並列推論マシンＰＩＭ

－共有メモリ構造クラスタにおけるユニフィケーション－

清水　肇，佐藤　正俊
（財）新世代コンピュータ技術開発機構

1．はじめに

第５世代コンピュータの研究開発プロジェクトでは，共有メモリを用いた密結合マルチプロセッサからなるクラスタを疎結合した構成の並列推論マシンＰＩＭの研究を進めている[1]．

ＰＩＭのＫＬ１(核言語)[2] の処理方式はＰＩＭの構成に従い密結合での処理方式と疎結合での処理方式に別けて検討を進めている．本稿では，前者のうちでも共有メモリのロック機構を用いたユニフィケーションにおける，デッドロック回避と，ロック期間と回数を減少させるための手法について述べる．

なお，ＰＩＭの機械語はＫＬ１－Ｂ抽象命令[3] に基づいている．

2．ユニフィケーションとロック

2.1 passive unification命令 (wait_xxx,read_xxx)

受動部におけるunificationを行う命令群である．値が未定義であっても原因となった変数をスタックに確保するだけで，書込みをしない（実際のフックの処理は後述のsuspend命令で行う）．従ってロックの必要がない．

2.2 引数準備命令 (put_xxx,set_xxx,write_xxx)

ボディゴールをforkする時の引数を準備するときの命令であり，変数／定数の受け渡しや，新たな変数／構造体を作成する．後者の場合でも，新たにheapに書き込みはするが，ＰＥ間で共有されていないからロックの必要はない．

2.3 active unification命令 (get_xxx,unify_xxx)

能動部におけるunification を行う命令群であり，その結果，値を書き込みや，サスペンドしていたゴールの再開（後述）のためにロックが必要である．

```
arg([],Y,Z):- Y=Z.
```

上の例の様に，変数同志のユニファイをする時，２ヶ所にロックをかけなければならないが，同時に，あるＰＥではＹ，Ｚの順に，他のＰＥではＺ，Ｙの順にロックすると，デッドロックに落ちいる．これを避けるためアドレスを比較しながらロックする．また，変数から変数へのポインタを単一の方向にはる．

構造体と変数のユニフィケーションは，変数の値が定まっている場合(read mode) と未定義の場合(write mode)で処理が異なるが，後者の場合，ロックが複数の命令にまたがってしまう(ex.1)．何故ならば，get_listでunlockすると他のＰＥの受動部のユニフィケーションで要素の値を読む可能性があり正しく動作しないからである．

```
ex.1  ap([A|X],Y,Z):- Z=[A|Z1], ap(X,Y,Z1).
        :
      get_list(A3)       ← lock          [_|Z1]
      unify_value(A5)    ← write
      unify_var(A3)      ← write, unlock   ↓
      get_list(A5)       ← lock          [A]
      unify_value(A4)    ← write
      unify_nil          ← write, unlock
        :
```

一方，ＫＬ１でかかれたプログラムをみると，現在評価したプログラムはwrite modeだけである．またプログラムのスタイルから考えてもread mode はfailのもとであるから，今後もwrite modeが多いと予想している．そこで，ロック期間を短縮するために，引数準備命令でローカルに構造体を作り，まとめてユニファイする方式(write mode 決め打ち方式 ex.2)を検討した．引数準備命令は構造体を内側から作るのに対して，従来のactive unificationは，外から評価していかなければならない(ex.1)．従って，この例のように構造体の要素がまた構造体であるような場合，ロック回数も少なくて済む．

```
ex.2  ap([A|X],Y,Z):- Z=[A|Z1], ap(X,Y,Z1).
        :
      put_list(A5)                       [A]
      write_value(A4)    ← write
      write_var(A4)      ← write          ↓
      put_list(A4)                       [_|Z1]
      write_value(A5)    ← write
      write_var(A6)      ← write
      get_value(A4,A3)   ← lock, unlock
      put_value(A6,A3)
        :
```

この方式の欠点はコード量が増えることと，read mode であった時でも構造体を作ってしまうオバーヘッド，及び，容易にはその構造体をダイナミックに回収出来ない点にある．

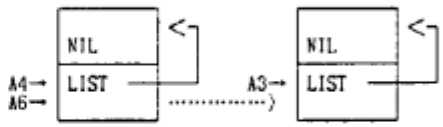

図１

図１のように，get_value(A4,A3)でread mode のときに，メモリを回収するためには，A6の内容を変える必要がある．

Parallel Inference Machine: ＰＩＭ
－Unification Algorithm for PIM Cluster－
Hajime SHIMIZU, Masatoshi SATO
ICOT　　ICOT

2.4 中断／再開

中断時には，逐次エミュレータ[4]と同じくサスペンド（サスペンジョン）レコード，ＯＲ待ち（サスペンジョン）フラグを経てゴールレコードをフックする．また，２つ以上の変数にフックしているゴールの再開は，ＯＲ待ちフラグを早くＯＮできた方の実行を行う（当然ロックを伴う）．ＰＩＭではゴールを各ＰＥ毎に管理するため，中断したＰＥと再開したＰＥが異なるときは，後者から前者へ再開要求のメッセージを送る．

中断時には，値が定まっている可能性があるため中断変数をロックしチェックする． Unboundならばサスペンドレコードをつなぐ． Boundedの時，ＯＲ待ちフラグをＯＮできたならば，このゴールをすぐさま再開できる．ただし，ＯＮできなかった場合，他のＰＥがすでに再開要求のメッセージを送る処理を開始しており，それを中止させることは困難であるため，再開要求のメッセージが来た時点で，ゴールレコードをレディーキューに繋ぐ．

2.5 ロック有りのderefアルゴリズム

ＫＬＩでは値が定まっている(Bounded) 変数に書込みをしない．従って値が定まっていない(Unbound) 変数にのみロックをすればよい．よって，変数をderef し，ロックをかけるアルゴリズムには，次の２つの方式が考えられる．

方式１：まずロックをかけ読み出し，Bounded ならば，アンロックする

方式２：まず読み出し，Unboundならば，ロックして読み出す

変数が Boundedであれば後者が， Unboundであれば逆に前者が１回メモリアクセスが少ない．

ロックの回数は，常に後者の方が少ない．

3．命令の動的特性

3.1 命令の出現頻度

表１．命令の出現頻度

ＰＥ８台	8-Queens	ＢＵＰ
passive unify	132,827 (24%)	145,439 (27%)
active unify	30,833 (6%)	48,569 (9%)
引数準備命令	132,108 (24%)	169,274 (32%)
制御命令	107,775 (19%)	164,791 (31%)
(try_me_else)	57,881 (10%)	113,886 (21%)
(suspend)	0 (0%)	1,609 (0%)
組込み命令	156,618 (28%)	3,977 (1%)
合計	560,161	532,050

表１に8-QueensとＢＵＰの各命令系の出現頻度を示す．lockの必要な命令は，たかだか10% で衝突は少ないと予想される．また，try_me_else命令の頻度が高く（特にＢＵＰは辞書検索に多用する），今後，インデキシング命令による減少をはかる．

3.2 write mode 決め打ち方式によるコード量

表２にwrite mode 決め打ちによるコード量を示す．これからコード量の増加は（さらにコードをオプティマイズしても）たかだか数％に過ぎない．従って，ロック期間の短縮等の要因を考え，今後この方式にしたい．

表２．write決め打ちによるコード量

ＰＥ８台	8-Queens	ＢＵＰ
リダクション数	38,878	35,717
get-unify方式	560,161(14.4)	532,050(14.9)
決め打ち方式	573,341(14.7)	545,108(15.3)
比	1.02	1.02

3.3 ロック有りderefの方式による差異

表3.にロック有り derefの方式による差異を示す．これらプログラムでは，変数に値が定まっている事は少なく，Deref の平均長も少ないため，Boundedである回数は少ない．従って，アクセスの回数は方式２の方が多い．しかしキャシュのシミュレータ[5] によれば必要なバス・サイクルには有意差がない．これは，不必要なロックによるキャシュのインバリデイトがないからと思われる．

表3.ロック有りderefの方式による差異

ＰＥ８台		8-Queens	ＢＵＰ
Bounded の回数		4,314	1,844
Unbound の回数		22,922	36,132
ロック有りDeref の平均長		1.10	1.01
方式１	アクセス回数	31,550	39,820
	ロック回数	27,236	37,976
方式２	アクセス回数	50,158	74,108
	ロック回数	22,922	36,132

4．おわりに

共有メモリを用いたのクラスタ内におけるユニフィケーションについて報告した．現在この方式にそった逐次マシン上のクラスタ内処理系により評価を進めている．さらに，市販の密結合並列マシン(Balance 21000) による処理系を開発中である．今後さらに評価プログラム数を増やし，各種データを収集する予定である．

（参考文献）

[1] 後藤他：" 並列推論マシンＰＩＭ"，情報処理学会第33回全国大会予稿集 3B-5～3B-7

[2] K.Ueda：" GUARDED HORN CLAUSE" , LPC '85, pp.255 1986-6, JAPAN

[3] 木村他：" - ＫＬ１の抽象命令仕様とコンパイラ -"，本全国大会2P-1

[4] 久門他：" - ＫＬ１の逐次版エミュレータ -"，本全国大会2P-2

[5] 松本他：" - 並列キャシュとロック機構 -"，本全国大会2P-6