

DOACROSS型ループにおける最適なデータ通信順序 8 L-4

山名早人 村岡洋一
早稲田大学理工学部

1. まえがき

本報告では、DOACROSS型ループ⁽¹⁾の実行時間を最小にするデータ通信順序を求める。DOACROSS型ループ実行に関する従来の研究⁽²⁾⁽³⁾は、プロセッサの処理能力を表すパラメータとして演算命令の実行時間(以下、演算時間)、及び、データ通信の遅延時間(以下、通信遅延時間)を用いてきた。しかし、演算と通信を並列に処理できるマルチプロセッサ上で、DOACROSS型ループを実行する場合、これらのパラメータ以外に、通信ピッチを考慮しなくてはならない。通信ピッチは、プロセッサと相互結合網間のデータ入出力時間間隔である。通信ピッチがデータ通信の発生する時間間隔より大きい場合、通信が全体の実行時間の経路となる。これは、データ通信が通信ピッチ以下の時間間隔で開始(以下、発行)できず、通信発行に遅延が生じるためである。この時、実際の実行時間は、従来の理論的な値よりも大きくなる。以下では、このような場合、データを定義順で他のプロセッサへ送らず、通信順序を変更することにより、実行時間を短縮できることを示す。

2. モデルの説明

2.1 対象マルチプロセッサシステム

対象システムは、演算及び通信を並列に実行できるプロセッサ(Pe)から構成される。Peは1つのALUとローカルメモリを持ち、他のPeとの通信のために入力出力ポートを各々1つ以上持つとする。入力ポートと出力ポートを別々に持つことにより、入出力間の競合を回避できる。以下の2つのパラメータによりモデル化する。

データ通信遅延時間(δ) Pe_iからPe_j($i \neq j$)へ1データを送信する時、Pe_iがデータ送信処理を開始後、Pe_jがデータを受信完了するまでの時間を表す。

通信ピッチ(P_c) 1データをPeと相互結合網間で入出力する時の最短時間間隔を表す。送信と受信は、独立に常に P_c の間隔で通信できるとする。

2.2 対象ループ

DOACROSS型ループは、繰返し回数N、文数m、ディレイ d_0 ⁽¹⁾(d_0 は、データ通信遅延時間 δ を含めて算出⁽¹⁾)。また、データ依存距離が1以上のフロー依存の総数をkとし、データ定義を行なう文番号が若い順に、1から番号付けし、C_i($i=1, \dots, k$)、C_iのデータ依存距離を Δ_{i-1} で表す。

2.3 DOACROSS型ループのスケジューリング

DOACROSS型ループは、文献⁽⁴⁾に示されるようにインデックス方向へ順にPeが割り当てられるものとする。

3. DOACROSS型ループの実行時間

通信ピッチ P_c 及びデータ通信の発生する時間間隔を考慮した場合のDOACROSS型ループの実行時間を求める。

3.1 記号定義

本節で用いる記号を表1に示す。

表1 記号定義

S _i (i)	i番目のフロー依存のソースとなる文の文番号。
S _d (i)	i番目のフロー依存のディステイニエーションとなる文の文番号。
(S _i , S _n)	文S _i の実行開始から文S _n の実行終了までの時間。
T _i	S _d (i)終了時刻とS _i (i)実行開始時刻との差の時間であり、符号はS _d (i) > S _i (i)の時に正、S _d (i) < S _i (i)の時に負とする。
TT _i	通信ピッチ P_c を考慮した場合のデータ通信発行時刻。

3.2 データ通信余裕時間

データ通信余裕時間(TM_i) 対象ループがディレイ d_0 で実行された時、 d_0 を増大させることなく、i番目のフロー依存関係を保証できる最大のデータ通信発行遅延時間であり、データ定義時刻から実際のデータ通信発行までの最大遅延可能時間を示す。

$$TM_i = d_0 + \Delta_{i-1} - T_i - \delta \quad (1)$$

図1に示すDOACROSS型ループを例のディレイ d_0 ⁽¹⁾は、 $\delta = 0$ 、各文の実行時間を1単位時間とすると、 $d_0 = 2$ となる(図2)。

```
DO 1-1,N
  A(1)->B(1-2)+37 ... S1
  B(1)->A(1)+5 ... S2
  C(1)->D(1-1)+B(1) ... S3
  D(1)->C(1)/2 ... S4
  E(1)->D(1)+C(1-1) ... S5
  ..... .... Sm
CONTINUE
```

図1 DOACROSS型ループ例

従来の実行方式では、データの定義順、すなわち、S2で定義したB(i)を逆信後、S3,S4で定義されるC(i), D(i)を順に通信する。C_iのフロー依存(S4→S3)は、ディレイ d_0 の決定要因となっており、データ通信の発行時刻を遅らすことができない。これに対してC_i、C_jは、図2の点線矢印で示すようにデータ通信発行時刻を遅らすことができる。この最大遅延可能時間時間がデータ通信余裕時間 TM_i であり、 $TM_1=2$ 、 $TM_2=3$ 、 $TM_3=0$ となる。

PE1 (i=1) PE2 (i=2) PE3 (i=3)

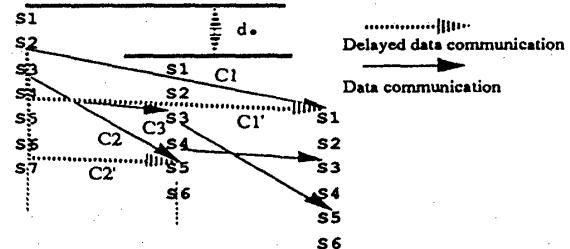


図2 図1の実行タイミングチャート

3.3 データ通信発行遅延時間

データ通信発行遅延時間(TD_i) 通信ピッチ P_c 間隔で通信を行った時、 TM_i を越えて、データ通信の発行が遅れる時間である。ただし、データ通信の開始時刻(データ通信発行時刻 TT_i)が TM_i 内におさまる場合は、0とする。

データ通信発行遅延 TD_i と、通信ピッチ P_c 、データ通信余裕時間 TM_i 、データ通信発行時刻 TT_i の関係を図3に示す。 TD_i と TT_i を求めるアルゴリズムを図4に示す。アルゴリズム適用時の前提を以下に示す。

- (1)データ通信はフロー依存C_iの添字iの順で行われる。
- (2)データ定義文が等しい複数のフロー依存は、データ通信余裕時間の小さい方に若いインデックスを付加する。

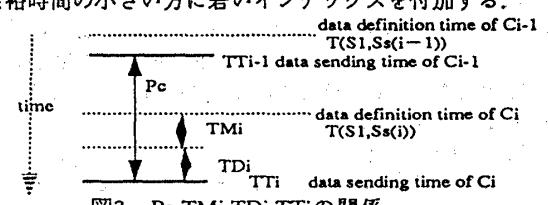


図3 P_c, TM_i, TD_i, TT_i の関係

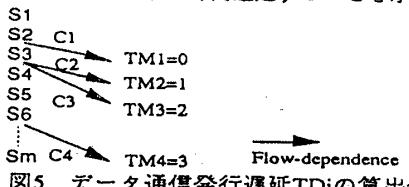
```

k個のフロー依存を持つDOACROSS型ループ実行時のデータ通信発行遅延時間
TDi及びデータ通信発行時刻TTiを求めるアルゴリズム。
INPUT : R, TM, Ss(i) (i=1,...,k)
OUTPUT : TDi, TTi (i=1,...,k)
TDi = 0
TTi = T(S1, Ss(i))
for i=2,k
    PITCH=T(S1, Ss(i))-TM
    ;Ci-1のデータ通信発行時刻からCiのデータ定義までの時間
    if (PITCH+TM<Pc) then
        TDi=Pc-PITCH-TM
        TM=T(S1, Ss(i))+TM+TDi
    else
        TDi=0
        TM=T(S1, Ss(i))+MAX(0,PcPITCH)
    endif
endfor

```

図4 データ通信発行遅延とデータ通信発行時刻を求めるアルゴリズム

図5において、各文の実行時間を全て1単位時間、P_c=2とすると、図4の手順により、TD₁=0, TD₂=0, TD₃=1, TD₄=0となる。これは、P_cの間隔でデータ通信を行った時、C₃のデータ通信が1単位時間遅延することを示す。

図5 データ通信発行遅延TD_iの算出例

3.4 通信ピッチを保証するディレイd_p

データ通信発行時及びデータ受信時に通信ピッチP_cの間隔でデータ通信が行われる時のディレイd_pを求める。

3.4.1 発行間隔保証 TD_iによってDOACROSS型ループが各イテレーション毎に遅れる時間は、TD_i/△_iなので、TD_i/△_i (i=1..k) の最大値で示される時間分、各イテレーションの実行開始が遅延を受ける。

$$d' = d_0 + \max\{TD_i / \Delta_i\} \quad (2)$$

3.4.2 受信間隔保証 データ通信発行が通信ピッチP_cの間隔で行われ、かつ、通信ピッチP_cの間隔でデータが受信される時のディレイd_pを求める。ディレイd'で各イテレーションが実行される時、フロー依存によるデータ通信は、各イテレーションの開始時刻を0とする。時刻TT_iに発行される。時刻TT_iに発行されたデータは、データ通信遅延時間δ後、ディスティネーションPEに到着する。ディスティネーションPEは、ソースPEに比較して、△_i × d'時間遅れて実行を開始しているので、ディスティネーションPEでは、時刻TT_i + δ - △_i × d' (データ到着時刻RT_iで表す) に到着する。しかし、ディスティネーションPEは、通信ピッチP_cより短い間隔でデータを受け取れないため、送信されたデータは相互結合網中に一時的にバッファリングされ、受信時刻が遅延する場合がある。この遅延を考慮したデータ受信時刻(RT_iで表す)をk個のフロー依存によるデータ通信について求め、受信データが演算中で参照される時刻(REFT_iで表す)までに全て受信されるようにディレイd_p (≥d')を求める。求出手順は、本報告では割愛する。詳細は文献⁽⁴⁾を参照していただきたい。

3.5 DOACROSS型ループ実行時間

通信ピッチP_cを考慮しない場合、すなわち、P_cが十分に小さく、データ通信の発生間隔でデータ通信が出来る場合と、P_cの間隔でデータ通信が行われるとした場合の各々について、DOACROSS型ループの実行時間を求める。

(1)通信ピッチP_cを考慮しない場合 文献⁽¹⁾に示される実行時間と等しい。実行時間をT_oとすると次式となる。

$$T_o = d_0 \times (N-1) + T(S_1, S_m) \quad (3)$$

(2)通信ピッチP_cを考慮する場合 P_cの間隔でデータ通信が行われる時の実行時間T_pは、次式となる。

$$T_p = d_p \times (N-1) + T(S_1, S_m) \quad (4)$$

以上より、通信による実行時間遅延T_dは、

$$T_d = T_o - T_p = (d_p - d_0) \times (N-1) \quad (5)$$

となる。このように、通信ピッチP_cを考慮することによって、実行時間に遅延が生じることが明らかになる。

4. 最適データ通信順序

本節では、データ通信順序を変更し、ディレイd_pをディレイd₀に近付ける手法を提案する。

4.1 データ通信順序が実行時間に与える影響

図2の例では、d₀=2、フロー依存C₁, C₂, C₃のデータ通信余裕時間TM_iは、2,3,0単位時間となる。データ定義順でデータ通信を行った時、P_c=2とすると、データ通信発行遅延時間TD_iは、0,0,2となる。この時、通信ピッチを保証するディレイd_pを求めるときd_p=4となる。これに対して、[C₁, C₂, C₃]の順で通信すると、全てのデータ通信発行遅延時間を0にできる。この時、通信ピッチを保証するディレイd_pを求めるときd_p=2となる。データ通信順を変更する前後の実行時間の差は、式(5)より、T_d=(4-2)×(N-1)となり、データ通信順序を変更する前の実行時間が(4×(N-1)+T(S₁, S_m)であるので、Nが十分に大きい時、ほぼ2倍に高速化される。

4.2 最適データ通信順序求出

DOACROSS型ループ内にk個のフロー依存が存在する時、k!通りのデータ通信順序が存在する。このk!通りの順序中に、データ定義順でデータ通信する場合のディレイd_pと同一、あるいは、ディレイd_pより大きなディレイを持つ順序対が存在する。このような順序対は、データ通信発行時に生じるデータ通信発行遅延時間に着目すると、d_p ≤ d₀ + MAX[TD_i / △_i]となるTD_iを持つ順序対である。このような順序対をk!通りのデータ通信順序対から除いた順序対についてのみ、P_cを考慮する場合の実行時間遅延T_dを求め、T_dを最小とするデータ通信順序を求める。求出手順は、本報告では割愛する。詳細は文献⁽⁴⁾を参照していただきたい。

図2の例では、3つのデータ通信が存在し、全ての順序の組み合せは3!=6通りとなるが、本節で示した手順を用いることにより、[C₁, C₂, C₃]と[C₂, C₃, C₁]の2通りの順序対を得ることができる。この2通りの場合について、ディレイd_pを求める、式(5)で与えられる実行時間遅延T_dを最小にする順列を求め、最適なデータ通信順序とする。図2の例では、[C₁, C₂, C₃]の順が最適なデータ通信順序となる。

5. むすび

本報告では、DOACROSS型ループ実行時における最適なデータ通信順序を求める手法を示した。本手法は、(データ通信発行時間間隔) < (通信ピッチ)の場合に、有効な手法である。最近の並列処理計算機は、PE間の通信能力に比較して、PE内部の処理能力が高くなる傾向にあり、本稿で示した手法は、今後、重要なと考えられる。

文献

- [1]Ron Cytron:"Doacross: Beyond Vectorization for Multiprocessors", Proc. of Int. Conf. on Parallel Processing '86, pp.836-844(1986).
- [2]Peiyi Tang, Pen-Chung Yew, and Chuan-Qi Zhu:"Compiler Techniques for Data Synchronization in Nested Parallel Loops", Proc. of ACM 1990 Int. Conf. on Supercomputing, pp.177-186(1990).
- [3]Hong-Men Su and Pen-Chung Yew:"Efficient Doacross on Distributed Shared-memory Multiprocessors", Proc. of Supercomputing'91, pp.842-853(1991).
- [4]山名,村岡:"DOACROSS型ループにおける最適なデータ通信順序",電子情報通信学会論文誌分冊D投稿中