知識ベースマシン *Hu-X*（3）
−ハードウェア実験機によるシミュレーション−

5E−6

伊藤文英　　酒井　浩，柴山茂樹
（財）新世代コンピュータ技術開発機構　研究所　　（株）東芝　総合研究所

1．はじめに

知識ベースマシンHu-Xの演算機能の1つに，Prologのファクトとルールを格納した項関係に対して入力演算を実行し，与えられたゴールの全解を探索する機能[1]がある．Hu-Xでは，演算を複数の要素プロセッサ（PE）で並列に実行して，システム性能の向上を図っている．入力演算による全解探索に対しては，入力データ量に基づく演算処理の分割方式を提案し，ソフトウェアシミュレーションによりそれを評価した．[2]

ソフトウェアシミュレーションでは，分割された演算処理の実行時間を，処理ハードウェアの動作シミュレーションにより正確に求めた．しかし，演算制御のための時間や並列実行でのPE間の競合は，制御ソフトウェアの処理方式や，それが試作されるハードウェアの特性により大きく異なるため，考慮しなかった．

そこで，Hu-Xの試作ハードウェアと制御ソフトウェア[3]上で演算処理のシミュレーションを行い，並列実行における試作システムのハードウェア構成の妥当性を評価した．本稿では，その概要と結果について述べる．

2．試作システムの構成と処理方式

試作システムのハードウェア構成を図1に示す．制御プロセッサ（SCP）は，演算を分割してPEにコマンドを送信する．コマンドを受信したPEは，マルチポートページメモリ（HPPM）[4]上の項関係データを入力して処理し，結果をHPPM上に出力し，レスポンスをSCPに送信する．SCPは，PEからのレスポンスの受信状況に従い，次の演算を分割する．共有メモリ（SH）は，コマンドとレスポンスの送受信と，HPPMの資源管理に使われる．HPPMでは原理的にアクセス競合がないが，SHでは発生する．

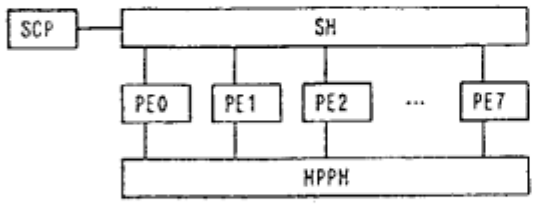

図1　試作システムのハードウェア構成

開発中のHu-X制御ソフトウェアでは，演算の分割をいずれかのPEで行うが，ソフトウェアシミュレーションとの比較のため，試作システム上のシミュレーションではSCPで行った．また，PEの演算処理機能は開発中のため，実際の演算処理は行わず，ストリーム処理方式の単一化エンジン（UE）[5]の動作をシミュレーションした．

3．演算処理の分割方式

OR並列の入力演算による全解探索では，項関係を次のように演算する．[1]

① ファクトとルールを格納した項関係（PR）をゴールで単一化制約し，中間結果の項関係（TR）とする
② TRのタプル数が0になるまで，次の③と④を繰返す
③ TRを単一化制約し，それに含まれる解を取出す
④ PRとTRを単一化結合し，結果を新しいTRとする

このとき，単一化制約においては，PRまたはTRをm個に分割し，m個の単一化制約として並列に実行できる．また，単一化結合においては，PRをm個，TRをn個に分割し，m×n個の単一化結合として並列に実行できる．

いま，分割して各PEに同時に割当てた全ての演算処理の終了を待って次の演算を分割する場合，1回の演算を次のように分割すると，並列実行による効果が高い．[2]

① 分割数をPE数とし，各PEに1個の処理を割当てる
② 個々の演算処理時間が等しく，かつ最小になるように分割する

しかし，各PEの演算処理時間は予測できない．そこで，データの分布が均一であるとし，次のように仮定する．

① 項関係のデータ量とタプル数は比例する
② 単一化結合で，入力項関係のタプルの組合せに対して結果タプルの出現する割合は一定である
③ UEの単一化結合処理では，まず単一化可能な組の候補を作り，それらが実際に単一化するかどうかを調べる．このとき，候補中の単一化しないものの率を非選択率と呼ぶが，これが一定である．

これにより，次のように，演算をその入力データ量により分割できる．

単一化制約では，UEの演算処理時間は入力データ量にほぼ比例する．よって，入力PRまたはTRをデータ量が等しく

Knowledge Base Machine *Hu-X*（3）A Simulation by Experimental Hardware
Fumihide ITOH[1], Hiroshi SAKAI[2], Shigeki SHIBAYAMA[2]
[1]Institute for New Generation Computer Technology, [2]Toshiba Corp.

なるようにPE数に分割する．このとき，各PEの処理時間は最小になる

単一化結合では，UEの演算処理時間はほぼ式(1)であらわされる．

$$\alpha(p+q)+\beta r+\gamma pq \qquad (1)$$

ここで，p，q，rはそれぞれ入力PR，入力TR，出力TRのデータ量である．また，α，βは定数であるが，γは非選択率に依存する．しかし，データの分布による仮定から，rはpとqの積に比例し，γは一定となる．よって，式(1)は式(2)のようになる．

$$\alpha(p+q)+\delta pq \qquad (2)$$

ただし，δは定数である．よって，PRの分割数mとTRの分割数nの積をPE数とし，それぞれをデータ量が等しくなるように分割する．式(2)の処理時間を最小にするためには，mとnの値によらずpとqの積は一定なので，pとqの和を最小にする，すなわち，pとqの比をできるだけ1に近づけるようなmとnを選ぶ．

このように演算を分割する場合，演算分割の回数，および各PEに割当てられる演算処理の数は，PE数によらず一定である．しかし，PE数の増加により演算がより細かく分割されるため，個々の演算処理の実行時間が減少し，全体の処理時間も減少する．

4．評価

4.1　PE間の処理の競合

SCPとPEの通信，およびHPPHの資源管理にSHを用いるため，複数PEにより演算処理を並列実行する場合，SHのアクセス競合が発生する．この影響を調べるために，試作システムにおいてPE数を変化させ，通信時間とHPPHアクセス時間を測定した．結果を図2に示す．

通信時間は，各PEとSCPの間でコマンドとそれに対するレスポンスの送受信のみを繰返した場合の，1往復あたりの通信時間である．実際の入力演算による全解探索では，分割された1回の演算処理の実行時間は，通信時間の約10倍である．よって，通信処理の競合の影響はほとんどないといえる．

HPPHアクセス時間は，各PEに大量データの項関係をアクセスさせた場合の，1ページあたりのアクセス時間で，試

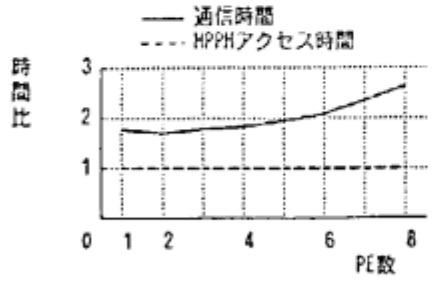

図2　PE間の処理競合の影響

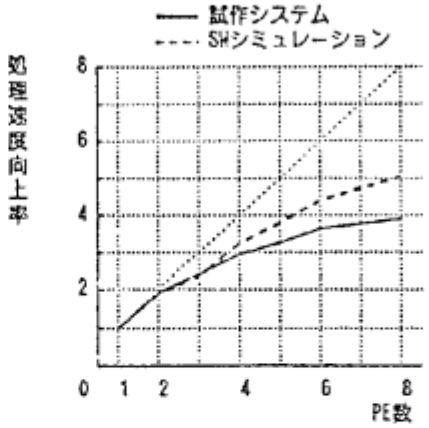

図3　PE並列実行の効果

出しと書込みでは，ほとんど差はなかった．各ページのアクセスの前にSH上の資源情報がアクセスされるが，その競合の影響はまったくない．

4.2　複数PEによる並列実行の効果

試作システムにおいてPE数を変化させ，先祖探索問題[2]の全解探索の処理時間を測定した．PE数による処理速度の向上率を図3に示す．

試作システム上のシミュレーションでもPEの並列実行による効果があらわれている．しかし，その向上率は，ソフトウェアシミュレーションの場合より悪い．これは，試作システム上のシミュレーションによって処理時間に新たに加えられた処理の中に，SCPにおける演算処理の分割やコマンド通信など，PE数によらない一定時間の処理が含まれているためである．

5．おわりに

Mu-Xの試作システムの，複数PEによる演算の並列実行の特性を評価した．試作システムの規模の並列度では，SHアクセスの競合による影響はほとんどないことがわかった．今後は，より高い並列性を引出すために，項関係の動的クラスタリングなどの手法を取入れて，システムの評価を行う．

参考文献

[1] Yokota, H., et al: A Model and Architecture for a Relational Knowledge Base, The 13th Annual International Symposium on Computer Architecture Cnference Proceedings, pp.2-9 (1986)

[2] Sakai, H., et al: A Simulation Study of a Knowledge Base Machine Architecture, 5th International Workshop on Database Machines Proceedings, pp.583-596 (1987)

[3] 酒井他：知識ベースマシンMu-X(2)，本論文集

[4] Tanaka, Y.: A Multiport Page-Memory Architecture and Multiport Disk-Cache System, New Generation Computing, Vol. 2, No. 3, pp.241-260 (1984)

[5] Horita, Y., et al: Retrieval-By-Unification Operation on a Relational Knowledge Base, Proceedings Twelfth International Conference on Very Large Data Bases, pp.52-59 (1986)