

分散メモリ型並列計算機における式評価時の 4M-2 データ転送回数最小化アルゴリズム

太田 寛¹⁾ 迫田 行介¹⁾ 斉藤 鉄郎²⁾ 前田 栄一郎³⁾ 山本 俊之²⁾

(株)日立製作所システム開発研究所¹⁾ (株)日立マイコンシステム²⁾ 日立ニュークリアエンジニアリング(株)³⁾

1. はじめに

分散メモリ型の並列計算機では、プロセッサ間のデータ転送に伴うオーバーヘッドが大きい、プログラム作成にあたっては、データ転送回数をできるだけ少なくすることが重要である。本報告では、分散メモリ上に割り付けられているデータから構成される式の値を評価するときに、各プロセッサのメモリへのデータの割り付け方が与えられているという条件のもとで、データ転送回数が最小になるように、式の中の演算をプロセッサ(以後PEと書く)に割り当てるアルゴリズムを提案する。

2. 問題設定

例として、FORTRANのDOループの中の

$$(A(I+1)*B(I+1) + C(I+2)) * D(I+3) + E(I+2) \dots \dots \text{[式1]}$$

という式を考える。IはDOループの制御変数である。配列要素は、図1のように各PEに1個ずつ順に割り付けられているものとする^{1) 2)}。添字がI+k(kは整数)という形の配列要素が割り付けられているPEをPE[k]で表し、kを(相対)PE番号と呼ぶことにする。以下では、式を木の形で表す。なお、式の中に同一データは2回以上現れないものとする。葉にはそれに対応するデータを持つPEのPE番号を付ける。また内点にはそれに対応する演算を実行するPEのPE番号を付ける。例えば、図2の[a]、[b]は、それぞれ、[式1]の値を次のような手順で評価することを表す。

[a] C(I+2)、E(I+2)の値を、PE[2]からPE[1]に転送する。また、D(I+3)の値を、PE[3]からPE[1]に転送する。PE[1]で[式1]全体の値を評価する。

[b] 部分式 $A(I+1)*B(I+1)$ の値をPE[1]で求め、PE[2]に転送する。また、D(I+3)の値を、PE[3]からPE[2]に転送する。PE[2]で[式1]全体の値を評価する。

木の枝で、両端の節点に付けられたPE番号が異なるものは、PE間のデータ転送を表している。このような枝を、以後「転送枝」と呼ぶ。転送枝の個数はデータ転送の回数を表している。図2の[a]、[b]では、転送枝の個数はそれぞれ、3、2である。

考えている問題は次のように述べるができる。

[問題]

葉のみにPE番号が付けられている木が与えられたときに、転送枝の個数が最小になるように、内点にPE番号を付けよ。

3. アルゴリズム

この問題を解くアルゴリズムを述べる。簡単のため、どの内点も2個の子を持つものとする。

まず、次の記号を定義する。

LPN(n): 葉nにもともと付いているPE番号。

DPN(n): 節点nに最終的に付けられるPE番号。以後、これを「確定PE番号」と呼ぶ。

CPN(n): 節点nの確定PE番号の「候補」のPE番号の集合。「候補」の具体的な決め方は、アルゴリズムの中で述べる。この集合に含まれるPE番号を、その節点の「候補PE番号」と呼ぶ。

LCH(n): 内点nの左の子。

RCH(n): 内点nの右の子。

PRNT(n): 節点nの親。

SELECT(penset): 候補PE番号の集合pensetから任意に選ばれた、1個のPE番号。

アルゴリズムの入力、出力、方法は、次の通りである。

[入力]

各葉nに対してLPN(n)が定義された木。

An Algorithm for Minimizing the Data Transfer in Evaluating an Expression on a Distributed-Memory Multiprocessor

Hiroshi OHTA¹⁾, Kousuke SAKODA¹⁾, Tetsuo SAITO²⁾, Eiichiro MAEDA³⁾, Toshiyuki YAMAMOTO²⁾

Hitachi Systems Development Lab.¹⁾, Hitachi Microcomputer Engineering Ltd.²⁾,

Hitachi Nuclear Engineering Co., Ltd.³⁾

図1 配列の分散割り付け

| PE1 | PE2 | PE3 | |
|------|------|------|-------|
| A(1) | A(2) | A(3) | |
| B(1) | B(2) | B(3) | |
| C(1) | C(2) | C(3) | |
| D(1) | D(2) | D(3) | |
| E(1) | E(2) | E(3) | |

[出力]

転送枝の個数が最小になるように、各節点nに対してDPN(n)が定義された木。

[方法]

[ステップ1] 木をボトムアップにたどり、次の規則にしたがって各節点のCPN(n)を求める。

```

if(葉)then
    CPN(n) := { LPN(n) }
else /* 内点 */
    if(CPN(LCH(n)) ∩ CPN(RCH(n)) ≠ ∅)then
        CPN(n) := CPN(LCH(n)) ∩ CPN(RCH(n))
    else
        CPN(n) := CPN(LCH(n)) ∪ CPN(RCH(n))

```

[ステップ2] トップダウンに木をたどり、次の規則に従って、各節点のDPN(n)を決める。

```

if(根)then
    DPEN(n) := SELECT(CPN(n))
else
    if(DPN(PRNT(n)) ∈ CPN(n))then
        DPN(n) := DPN(PRNT(n))
    else
        DPN(n) := SELECT(CPN(n))

```

4. 正当性の証明(概略)

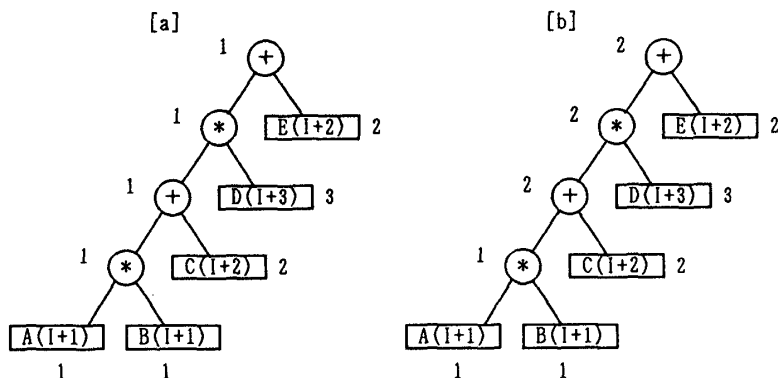
[定理]

上述のアルゴリズムに従って内点に確定PE番号を付けた木は、転送枝の数が最小になっている。

この定理の証明の概略を述べる。

証明の第一段階では、アルゴリズムの前半の[ステップ1]の過程で付けられた候補PE番号について、次の補題が成り立つことを、証明する。

図2 PE割り当ての例



[補題]

```

∀n, ∀i ∈ CPN(n)
    MINTR(n | DPN(n) = i) = MINTR(n)
かつ、
∀n, ∀i ∉ CPN(n)
    MINTR(n | DPN(n) = i) > MINTR(n)

```

ここで、nは節点、MINTR(n)はnを根とする部分木の転送枝の最小値、MINTR(n|<条件>)は<条件>のもとでのnを根とする部分木の転送枝の最小値である。

この補題は、葉を出発点とする帰納法によって証明できる。

証明の第二段階では、この補題を用いて、アルゴリズムの後半の[ステップ2]の手順に従って確定PE番号を付ければ、確かに転送枝が最小になるようなPE割り当てが得られることを示す。

5. おわりに

分散割り付けされたデータから構成される式の値を評価するとき、データ転送回数が最小になるように演算をPEに割り当てるアルゴリズムを提案した。(1) 式が多分木で表される場合、(2) 代入がある場合、(3) 加算や乗算の順序交換が可能な場合についても類似のアルゴリズムを見いだしている。

6. 参考文献

- 1) 太田、他：並列計算機用FORTRANの並列化トランスレータの基本構想；第40回情報処全国大会講演論文集、1990.
- 2) 斉藤、他：並列計算機用FORTRANのDOループ実行制御方式；第40回情報処全国大会講演論文集、1990.