

有限要素法解析プログラムの超並列化(1) - 超並列処理用プログラムの検証 -

6 F - 6

磯部俊夫¹⁾・原田公一¹⁾・奥村秀人²⁾¹⁾航空宇宙技術研究所・²⁾神奈川工科大学

1. はじめに

情報処理学会第51回全国大会において、二次元熱伝導解析プログラムを用いて、有限要素法の解析プログラムの超並列化の条件を見出し、その結果について報告した¹⁾。ここでは、その条件を満たす並列処理アルゴリズムによるプログラムの妥当性を超並列計算機 Paragon(米国インテル社製)で検証した結果について述べる。

2. 超並列計算機

本研究では、超並列計算機として、原田らの提案しているSIMD型隣接結合並列計算機²⁾を想定している。この計算機は、独立した演算器・メモリをもつ128×128台の要素計算機が、格子状に配置された構成となっている。

本並列計算機では、要素計算機(以後PEと記す)間のデータ交換は上下左右の4PE間に限定されるが、データ交換におけるオーバーヘッドがないという特徴がある。この計算機の性能評価のためのシミュレータが開発されており、これを用いて並列処理アルゴリズムの性能評価を行っている。

今回、検証に使用した超並列計算機 Paragon は、64bits RISCプロセッサi860XP CPU をPEとし、16×21の336PEから構成されている。PEは、上下左右の隣接した4PEとハードウェアにより結合されている。PE間の通信は、OSによるメッセージ・パッシング方式で行われる。それぞれのPEは、32Mバイトのメモリを

もつ独立した計算機システムを構成していて、ほかのPEとは完全に独立して動作する。

プログラムは通常SPMD(Single Program Multiple Data)で実行される。すなわち、利用するPEに同一のプログラムがロードされ、それぞれのPEは、ロードされたプログラムに従って独立に動作する。

3. 並列処理プログラム

並列化プログラムは、M行N列の格子状に配置されているPEにより、効率的に計算を実行するものとしてコード化されている。隣接結合型並列計算機のシミュレータは、このような構造の超並列計算を逐次型計算機で実行するためのプログラムであり、プログラムの構造は、

```
for (i=0;i<N;i++){
    for(j=0;j<M;j++){
        処理
    }
}
```

となっている。超並列化プログラムは、このような構造で記述できることが条件となっている。

想定しているPE数M×N台の隣接結合型並列計算機が存在すれば、この二重構造のforループを取り除くことができ、計算速度は、PEの台数分のM×N倍となる。

理論上は、台数分だけ計算速度が高まるが、実際の超並列計算機で計算した場合、台数分だけ計算速度が上がるであろうかという疑問が当

Massively Parallel Processing of Finite Element Analysis Program (1)

Toshio Isobe¹⁾・Koichi Harada¹⁾・Hidehito Okumura²⁾¹⁾National Aerospace Laboratory・²⁾Kanagawa Institute of Technology

然出てくる。そこで現在稼働している超並列計算機を使用して、超並列化プログラムを実行し、実験的に検証することを試みた。

4. 検証実験

まず、128×128台のPEを使用するシミュレータ用の有限要素解析プログラムを作成して、その計算結果が逐次型計算機の計算結果と一致することを確認した。

次に、このプログラムをParagonの1台のPEを使い、並列機能を使用せず、逐次型計算機として計算して、その計算時間を測定した。

更に、Paragonの128台のPEを用いて行った。すなわち、隣接結合型並列計算機の1列128台のPEをParagonのPE1台に割り当てる。したがって、検証実験でのプログラムは、forループが1重の次の形式となる。

```
for(j=0;j<128;j++){
```

```
    処理
```

```
}
```

本プログラムでは、PE間において

- ・ 左右上下のPEとのデータ交換
- ・ 各PEのある変数の値の総和を計算

の相互作用がある。

検証実験では、上下のPEとのデータ交換は必要なく、左右のPEとのデータ交換だけとなる。計算中に左右のPEとのデータ転送が必要になったとき、メッセージ・パッシングによるデータ転送の処理を追加することで、隣接結合型並列計算機をシミュレーションすることができる。したがって、基本処理は、

```
for(j=0;j<128;j++){
```

```
    処理
```

```
}
```

左右PE間のデータ転送

となる。想定の際接結合型並列計算機では、このようなデータ転送は必要はない。したがって、検証実験では、このデータ転送処理は、シミュ

レーションのためのオーバーヘッドとなる。

全PEのある変数の総計算は、システムがもっている変数の総計算のための関数を使用した。128台のPEは、全く独立には計算を実行するが、左右PE間のデータを転送するとき、および変数の値の総和を計算するときに、それぞれのPEは同期が取られる。

以上の方法で、超並列化プログラムから書き換えた検証実験プログラムを128台のPEで実行した。その結果は、次のようになった。

プログラム	計算時間	(a)/(b)
(a)PE 1台	3956.2秒	
(b)PE 128台	29.9秒	131.9

128台の計算では、データ転送のオーバーヘッドのため計算速度は128倍以下になるはずであるが、これを上回る結果となった。

5. 考察と結論

本プログラムでは、(1)計算、(2)データ転送、(3)総和計算の部分から構成されている。これらの部分の処理時間を測定してみると計算処理で128倍以上の高速化が達成されている。これに対して考えられる理由としては、キャッシュの効果がある。Paragonでは、各PEに16Kバイトのキャッシュをもっている。本計算方式では、128台の計算機を使うことにより、各PEの使用メモリ容量は1/100程度に軽減される。この使用メモリの軽減により、キャッシュのヒット率が高くなったためと考えられる。

以上、前報で提案した超並列化の手法は、原田の隣接結合型並列計算機に対するものであるが、市販の超並列計算機においても有効に機能することがわかった。

[参考文献]

- 1)原田他、熱伝導有限要素法プログラムの超並列化条件、情報処理学会第51回全国大会、1995.
- 2)原田公一、並列計算機のアーキテクチャシミュレータ、航空宇宙技術研究所TM-583、1988.