

超並列 C 言語 NCX の MIMD 計算機用コンパイラにおける 仮想プロセッサ割り当ての手法

4 T-11

日高 宗一郎, 川合 史朗, 相田 仁, 齊藤 忠夫

東京大学 工学部

1 はじめに

超並列 C 言語 NCX* [1] は、無限次元のメッシュ、超立方体などのトポロジで結合された無限個の仮想プロセッサ (以下 VP と略記) による並列計算のセマンティクスを採用している。実装上は、宣言された VP の集合を並列計算機上の有限個の物理プロセッサ (以下 PP と略記) へ割り当てることになるが、その方法によっては PP 同士の通信量の増減等で大きく性能が左右される。本稿では、この VP の PP への割り当てに多様な自由度を持たせるための割り当て記述法、および効率のより良い割り当ての実現の手法について述べる。

2 マッピングジェネレータの概要

NCX では、有限個の VP の集合であるフィールド (field) を選択することにより、それを構成する VP により実際の並列計算を行なう。プログラマはこのフィールドから VP へのマッピングをある程度の自由度 (メッシュトポロジではフィールドの次元以上の次元の空間への制限付き線形変換) をもって指定することが出来る。コンパイラは、このフィールドマッピングにこめられたプログラマの意図を汲みつつ、効率よくフィールドを PP へ割り当てなければならない。

今回のコンパイラでは、上記の写像を行う関数及び逆関数を、幾つかのパラメータから VP エミュレーションループから直接呼び出せる形で機械的に生成する。この生成は独立したパスで行なう。ここでいう逆関数は、VP エミュレーションの際に、フィールドインデックスの参照が起こる時、ループインデックスと PP 番号からフィールドインデックスを逆算する際に必要となる。

上記のパラメータとは、マッピングの方法を各次元別に指定する記号列または数列である。これらのパラメータは、前段までに抽出されたフィールド情報等 (図 1) を用いて、写像指令決定部により、自動生成を目指す。プログラマが明示的に与えることもできる。現段階では、NCX の提供するいくつかの VP トポロジのうち、メッシュ

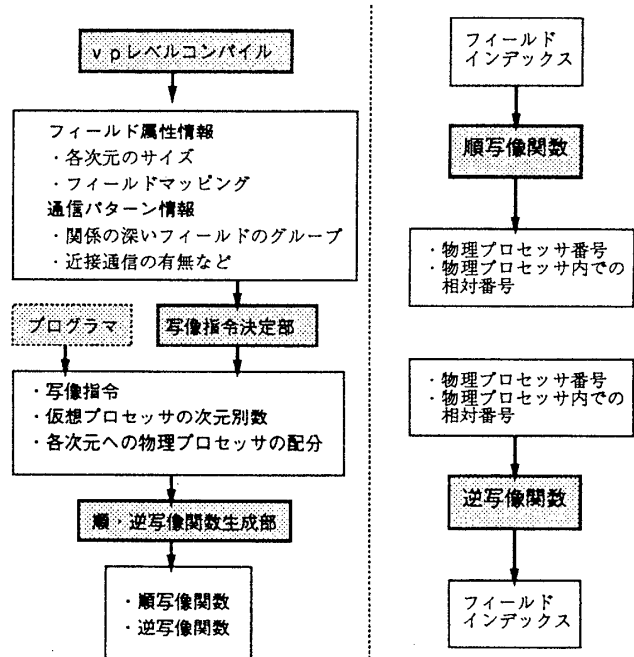


図 1: 写像関数の生成過程 (左) と写像関数の働き (右)

トポロジのみに対応している。

フィールドは、各次元の大きさがコンパイル時に評価できる静的なフィールドと、そうでない動的なフィールドがあり、本稿では、コンパイル時にマッピング関数が決定できる静的なフィールドを対象とする。従って写像関数は、C 言語のマクロとして生成される。

3 写像パラメータ

3.1 パラメータ

まず、 N 個の正数列 $\{a_i | 0 \leq i \leq N-1\}$ をパラメータとする N 個の正数列 $\{b_j | 0 \leq j \leq N-1\}$ から正整数への関数

$$f[\{a_k\}](\{b_k\}) = \sum_{j=0}^{N-1} \left(\prod_{i=0}^{j-1} a_i \right) b_j$$

を定義する。但しここで積記号については、インデックスが負の範囲では 1 による乗算を行うとする。

以下に、上で述べたパラメータを列挙する。また、ここで述べられる VP インデックスとは、フィールドイン

*Assigning Virtual Processors onto MIMD Computers in a Compiler for Massively-Parallel C Language NCX"

Souichirou Hidaka, Shiro Kawai, Hitoshi Aida and Tadao Saito

Faculty of Engineering, The University of Tokyo

*NCX は文部省重点領域研究「超並列原理に基づく情報処理基本体系」で開発された言語である

デックスにフィールドマッピングを施したのちのインデックスを言う。[†]

- 写像指令 (mapper directive) 各次元ごとの写像方法を指定する N 個の記号列 $\{d_i | d_i \in \{b, c, s, m\}, 0 \leq i \leq N-1\}$ である。
 - b Block 割り当て
b を指定された次元は、隣り合う VP が同一の PP でエミュレートされるように写像される。
 - c Cyclic 割り当て
c を指定された次元は、隣り合う VP が異なる PP で (同じ voffset で) エミュレートされるように写像される。この指令は、例えば VP の計算量に著しい偏りがある時など、負荷を PP へ均等に配分するのに有効である。
 - s Shuffle 割り当て
s を指定された互いに隣接する次元は、前に定義した関数 f を用いて 1 次元に折り畳まれたのち、mapper directive に c を与えられた時と同じ処理を受ける。この指令は、VP を PP に均等に割り当てる手段でもある。
 - m Meet
m を指定された全ての次元は、隣接していてもすべて s と同じ処理により 1 次元に折り畳まれる。内部では、m が隣接するようになる置換操作を与えられた他のパラメータすべてに対して施し、しかる後隣接するようになった m の並びを s へ置き換えて処理を続行する。
- 各次元への PP の配分を表す正数列 $\{p_i | 0 \leq i \leq N-1\}$
- 各次元の大きさを表す正数列 $\{s_i | 0 \leq i \leq N-1\}$

3.2 VP 間通信パターンと写像パラメータ

ソース上の全ての VP 間通信パターンがランダムであれば、PP の負荷が均等になるように各 PP に均等に VP を割り付ける以上の工夫の余地はないが、多くのアプリケーションにおいて VP 間通信は何らかの”偏り”を持つ。その偏りを利用してマッピングを行えば PP 間通信を軽減することが出来る。以下に、前段までに抽出される通信パターンと、それぞれに適したマッピングが如何にして写像パラメータで表現できるかを検討する。

- reduction
複数の VP が一つの VP に値を書き込む場合である。和などの演算を伴う場合も含む。リダクションの方

[†]VP インデックスは負の値になる場合があるが、その時はその次元の VP の数で modulo がとられる。

向が、ある次元に偏っていることが分かり、その方向での通信のコストが大きいならば、その次元は同一の PP にマップされることが望ましい。この場合、対応する次元の p_i を小さくする (1 を指定すると全て同一の PP にマップされる)。

- broadcast
一つの VP が複数の VP に値を書き込む場合である。これも reduction と同様に次元の偏りがあれば、対応する次元の p_i を小さくする。
- permutation
値を送信する VP と受信する VP が 1 対 1 に対応する場合である。この場合も相手の次元に偏りがある場合は上記の手法が有効である。また、相手が隣接している場合はその次元に写像指令 b を指定すれば隣接する VP の多くが同一 PP に割り当てられ、PP 間通信が軽減され得る。さらに、相手との間隔が一定である場合、写像指令 c を指定し、更にその次元の s_i を考慮して p_i を適当に決めれば通信相手を同一の PP にマップすることが可能である。

4 写像関数の生成

例として、前節のパラメータのうち、フィールドが 2 次元で $\{s_i\} = \{256, 32\}$ 、フィールドインデックスが i, j 、 $\{p_i\} = \{8, 4\}$ 、写像指令が $\{b, c\}$ である場合に生成された関数を示す。S 式の形を取っているが、コード生成部で C 言語の式に変換される。

```
(+ (* 1 (% i 8)) /* PP 番号 */
  (* (* 1 8) (quotient j (ceiling (/ 32 4))))
  (+ (* 1 (quotient i 8)) /* PP 内の相対位置 */
    (* (* 1 (ceiling (/ 256 8))
      (% j (ceiling (/ 32 4))))
    (* (ceiling (/ 256 8)) /* 1PP でエミュレートされる VP 数 */
      (ceiling (/ 32 4)))
```

5 まとめ

本稿では、NCX のコンパイラのうち、VP を効率良く PP へ割り当てるための写像関数を生成するパスについて述べた。写像関数は、いくつかのパラメータをもとに自動生成される。パラメータは自動生成またはプログラマにより指定される。3.2 節において、自動生成の指針を示した。

参考文献

- [1] 文部省重点領域研究「超並列原理に基づく情報処理基本体系」B 班、「超並列 C 言語 NCX 言語仕様案 Version3」, 1993 年 9 月