータを収集し、他のマッピング法も加えて、Lispシステムにおける最適なキャッシュの方式を決定したい。

最後に研究を進めるにあたって、有益な助言とご議論を頂いた、言語メディア研究部、日比野靖主幹研究員に深く感謝いたします。

参考文献

- [1] 日比野、渡辺、大里: Lisp マシン ELIS の基本設計、情処記号処理研資 12-15、1980
- [2] 川村、渡辺: Lisp マシンにおけるキャッシュメモリの効果測定、信学技報、CPSY88-、1988
- [3] C-L. Chang and R. C-T. Lee., Symbolic Logic and Mechanical Theorem Proving, Academic Press, 1973
- [4] R.P.Gabriel, Performance and Evaluation of Lisp Systems, The MIT Press, 1986