

画像処理用超並列プロセッサAMPにおける動的PE間通信とその評価

4V-6

山元規靖 鶴田直之 谷口倫一郎 雨宮真人
九州大学大学院総合理工学研究科

1 はじめに

筆者らは、柔軟で高能率な並列処理を実現する画像処理用超並列プロセッサAMPとそのプログラミング言語Valid-Aを開発している[1][2]。AMPでは、画像の早期処理から画像理解のレベルまで種々の処理を実行させることを設計の基本方針としている。

AMP上の各PE(Processor Elements)は、共有メモリを持たず非同期に動作し、格子状のネットワークにより結合されている。また、各PEは単純な静的データフローモデルを採用しており、通常、通信の行き先は静的に決定される。よって、入力データなどにより通信の行き先を変える処理では、プログラムの記述量が膨大になる等の問題が生じる。本稿では、効率良く通信の行き先を変える手法とそれに伴うValid-Aの拡張、およびその評価について述べる

2 AMPにおける動的通信

AMPにおける連結成分のラベル付け処理では、効率の良い手法として画像をいくつかの細領域に分割して領域の境界線を並列にトレースし、この情報

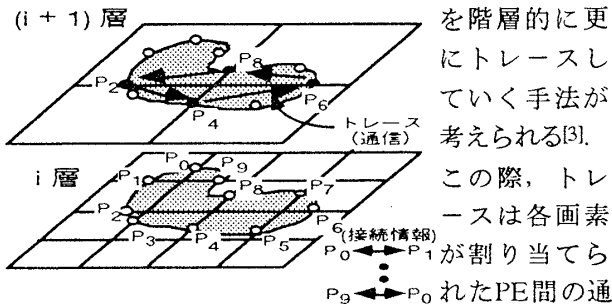


図1 ラベル付けにおける境界線トレース

を階層的に更にトレースしていく手法が考えられる[3]。この際、トレースは各画素が割り当てられたPE間の通信を行うことによってなされ、この通信の行き先は入力(画像)によって動的に決まるものである(図1参照)。

AMPにおいて、このような行き先が動的に変わる通信では、予め可能性がある行き先を求めておき、条件式により行き先を決める手法が考えられる。しかし、この場合、可能性がある行き先をすべて保持

しておくための記憶領域が費やされ、実用的ではない。また、行き先を決定するために多数の条件式を実行しなければならないため、効率が悪い。

AMPでは、通信は通信データと行き先(行き先のPE番号+OMアドレス)の2オペランド命令sendによって実行される。動的通信の行き先を格子状の各PEのOM(Operand Memory)に規則的に配置すれば行き先を動的に簡単な計算により求められる。この計算によって求めた行き先を通信の行き先オペランドとして使えばメモリ(OM)を無用に費やさず効率的な動的通信を行うことができる。

実際の通信の行き先の規則的な配置は、以下の2つの方法が考えられる。

2.1 行き先テーブルを作る手法

図4(a)に示すように、行き先となるPE内OMの一定アドレス(base address)に行き先のテーブルを作る。このテーブルは、行き先の所定アドレスへのsendテーブルとなっている。この手法では、計算体を規則的に並べる必要がないので、メモリの断片化が起こらないが、sendテーブルから所定の位置に再度sendされるため多少処理が遅くなる。

2.2 計算体の規則的配置による手法

図4(b)に示すように、前述のような行き先テーブルは作らず、行き先を含む計算体を規則的にOMに配置する。また、計算体内での行き先のアドレスの位置を一定にして行き先を計算により求められるようにする。この手法では、行き先と共に計算体も規則的に配置しなければならないので、メモリの断片化が起こる可能性がある。しかし、通信データがダイレクトに行き先に送られるので、前者の手法に比べ高速である。

通常、Valid-Aコンパイラは後者の手法でコンパイルするが、メモリが不足する場合は前者の手法に自動的に切り替える。

3 動的通信のためのValid-Aの拡張

Dynamic Communication on Massively Parallel Image Processor AMP

Noriyasu YAMAMOTO, Naoyuki TSURUTA, Rin-ichiro TANIGUCHI, Makoto AMAMIYA

Department of Information Systems, Graduate School of Engineering Sciences, Kyushu University

超並列マシンのプログラミング言語たるValid-Aでは、独自の計算能力を持ち、お互いに自由度の高い非同期の通信を行う計算体 (Module) により並列処理を陽に記述する (図2参照)。モジュールは、計算体 (body)、他のモジュールへの入出力ポート (inports, outputs)、および履歴依存処理のための内部状態を保持する変数 (states) から成り、関数型言語では表現しにくい履歴処理を表現できる。また、Valid-Aでは超並列処理を記述するため

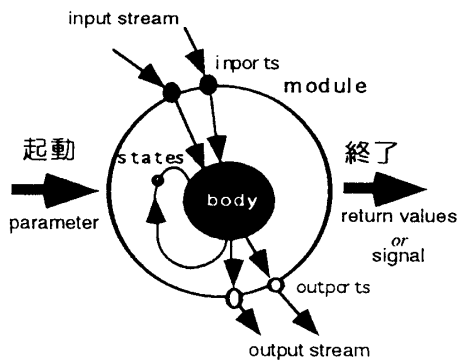


図2 モジュール

にモジュールの並列表現をサポートする。動的な通信は、図3に示したように、あるモジュールのoutputsから複数のモジュールのinportsへの接続で表現する。図4は、前述した2つの手法による入出力ポートの接続方法を示したものである。

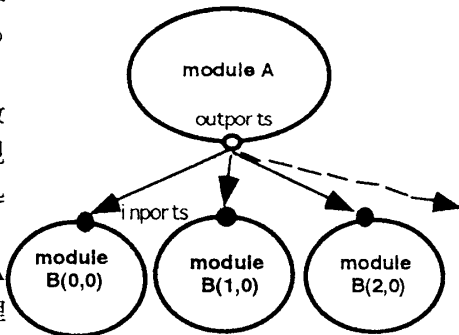


図3 動的モジュール間接続

動的モジュール間接続は、Valid-Aにおいてモジュール配列のインデックスを使用して次のように記述

する。

する。

```
outputs vname->[id_x][id_y]
to module_name[id_x][id_y].target
```

ここでは、あるモジュールの出力ポートvnameが、モジュール配列のインデックスid_x, id_yによってmodule_nameで示されるモジュール配列の入力ポートtargetに動的接続されていることを示す。

4 評価

前述の連結成分のラベル付け処理では、動的通信管理を行う場合は行わない場合に比べて約31%の記憶容量しか費やさない。また、4 PEによる32×32画素の渦巻状図形のラベリング処理時間は、通常0.558[msec]であったのに対して、本稿で提案した動的通信管理を行う手法を用いると約27.6%の0.154[msec]である。行き先が動的に変わる通信において、本稿で提案した手法は、従来の条件式で行き先を決定していた方法に比べて記憶容量の低減と処理の高速化に非常に効果がある。

5 むすび

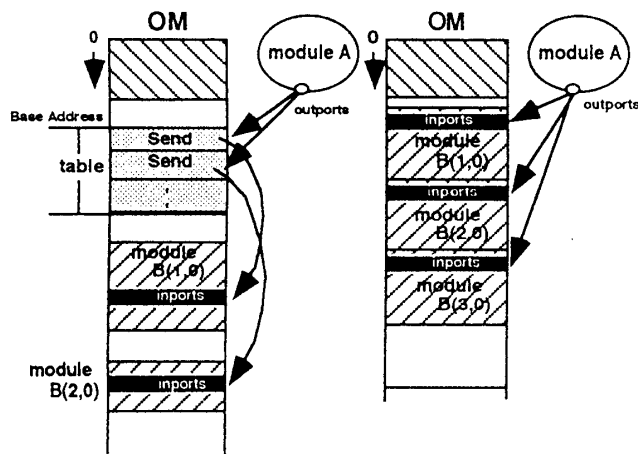
本稿では、AMPにおける効率の良い動的な通信の手法とそれに伴うValid-Aの拡張、およびその評価について述べた。

通信の行き先が動的に変わる画像処理アプリケーションでは、行き先をAMPの各PEに規則的に配置し、計算によって求めた行き先アドレスにデータを送信する手法が効率的である。

本稿で述べた計算体間 (PE間) の動的通信の手法は、計算体内の多分岐処理においても、分岐処理を行き先が動的に変化する通信に置き換えることにより適用でき、効率の良い処理が可能である。

参考文献

- [1] 谷口, 雨宮: “画像処理と理解のための自律型非同期超並列プロセッサAMP”, 「画像理解の高度化と高速化」シンポジウム講演論文集, pp.53-58(1989).
- [2] 山元, 鶴田, 谷口, 雨宮: “画像処理用超並列プロセッサAMPのプログラミングと性能評価について”, 情報処理学会論文誌, Vol.32, No.7, pp.933-940(1991).
- [3] 山元, 鶴田, 谷口, 雨宮: “画像処理用超並列プロセッサAMPにおける連結成分のラベル付けとその評価”, 情報処理学会第46回全国大会講演論文集(2), pp.111-112(1993).



(a) 行き先テーブルを作る手法 (b) 計算体の規則的による手法

図4 動的通信の行き先の規則的配置の手法