ショートノート

BC プロセッサアレイによる連立1次方程式の軸選択消去計算†

小 畑 正 貴††

放送機能を持つ1次元バス型マイクロプロセッサアレイ (BC プロセッサアレイ) 上での，軸選択を伴う連立方程式の消去計算について述べる．まずバス上でのワイヤードオアと放送を利用して軸選択に必要な最大値計算を効率よく行う方法を示す．次に試作機上で軸選択を伴う消去計算を実行し，実行時間をほとんど増加させることなく軸選択が導入できることを実証する．

1. まえがき

放送バスを持つプロセッサアレイにおいて，枢軸要素を逐次放送することにより並列消去が効率よく行えることは既に示しているが，実用上重要な軸選択の導入方法に問題が残されていた[1),2)].

BC プロセッサアレイ[3)]はバスによる放送とバス上でのワイヤードオアを特徴としたプロセッサアレイである．本論文で述べる並列消去方法はバス上でのワイヤードオアを利用して最大値計算を行い，計算時間をほとんど増加させずに軸操作を導入することができる．以下では軸選択に必要な最大値計算とこれを用いた消去計算方法を述べ，試作機による実行結果を示す．

2. 試作 BC プロセッサアレイ

試作機の構成を図1に示す．全体で256台のプロセッサを1次元的に接続した形になっている．プロセッサエレメント (PE) は，8ビットマイクロコンピュータ Z8 を CPU に使い，2k バイトの RAM と 256 バイトの ROM を備えている．また，ホストコンピュータには PC-9801 を用いている．

8ビットの入力バス (BI) と1ビットのオープンコレクタ出力バス (BO) をホストコンピュータに接続する．また PE 間は8ビットの入力 (LI) と出力 (RO) で直列接続する．

データ転送には以下の3種のモードがある．

(1) パイプラインモード

LI と RO を用いてアレイ全体のデータを右にシフトする．

(2) ダイレクトモード

ホストは PE アドレスによって特定の PE を選択し， BI と BO を使ってデータ転送を行う．

(3) ブロードキャストモード

ホストが出力したデータは同時に全 PE に放送される．また全 PE の出力データはバス上で論理和がとられ，ホストに読み込まれる．

3. 最大値計算

ブロードキャストモードにおける以下の手順によって全 PE の持つデータの最大値を求めることができる (図2).

(1) 全 PE はデータの第1ビット (MSB: Most Significant Bit) を BO に出力する．このとき，論理和がとられるため MSB=1 は MSB=0 に優先する．

(2) ホストはこれを入力し，最大値の第1ビットとする．同時に，この値を全 PE にブロードキャストする．

(3) 各 PE は出力したデータ (MSB) とブロードキャストされてきたデータ (MSB) とを比較し，違っていれば計算を中止する．

(4) 以上の操作を第2ビット以降 LSB (Least Significant Bit) まで繰り返していくとホスト上に最大値が求められる．また，計算を最後まで続行できた PE が最大値を持つ PE であることがわかる．

以上により m ビットデータの最大値を m ステップで求めることができる．計算時間はデータの数に依

† Parallel Elimination of Simultaneous Linear Equations with Pivoting on BC Processor Array by MASAKI KOHATA (Department of Electronic Engineering, Faculty of Engineering, Okayama University of Science).

†† 岡山理科大学工学部電子工学科

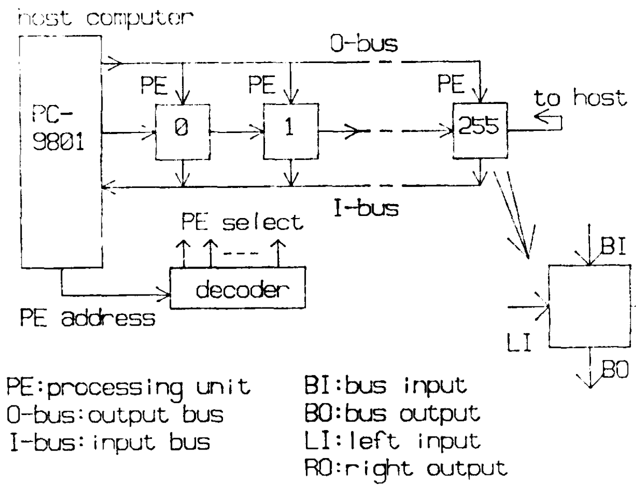

図 1 BC プロセッサアレイシステムの構成
Fig. 1 Structure of the BC processor array system.

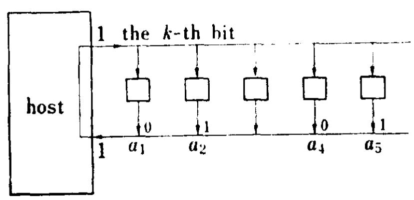

図 2 最大値計算
Fig. 2 Computing maximum number.

存しない．試作機は8ビットデータの最大値を約 300 μs で求めている．最大値を持つ PE が複数ある場合には，データの後に PE 番号をつないで上記の操作を行うことにより1台だけを決定できる．

4. 消去計算

n 元連立一次方程式 $\boldsymbol{Ax}=\boldsymbol{b}$ が与えられた場合，ガウス消去法における前進消去は $k=1\sim n$ に対する以下の操作で与えられる．

（1） a_{ik} $(i=k\sim n)$ の最大値を求め第 k 行と第 i 行とを交換する．

（2） $j=k+1\sim n$ に対して $a_{kj}=a_{kj}/a_{kk}$ を計算．

（3） $i=k+1\sim n, j=k+1\sim n$ に対して $a_{ij}=a_{ij}-a_{ik}*a_{kj}$ を計算．

BC プロセッサアレイへの実装では PE を n 台用意し，係数行列 $\boldsymbol{A}$ の第 i 行とベクトル $\boldsymbol{b}$ の第 i 要素を i 番目の PE に割り当てる（図3）．この場合，上の(2)，(3)の計算時には $PE_1\sim PE_k$ が遊んでしまうことになるので(3)の計算は $i=1$ から行うことにしても計算時間は変わらない．これはいわゆる掃き出し法であり後退代入がいらなくなる．

また(1)での行の交換は実際に行うとデータ転送が多くなるため，データを動かさずにピボット行を担当する PE を換えるようにする．最終結果は図4のようになるが，各 PE が何回目にピボットになったかを記憶しておくことにより，結果の回収時に正しく並べ換えることができる．以上より計算アルゴリズムは次のようになる．

（1） ホストは $\boldsymbol{A}$ と $\boldsymbol{b}$ を行方向に分解し，各 PE に分配する．

（2） k 回目の消去時にはいままでにピボットになったことのない PE のうちの最大値を求め，その PE がピボット PE となる．ピボットとなった PE はその時点で k の値をインデックス値として記憶しておく．

（3） ホストはピボット PE から受け取ったピボット要素を全PEにブロードキャストし，ピボットPEを除く全 PE は並列に消去計算を行う．

（4） 上の(2)，(3)を n 回繰り返す．

（5） ホストは結果とインデックス値を回収し，インデックスによって結果を並べ換える．

5. 実行結果

以下に，試作機による実行時間の測定結果を示す．計算は8ビット固定小数点の形で行っている．PE の演算能力は加減算 数 μs，乗算 約 100 μs，除算 約 120 μs である．

図5に軸操作を行った場合と行わなかった場合の計算時間を示す．軸操作を行っても計算時間はあまり増

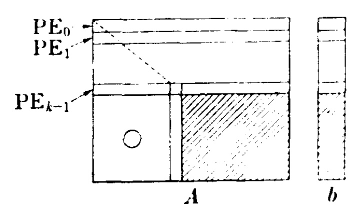

図 3 消去法の計算
Fig. 3 Parallel elimination of linear equations.

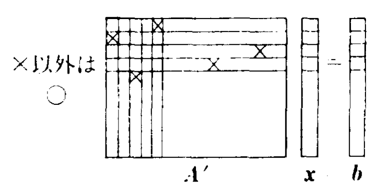

図 4 最終結果
Fig 4 Result matrix.

加せず，いずれも $O(n^2)$ 時間で計算できている．図6では2本のグラフの差をとり，軸操作に要する時間を示しているが，本システムが軸操作を $O(n)$ 時間で処理しているのがわかる．

6. 考　察

以上に示した計算方式および実験結果には，

- データ表現が8ビット符号なし整数である
- PE 台数は方程式の次元数より必ず大きい

という前提があり，このままでは浮動小数点計算および PE 台数より大きい連立方程式に対して適用できない．以下，この点について考察する．

6.1 浮動小数点の場合

浮動小数点計算の場合には軸選択を以下のようにすればよい．

（1） 各 PE で絶対値をとる．

（2） 指数部について最大値計算を行う．

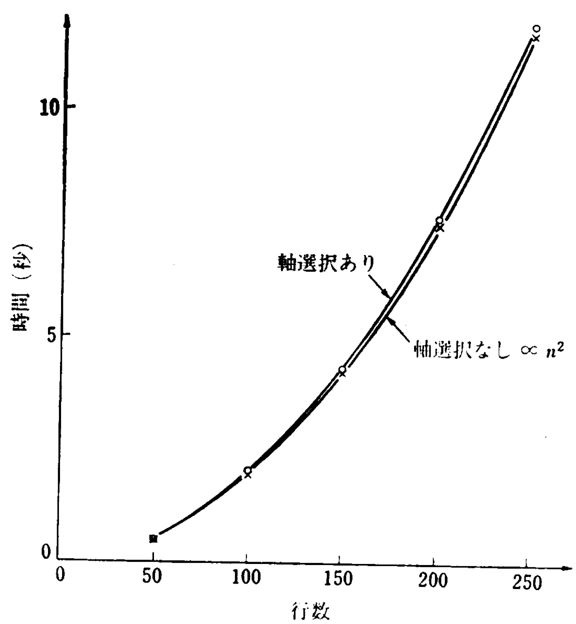

図 5 全計算時間
Fig. 5 Total computing time.

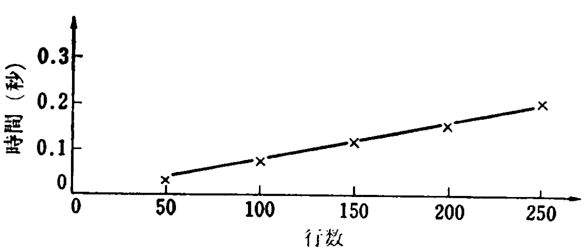

図 6 軸選択の実行時間
Fig. 6 Computing time for pivot selection.

（3） 残った PE の間で仮数部の最大値計算を行う．

（4） さらに PE 番号の最大値を求める．

以上の各操作はすべて単位時間で行われるので軸選択に要する全計算時間は $O(n)$ となる．なお試作機では浮動小数点計算ができないため実測値は示せない．

6.2 方程式の次元数が PE 台数より大きい場合

PE 台数を n，次元数を m として以下の（2），（3）を m 回繰り返せばよい（全計算時間は $O(m^2/n)$）．

（1） PE_i に方程式の $i, i+n, i+2n, \cdots$ 行を割り当てる．

（2） 消去計算は各 PE でそれぞれ複数行に対して行う（$O(m/n)$ 時間）．

（3） 軸選択ではまず PE 内で最大値を求め（$O(m/n)$ 時間），次に PE 間で 6.1 節と同様にすればよい．

軸選択が全計算時間に及ぼす影響については検討中．

7. ま と め

本稿では，BC プロセッサアレイの持つ放送機能を利用した最大値計算方法と，これを用いた軸選択消去計算の並列計算方法を示した．次に試作機での実行により，消去計算時に軸操作を行っても計算時間はほとんど増加せず，（PE 台数≧方程式の次元数）の場合 $O(n^2)$ 時間で処理できることを実証した．

参 考 文 献

1) 金田悠紀夫，小畑正貴，前川禎男：マトリクスブロードキャストメモリ結合形並列計算機による n 元連立方程式の $O(n)$ 時間計算，情報処理学会論文誌，Vol. 23, No. 5, pp. 570-575 (1982).

2) 小畑正貴，金田悠紀夫，前川禎男：ブロードキャストメモリ結合形マルチマイクロプロセッサシステムの試作，情報処理学会論文誌，Vol. 24, No. 3, pp. 351-356 (1983).

3) 小畑正貴，宮垣嘉也，金田悠紀夫：試作 BC プロセッサアレイとその評価，情報処理学会論文誌，Vol. 27, No. 9, pp. 909-915 (1986).

（昭和 62 年 3 月 12 日受付）
（昭和 62 年 9 月 9 日採録）

小畑　正貴（正会員）

昭和 32 年生．昭和 55 年神戸大学工学部電子工学科卒業．昭和 60 年同大学院自然科学研究科システム科学専攻（博士後期）修了．学術博士．現在，岡山理科大学講師．計算機アーキテクチャ，並列処理などの研究に従事．電子情報通信学会会員．