

4B-2

LISPマシンSYNAPSEの

ガーベッジ・コレクション・システムについて

寺村¹ 信介、松井² 祥悟、加藤² 良信、中西² 正和

(株)リコー中央研究所、慶應義塾大学

1. はじめに

SYNAPSEは、セル空間として大容量の共有メモリを持つマルチマイクロプロセッサLISPマシンである。セル空間は現在8メガバイトであるが、これくらいの大きさになるとガーベッジ・コレクション(以下G. C. と略す)をどのように行なうかが大きな問題となる。通常の一括型G. C. ではかなりの時間がかかり、極めて応答性の悪いシステムになる。SYNAPSEのG. C. システムでは並列G. C. を採用し、この問題を解決している。以下ではG. C. のアルゴリズム、ハードウェアの構成およびその評価について述べる。

2. アルゴリズム

基本的なアルゴリズムはKung & Song[1]のアルゴリズムである。このアルゴリズムは、リストたどり法を用いてマーキングを行ない、セル領域全体を走査してガーベッジ・セル(ごみ)を回収するものである。マーキング・ビットとしては各セルごとに2ビット必要である。SYNAPSEでは、このアルゴリズムの実現および高速化のために改良が加えられている。最も大きな違いはプロセッサの台数である。Kung & Songのアルゴリズムではリスト処理をするリスト・プロセッサ(LP)と、ガーベッジ・コレクションを行なうガーベッジ・コレクタ(GC)がそれぞれ一台ずつであったが、SYNAPSEの場合には複数のLP、複数のGCが同時に処理を行なうようになっている。また、実現上問題のあったKung & Songのアルゴリズムにおけ

るdeque(双頭待ち行列)を廃し、ハードウェアを使って解決するなど、構成上様々な工夫がなされている。

3. 構成

LP、GCはいずれもモトローラ社のMC68000を用いている。GCの内の1台はスーパーバイザとして、システム全体を監視するようになっている。SYNAPSEのG. C. システムの概要を図1に示す。図から分るように、共有メモリは4つのブロックに分かれている。セル・ブロックはセルの存在するメモリで、LPはこのブロックのフリー・セルを用いて処理を行なう。ハードウェア・スタック・ブロックは、マーキングの必要が生じたセルを、LPからGCに渡すためのメモリ・ブロックであり、Kung & Songのアルゴリズムにおけるdequeに相当する。ハードウェア・スタックに対するデータのプッシュ・ポップは、固定番地に対するデータ転送命令によって行なわれる。またこの操作は非可分であるため、高速に処理ができる。タグ・ブロックはマーキング用タグメモリのブロックであり、GCはこのブロックを用いてマーキングを行なう。タグ・ブロックはセル・ブロックと同じアドレス空間に存在している。ロングワードアクセスのときはセルブロックを、バイトアクセスのときはタグ・ブロックをアクセスするようになっていて、GCとLPのメモリアクセス時の衝突を減らしている。コントロール・ブロックはフリー・リストのヘッダ等、非可分にアクセスすることが必要な領域のためのメモリである。各ブロックにはそれぞれプロセッサの台数分のバス・アービタがついている。バス・アービタとは各プロセッサからのメモリ要求信号を仲裁するためのハードウェアであり、これにより

Garbage Collection System for Lisp Machine SYNAPSE

Shinsuke TERAMURA, Shogo MATSUI, Yoshinobu KATO, Masakazu NAKANISHI

1. RICOH, Ltd. 2. KEIO, Univ.

異なるブロックのアクセスを別々のプロセッサが同時に行なうことができる。

このような構成をとることにより、LPのG. C. に関するオーバーヘッドを大きく減らし、G. C. を高速に行なうことができる。

4. 結果

実験結果では、典型的なプログラムに対して、GC、LP各1台ずつの場合、一括型のものに比べ、処理時間が80%程度になる。GCを2台にすると1台の場合に比べ、更にまた20%短縮された。

プロセッサの台数が増えてくるとバスの競合が大きくなり、処理時間が長くなってしまふ。この処理時間の伸びは、プロセッサの台数 n とメモリアクセスの割合 x の関数になる [2]。メモリアクセスの割合とは、全処理時間を1としたときのメモリアクセスに要する時間である。 $x=0.2$ の場合、 $n=10$ で処理時間が約2倍、 $n=15$ で約3倍になる。

上述の実験ではLP1台、GC2台の $n=3$ の場合であり、 $x=0.3$ とすれば処理時間の伸びは約1.2倍である。したがってバスの競合を無視した場合、一括型GCからGCを1台にしたとき、およびGCが1台から2台にしたときの処理時間の短縮率は、いずれも75%前後であると考えられる。SYNAPSEでは $x=0.3$ 、GCの台数 $n=4$ を想定しているが、この場合処理時間の伸びは約1.4倍程度となる。

5. まとめ

実験結果を見ると、GCの台数を増やすことで、かなり高速化が期待できる。もちろんこれは繁雑にG. C. が起こるような場合であり、まったくG. C. が起こらない場合は逆にオーバーヘッドの分だけ遅くなる。これはLISPプログラムまたは処理形のアルゴリズムと密接に関係する問題となる。

即時型のG. C. に比べると、オーバーヘッドの小さい分、処理時間は短くなる。しかし、LPのセル消費速度がある臨界点を越えてしまうとごみの回収が間に合わなくなり、LPが待たされて処理時間が延びてしまう。この臨界点はセル空間が大きくなればなるほど大きくなる。したがって、セル空間の大きなシステムにおいては並列G. C. の方が有利である。

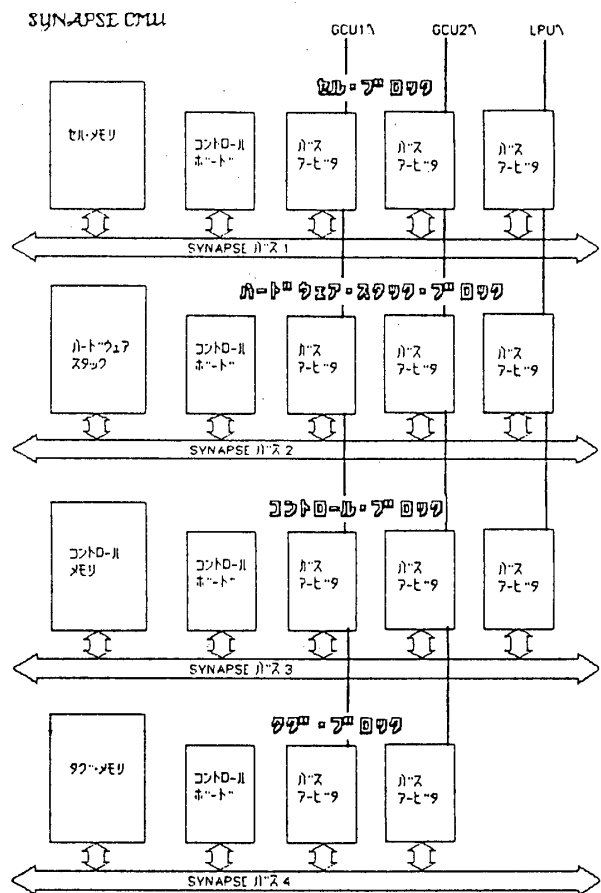


図1 共有メモリの構成

参考文献

[1] Kung, H. T., and Song, S. W. : An efficient parallel garbage collection system and its correctness proof, Dep. Computer Sci., Carnegie-Mellon Univ., Pittsburgh (1977)
 [2] 加藤、他 : SYNAPSEの機能と動作について、情報処理学会記号処理研究会資料30-3, 1984