

## テクニカルノート

バス結合共有メモリ型マルチプロセッサにおける  
面状バリア同期機構東 由美<sup>†</sup> 茶屋道 宏貴<sup>†</sup> 岩根 雅彦<sup>†</sup>

バス結合共有メモリ型のマルチプロセッサにおいて細粒度の並列処理を行うためには高速な同期機構が必要であり、このようなシステムでは、バスアクセス競合による性能低下が問題となる。そこでバリア型同期機構のハードウェア量について検討し、包含検索機能付 CAM を用いた面状バリア同期機構を提案する。それをバス結合共有メモリ構成である細粒度並列計算機 MSBM 上に実装し、do-loop プログラムを用いて定量的な評価を行った。その結果普通のバリアを用いたものと比較して 16PU 時では 26.8~58.9% の速度向上を得、面状バリアはバストラフィックを抑える効果が得られた。

The Range Barrier Synchronization Mechanism  
on a Bus-based MultiprocessorYUMI HIGASHI,<sup>†</sup> HIROTAKA CHAYAMICHI<sup>†</sup> and MASAHICO IWANE<sup>†</sup>

The bus-based multiprocessor system requires a high-speed synchronization mechanism for executing fine grain parallel tasks. In this paper, we describe the amount of hardware of barrier models and propose a range barrier mechanism using the modified CAM. This mechanism is implemented on bus-based multiprocessor MSBM and is evaluated quantitatively. The speed-up ratio to the normal barrier is 1.27 to 1.59 with 16PUs for doacross-loop programs. Using the range barrier causes a decrease in bus-traffic.

## 1. はじめに

共有メモリ構成された複数のプロセッサ間で協調して並列処理を行うためには同期機構が必要である。これまでにバリア同期機構として普通のバリアの他に SBM<sup>1)</sup>, DBM<sup>2)</sup>, Fuzzy バリア<sup>3)</sup>, Elastic バリア<sup>4)</sup> 等が提案されている。また、このようなバリア同期モデルを分類定式化し、性能の定量的な比較考察がなされている<sup>5)</sup>。一方、単一バス結合による共有メモリ構成のマルチプロセッサではバスアクセス競合による性能低下が問題である。これを解消するためにバスバンド幅の増大、多重バス構成やコードスケジューリングでバス競合を調整するアルゴリズム<sup>6)</sup>も提案されている。

本論文ではバリア型同期機構のハードウェア量を検討し、ハードウェア量と性能向上率を考慮して、MCAM<sup>7)</sup>を用いた面状バリア<sup>5)</sup> (range barrier) 同

期機構を提案する。それをバス結合共有メモリ構成である細粒度並列計算機 MSBM<sup>7)</sup> 上に実装し do-loop プログラムを用いて定量的な評価を行う。

## 2. バリア型同期機構のハードウェア量

同期成立を専用のハードウェアで検出する集中型同期機構として、普通のバリア、バリアキューや CAM (Content Addressable Memory) を用いた SBM および DBM、包含検索機能付メモリ MCAM を用いたバリア同期 SsBM<sup>7)</sup> 等があげられる。普通のバリアは全プロセッサで同期を行う線状バリア<sup>5)</sup> であり、非常にハードウェア量は少ない。同じく線状バリア同期機構である SBM, DBM, SsBM ではマスクを使用することで任意のプロセッサ間での同期が可能である。SBM ではバリア間には全順序関係が存在しエン트리数  $k$  のバリアキューを用いて同期を行う。DBM の一種である Full DBM (FDBM)<sup>2)</sup> ではバリア間には半順序関係が存在し CAM とエン트리数  $k$  の dag memory<sup>2)</sup> を用いて同期を行う。SBM, FDBM ではバリアキューまたは dag memory の管理のためにハードウェ

<sup>†</sup> 九州工業大学工学部電気工学科

Department of Electrical Engineering, Faculty of Engineering, Kyushu Institute of Technology

表1 nプロセッサシステムにおけるバリア同期機構のハードウェア量

Table 1 Amount of hardware for synchronization mechanism on a n-processor system.

|         | MB  | M        | C  | SI | L           |
|---------|-----|----------|----|----|-------------|
| 普通のバリア  | —   | —        | —  | n  | 2n          |
| SBM     | n   | 1+k§     | —  | n  | 2n          |
| FDBM    | n   | n/2   k† | —  | n  | 2n          |
| SsBM    | n   | n/2      | —  | n  | 2n          |
| Fuzzy   | n-1 | n        | —  | 2n | n(n-1) × m‡ |
| Elastic | n-1 | n        | 3n | 2n | n(n-1)      |

§: キューのエントリ数, †: dag memoryのエントリ数, ‡: 多重バリア発行を使用しない場合 m=1, MB: マスクビット幅, M: マスク数, C: カウンタ数, SI: 同期情報ビット数, L: 結線数

アが必要である。SsBMはプロセッサグループ(1つの応用プログラムの実行に割り当てられるPU)に参加する全PUで同期がとられる強制参加バリアである。静的バリア同期管理を前提とし包含検索機能を備えたMCAMを用いることでハードウェア量を削減している。

同期成立検出をそれぞれのプロセッサごとに行う分散型同期機構としてFuzzyバリア, Elasticバリア等があげられる。FuzzyバリアとElasticバリアは面状バリアである。Fuzzyバリアではマスクの他にプロセッサごとに同期識別用のタグを持つことで, 多重バリア発行<sup>3)</sup>が可能である。2<sup>m</sup> - 1のバリアを識別するためにはタグ幅はmビット必要である。またバリア領域のオーバーラップはできない。Elasticバリアではマスクの他に各プロセッサが同期用カウンタを持つことで, バリア領域のオーバーラップを可能にしている<sup>4)</sup>。

バリア同期機構をnプロセッサシステム上に実装したときのハードウェア量を表1にまとめる。なおSsBM, Fuzzyバリア, Elasticバリア等では多重プログラミング環境<sup>7)</sup>が可能である。

### 3. MCAMを用いた面状バリア同期機構

分散型同期機構ではマスクを各プロセッサが保有し, 同期要求信号を各プロセッサがブロードキャストするためハードウェア量は集中型よりも多くなる。また線状バリアに対する面状バリアの性能向上率は9.0~17.8%と高い性能向上が見込まれるのに対して, オーバラップ不可能バリアに対してオーバーラップ可能バリアは1.7~3.2%と大幅な性能向上は見込めないことが報告されている<sup>5)</sup>。そこで面状かつオーバーラップ不可能なバリア同期機構を, MCAMを用いたハードウェア量が少ない集中型同期機構として提案する<sup>7)</sup>。

集中型面状バリア同期機構を持つ細粒度並列計算機MSBMの構成を図1に示す。MSBMは16台のプロ

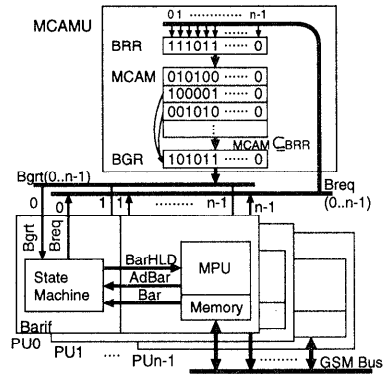


図1 MCAMを用いた面状バリア同期機構

Fig. 1 Range barrier synchronization mechanism using MCAM.

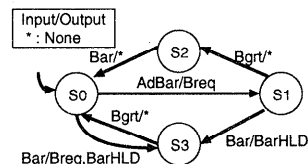


図2 状態遷移図

Fig. 2 The state diagram.

セッサ (PU) から構成されるバス結合共有メモリ型マルチプロセッサである。同期成立検出機構MCAMユニット(MCAMU)は16ビット×8のMCAMを持つ。同期要求(Breq), 同期完了(Bgrt)信号はそれぞれ1ビットである。PUがBreq信号を出すとBRR内の当該ビットがセットされ, BRRの内容についてMCAMを検索して同期成立検出を行う。同期成立したPUへはBGR内の当該ビットがセットされBgrt信号により伝えられる。PUは局所メモリ(LM)とグループ共有メモリ(GSM)を持ち, プロセッサグループごとにGSMのコヒーレンスを保つ。なおメモリバス幅は8ビットである。

プロセッサ側には入出力命令としてAdBar命令とBar命令を用意した。AdBar命令とBar命令でバリア領域の最初と最後を表すことで面状バリア同期を実現し, Bar命令単体では普通のバリア同期を実現する。AdBar/Bar命令を実行するとBarifに対してAdBar/Bar信号が発行される。AdBar, Bar, Bgrt信号を受けてBarifは図2の状態遷移図に従ってMCAMUに対するBreq信号およびMPUに対する実行中断を表すBarHLD信号を出力する。図2の状態遷移図は矢印に添えられた斜線の左側を入力, 右側がそれに対する出力を表す。Barifの状態はS0(PUは非バリア領域の命令を実行, 初期状態), S1(PUはバリア領

```

global: a[M][N]      global: a[N]
local: b[M][N],c[M][N] local: b[N], c[N]
for(i=1;<N;i=i+1)    for(i=1;<N;i=i+1){
  for(j=1;<M;j=j+1){  a[j]=b[i];
    a[j]=a[j]+i[-1]+2.0;  c[j]=a[i+1];
    b[j]=b[j]+c[j][i];  }
  }
  (a) Exam.1      (b) Exam.2
    
```

図3 do ループプログラム  
Fig.3 Do-loop programs.

表2 ループボディ実行に要する実行時間  
Table 2 Time required for executing loop-body.

|       | Exam.1      | Exam.2    |            |            |
|-------|-------------|-----------|------------|------------|
|       |             | 1 byte    | 4 byte     | 8 byte     |
| 線状    | 31.7        | 3.1       | 11.7       | 13.0       |
| 面状    | 31.7 (21.9) | 3.1 (1.4) | 11.7 (5.6) | 13.0 (6.3) |
| doall | 11.7+26.6   | 1.6+1.5   | 6.3+6.4    | 6.9+7.0    |

域の命令を実行，同期未完了)，S2 (PU はバリア領域の命令を実行，同期完了) および S3 (PU はバリア領域の命令の実行終了，同期の完了待ち) の4つである。S3の間PUの実行は停止され，Bgrt信号が伝えられるとただちに実行は再開される。

MSBM 上での普通のバリア構成と面状バリア構成とのハードウェア量を比較すると，面状バリア構成には同期情報として  $2n$  ビット必要である。

#### 4. 実験

細粒度並列処理では頻繁に同期を行うために同期機構が大きく性能に影響する。そこで2つの簡単なdoループプログラム(図3)を，ループ分離によりバリア同期を使用しないdoallの形と，面状バリア，普通のバリアを用いて細粒度doacrossの形でそれぞれ並列化した。Exam.1で扱う配列要素サイズは4byte(浮動小数点型)である。Exam.2では要素サイズを1byte, 4byteそして8byte(倍精度浮動小数点型)の3通りとして実行を行った。それぞれの並列版におけるループボディの実行に要する実行時間を表2に示す。面状バリア版における括弧内の値はループボディにおけるバリア領域の実行時間を示している。doall版ではループ分離されているため各ループボディの実行時間を+で結合して示す。なお，これらの値はAdBar/Bar命令の実行時間を1として正規化したものである。

各PU数に対する結果を図4および図5に示す。これはプログラムの逐次版の実行時間  $T_s$  および各PU数における実行時間  $T_p$  を測定し，逐次版に対する各PU数での速度向上比  $T_s/T_p$  を算出したものを示している。

中粒度並列処理としてポアソン方程式の数値計算を行うプログラムを用いた。ポアソン方程式では，各点

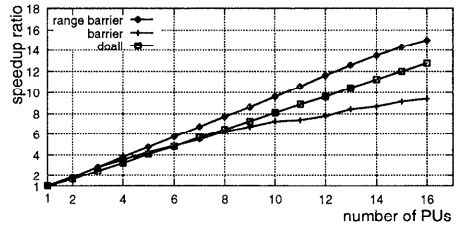


図4 速度向上比 (Exam.1)  
Fig.4 Speedup ratios (Exam.1).

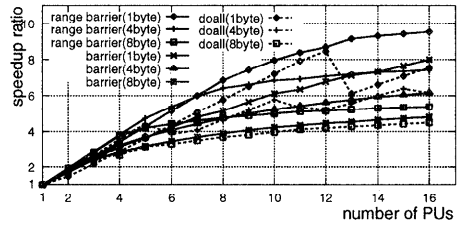


図5 速度向上比 (Exam.2)  
Fig.5 Speedup ratios (Exam.2).

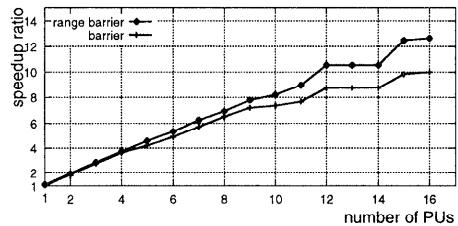


図6 速度向上比 (ポアソン方程式)  
Fig.6 Speedup ratios (Poisson solver).

の計算に4隣接点のデータが必要である。そのため複数のPUで問題のサイズをストライプ分割し，境界部分を共有メモリにそれを局所メモリに配置した。問題のサイズを  $64 \times 64$  とし，普通のバリアおよび面状バリアを用いて並列化を行った。各PU台数時での逐次版に対する速度向上比を図6に示す。

#### 5. 考察

Exam.1において面状バリアは線状バリアに対して0.5~58.9%，doallに対しては14.8~20.6%，Exam.2では線状バリアに対して1.2~35.4%，doallに対しては3.1~49.3%の速度向上が認められ，いずれの実験においても面状バリアを用いたものでは良い結果が得られた。

MSBMのようなバス結合共有メモリ型マルチプロセッサではバス競合による性能低下が問題となる。特に普通のバリアを用いてdoacross処理を行う際にはほぼ同じタイミングでそれぞれのPUがGSM書き込みを要求し，Exam.1の結果が示すように使用PU

数が増えるにつれ台数効果による性能の伸びは小さくなってしまふ。また, Exam.2の結果が示すように扱うデータサイズが大きくなりバストラフィックが増加すると性能の伸びはますます小さくなる。一方, Exam.1において面状バリアを用いたものはPU数にほぼ比例して性能が伸びており, Exam.2においては性能の伸びの低下は見られるものの, バリア領域の実行時間が大きくなるにつれ比例して性能向上が得られるPU台数は増加する。これは初回のGSM書き込み時のバス競合でPUの実行にずれが生じそれ以降のバス競合が緩和されたためである。

ポアソン方程式の数値計算においても面状バリアは線状バリアに対して3.6~26.8%の速度向上が得られた。PU台数が増えるにつれ速度向上比が階段状に増えていくのは, 各PUの実行イタレーション数が均一でなく, PUの最大実行時間で性能が決定するためである。また, Exam.1より面状バリアと線状バリアとの性能の差は小さくなっている。これは粒度が大きくなることでGSMバス競合による性能低下の影響が小さくなったためである。

同期状態の情報を各PUに持たせ, 同期成立検出は専用のハードウェアで行う集中型面状バリア同期機構とすることで, 分散型よりもより少ないハードウェア量で実現することができた。また, モデルによる評価<sup>5)</sup>ではバストラフィックに対する面状バリアの効果は不明であったが, 本実験から面状バリアによりバス競合を緩和することができるという結果が得られた。

## 6. むすび

MCAMを用いた集中型面状バリア同期機構を提案し, バス結合共有メモリ型である細粒度並列計算機MSBM上に機能を限定した形で実装し評価を行った。その結果今回提案したMCAMによる面状バリア同期機構の有用性が確認できた。面状バリアは共有バスのバストラフィックの緩和に役立つため, doacross処理するには有効である。今後の課題は, コンパイラによるバリア領域拡張の自動化である。

## 参考文献

- 1) O'Keefe, M.T. and Dietz, H.G.: Hardware Barrier Synchronization: Static Barrier MIMD (SBM), *1990 Int. Conf. on Parallel Processing*, Vol.1, pp.35-42 (1990).
- 2) O'Keefe, M.T. and Dietz, H.G.: Hard-

ware Barrier Synchronization: Dynamic Barrier MIMD (DBM), *1990 Int. Conf. on Parallel Processing*, Vol.1, pp.43-46 (1990).

- 3) Gupta, R.: The Fuzzy Barrier: A Mechanism for High Speed Synchronization of Processors, *Proc.3rd Int. Conf. on ASPLOS*, pp.54-63 (Apr. 1989).
- 4) 松本 尚: Elastic Barrier: 一般化されたバリア型同期機構, *情報処理学会論文誌*, Vol.32, No.7, pp.886-896 (1991).
- 5) 山家 陽, 村上和彰: バリア同期モデル—Taxonomyと新モデルの提案, および, モデル間性能比較, 並列処理シンポジウムJSP'93, pp.119-126 (May. 1993).
- 6) 尾形 航ほか: マルチプロセッサシステム上の無同期細粒度並列処理, *情報処理学会研究報告*, Vol.92, No.82, pp.149-156 (1992).
- 7) 岩根雅彦ほか: 細粒度マルチプロセッサMSBM, *情報処理学会論文誌*, Vol.37, No.6, pp.1196-1205 (1996).

(平成9年5月9日受付)

(平成9年9月10日採録)



東 由美 (学生会員)

1974年生。1996年九州工業大学工学部電気工学科卒業。現在, 同大学大学院工学研究科電気工学専攻博士前期課程在学中。計算機アーキテクチャの研究に従事。電子情報通信学会会員。



茶屋道宏貴 (学生会員)

1972年生。1992年鹿児島水産高校専攻科修了。同年九州工業大学工学部電気工学科技官。現在九州工業大学工学部電気工学科在学中。計算機アーキテクチャに興味を持つ。



岩根 雅彦 (正会員)

1946年生。1968年京都大学工学部数理工学科卒業。同年(株)東芝に入社。1988年より九州工業大学工学部教授。工学博士。計算機アーキテクチャの研究に従事。電子情報通信学会会員。