

3P-4

階層記憶型並列処理

橋本伸, 長倉浩士, 岡田信, 神谷幸男

富士通 (株)

1. はじめに

通商産業省工業技術院大型プロジェクト「科学技術用高速計算システムの研究開発」の一環として、高速演算用並列処理装置(HPP; High-speed Parallel Processor)を構築した。このハードウェア上で科学技術計算を並列に実行するために作成したソフトウェアシステムの構成、並列処理言語Philなどについて報告する。

2. 背景

科学技術計算分野での高速・大容量コンピュータに対する需要は、非常に大きい。しかし、素子技術や実装技術の進歩では、この需要に応えられるだけの性能を与えることができない。現在、演算性能を向上させる方法として、密結合型マルチプロセッサ(TCMP; Tightly Coupled Multi-Processor)が実現されている。しかし、密結合型並列処理では結合できるプロセッサの数に制限があり、主記憶容量の拡張も難しい。プロセッサ数をさらに増やすために、様々な結合方法が提案されている。

当プロジェクトでは、高速なプロセッサエレメント(PE; Processor Element, 従来のベクトルアーキテクチャのスーパーコンピュータ)を共用記憶装置(CSU; Common Storage Unit)で結合した、HPPと呼ばれるシステムを作成した。このシステムは、従来のTCMPの限界を超えて、より多くのプロセッサを結合するために、局所記憶(各PEの主記憶)と共用記憶とを分離した階層記憶型マルチプロセッサ(HCMP; Hierarchical-memory Connected Multi-Processor)を実現したものである。TCMPを結合してHCMPを構成することもできるので、HCMPではTCMPに比べて1けた上の数のプロセッサを結合することが可能である。

3. 課題

TCMP上の並列処理は商用機でも実現されつつあるが、一般に普及するレベルには到っていない。

この原因は主にソフトウェアにある。ひとつは、効率良く並列処理を制御するソフトウェアの開発が難しいことで

ある。もうひとつは、並列実行するためのアプリケーションプログラムの開発が難しいことである。

これに加えて、HCMP上の並列処理においては、データの割当や転送などの階層記憶管理や、共用記憶アクセスが主記憶よりも低速であることによるオーバーヘッドなどの問題を解決しなければならない。

4. ハードウェアシステムの構成

HPPシステムは、図1のように4台のPEが共用記憶制御装置(CMU; Common Mapping Unit)によってCSUに結合された構造を持つ。PEは、PE記憶(PES; PE Storage)とスカラプロセッサ(SU; Scalar Unit)とベクトルプロセッサ(VU; Vector Unit)から構成されている。HPPシステムは大容量高速記憶(LHS; Large High-speed Storage)と接続されているが、ここでは省略している。

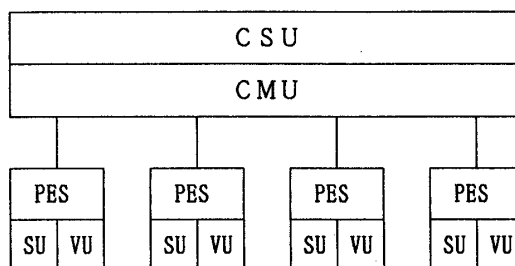
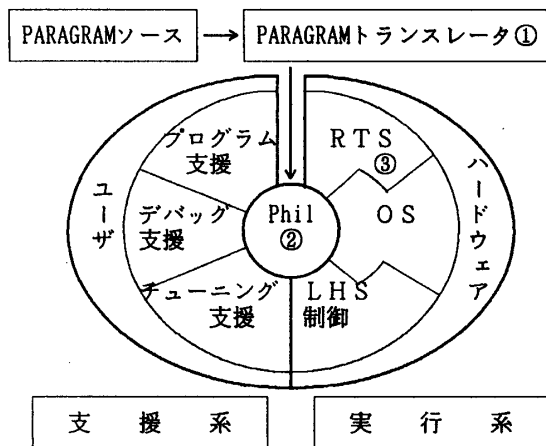


図1 HPPシステムのハードウェア構成

5. ソフトウェアシステムの構成

アプリケーションプログラム開発の困難さを解決するために、ソフトウェアシステムを、言語系・実行系に支援系を加えた構成とした。言語系は並列処理言語PARAGRAPH, Philとそのコンパイラである。実行系は、階層記憶管理ライブラリや並列処理管理ライブラリなどの制御用ソフトウェアを指す。支援系は、プログラム開発支援ツールの総称である。アプリケーションプログラム開発における支援ツールが重要であるとの観点から、言語系と実行系は、支援系とのインタフェースを重視して設計した。図2にソフトウェアシステムの構成図を示す。



- ① PARAGRAM: 問題向き並列記述言語
 - ② Phil: 手続き向き並列記述言語
 - ③ RTS: 並列処理管理と階層構造記憶管理を行う実行時システム
- 図2 HPPソフトウェアシステムの構成

6. 並列処理言語Philの特長

言語Philは、階層記憶記述と並列処理記述を持つ科学技術計算向けの言語である。

Phil言語のプログラムは、並列処理のソース上での単位である複数のクラスタと共有変数で構成されている。各クラスタは、クラスタ内共有変数と1個のクラスタ主手続き、複数の副手続きから構成される。Phil言語の構造を図3に示す。

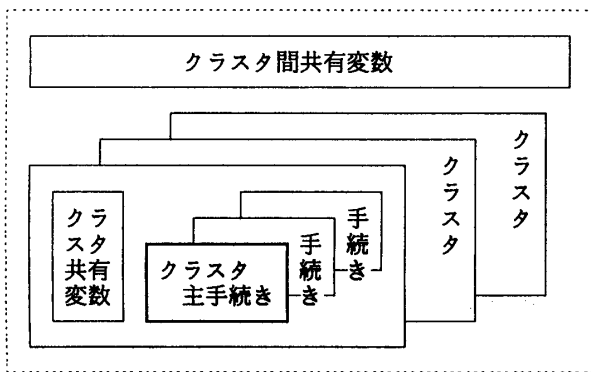


図3 Phil言語の構造

Phil言語の変数は、near又はfarの領域属性を記述することができる。

Phil言語では、繰り返し型並列処理と列挙型並列処理を記述できる。また、lock, barrier, post/waitの3種類の同期機構が準備されている。Phil言語の構文の例を図4に示す。

```

pardo I=1,N      lock LockVar
  call subclust(I,A)  s=s+a(i)
                    end lock
par
  call clust1(A1,A2) barrier
  call clust2(A3)
  call clust3()      wait EventVar
end par
                    post EventVar
    
```

図4 Phil言語の構文例

7. 成果

原子力コードSPIN, 土木工学コードNIEL, 円周率計算, 行列計算等のプログラムをPhil言語で記述し、並列実行に成功した。

4台のPEを結合して、2.1倍から3.8倍の台数効果を得た。

8. 終わりに

HCMPアーキテクチャのシステムでアプリケーションプログラムが並列に実行できたことは、さらに高速な計算の第一歩となるものと確信している。ユーザプログラムを容易に高速実行できる環境を開発したいと考えている。

謝辞 本研究は、通商産業省工業技術院大型プロジェクト「科学技術用高速計算システムの研究開発」の一環として、新エネルギー・産業技術総合開発機構(NEDO)から委託を受けて、実施したものである。

御協力頂いた、日本原子力研究所物理部 別役主任研究員、日本原子力研究所東海研究所計算センター 横川氏、東京大学大型計算機センター 金田助教授、京都大学工学部土木工学教室 渡邊教授に感謝する。

【参考文献】

加藤, 他: 階層記憶型並列処理の制御ソフトウェア, 情報処理学会第41回全国大会, 1990
 渡辺, 他: 並列処理におけるプログラム開発支援システム, 情報処理学会第41回全国大会, 1990