5F−3

超並列システム用OS「超流動OS」の基本概念と実行制御のためのアクティビティの基本設計

田沼均　平野聡　須崎有康

電子技術総合研究所

1 はじめに

PE数が百万台規模の超並列計算機用のオペレーティングシステム「超流動OS」の開発を行なっている[1]。本論文では基本概念とその構想および実行制御の基本設計について述べる。

既にCM-5やParagonといった百台から数万台規模のPEを有する並列計算機が商用化されている。しかしそれらのシステムでは、OSが存在せず計算のみを行なうバックエンド計算機であるか、OSを有してもPE空間をパーテションに区切り、パーティション毎に異なるプログラムを動作させている。この方法は並列度や規則性が変化していくプログラムには不適当であり、遊んでいるPEが多くなりスループットも犠牲になる(図1)。

しかし超並列システムを有効に利用するにはプログラムを複数同時に実行し、各々のプログラムの性質に応じた最適な並列度での実行と時間空間の効率の良い分割共有を可能とする必要がある。

2 超流動OSの目的

「超流動OS」の目的は、百万台規模のPEを有する超並列計算機上で様々なパラダイムに基づいたプログラムを複数同時に動作させ、個々のプログラムが高速に実行可能な環境を提供すると共にシステムの資源を有効に共有し利用させシステム全体が高いスループットのもとに動作するように管理することである。

汎用の超並列システム上で動作するプログラムは、大規模数値計算などで用いられるデータパラレルや、並列オブジェクト指向のようなコントロールパラレルなど様々な形態をとると考えられる。超流動OSではその形態を以下の諸性質により特徴づけ、プロセスアトリビュートと呼ぶ。

粒度 プロセス中の並列処理の単位の大きさ。

規則性 プロセスの空間的トポロジ、時間的依存性、並列度の変化など。

大きさ プログラムやデータの全体の規模。

純度 あるアクティビティがどれだけメモリあるいは二次記憶中のデータと関わっているか。副作用のない関数の計算だけなら最高の純度を有する。

生存時間 プロセスやアクティビティの生成、消滅の頻度

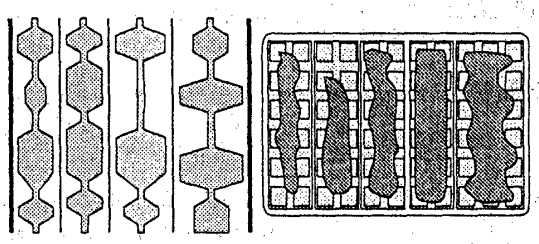

図1: パーティションによる管理

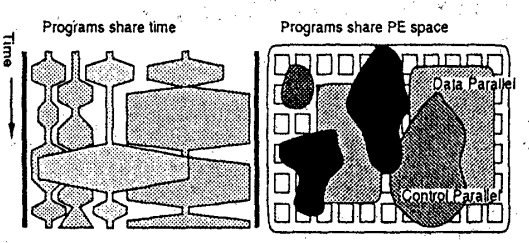

図2: 時間と空間を分割共有する超流動OS

これらの様々な特徴を有する並列プログラムがシステム内で複数存在し、資源を共有しあって同時に実行可能とすることが必要である。固定的なパーティションを用いると図1のように最大使用のために確保はされてはいるが実際にはほとんど使われない資源を多く残してしまう。「超流動OS」では超並列システムの情報を流動させ、並列度の変化や同期待ちによるアイドルなPEなど、即ち時空間上にできる不活性な資源を有効に活用することにより高性能なシステムの実現を、目指す(図2)。図3のように複数プログラムを実行することによりスループットの向上が期待できる。

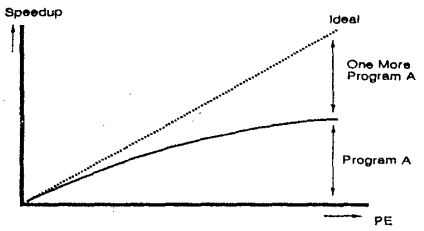

図3: 複数プログラムの実行によるスループットの向上の可能性

"Fluid",An OS for Massively Parallel systems: its concept and basic design of activity management
Hitoshi TANUMA,Satoshi HIRANO,Kuniyasu SUZAKI
Electrotechnical Laboratory

3 超流動OSの基本要素

超流動性の実現のため、現在、以下に挙げる基本要素の開発を行なっている。

2レベルスケジューリング 空間的に規則性の高いプロセスをPE空間内で同期実行するグローバルスケジューラとPEのアイドルタイムを有効に活用するローカルスケジューラ

アクティビティの動的な配置,再配置 PEの時間的実行状況を考慮して動的にアクティビティの配置、再配置し、PEの共有を行なう。

大域的仮想仮想記憶 大域的に実記憶、仮想記憶の管理を行ない仮想記憶の性能の大幅な向上を図る[2]。

アルゴリズム・アダプテーション(AASM) 与えられたデータやシステムの実行環境に合わせて自律的にプログラムの処理形態を適応させていく[3]。

高信頼性化 超並列システムの高速計算性能やPEの冗長性を利用してシステムのフォルトトレランス性を確保する。

4 超並列システムにおけるプロセスとアクティビティ

超並列システムでは1つの仕事が多数のPEを使用して並列に実行される。プログラムの実行制御の単位としてプロセスとアクティビティの二つを導入する。

アクティビティ OSが実行を管理するプログラムの最小単位である。アクティビティはお互いに並列に実行される可能性はあるが、アクティビティ内部ではプログラムは並列には実行されない。またアクティビティは1つのPEの中に存在し、PEをまたがることはない。1つのPEの中には仮想プロセッサ機能により複数のアクティビティが存在することができる。

プロセス OSが管理する一つの論理的な仕事の単位であり、ある計算なり仕事なりを行なうプログラムの実行イメージである。複数のPEにまたがり、内部はそれぞれ並列に動作するアクティビティにより構成される。プロセスは前述のようにプロセスの性質を示すプロセス・アトリビュートを有する。プロセス・アトリビュートはAASMや言語処理系などより与えられ、この情報をもとにOSはスケジューリングや空間配置の戦略を決定する。

5 スケジューリング

データパラレルのプログラムのような規則性の高いプログラムを効率良く実行するためには、各アクティビティを同期して実行する必要がある。各PEにおいてプロセスやアクティビティのスケジューリングを行なうためにプログラムの空間的配置を考慮しなければならない。

スケジューラはPE間での同期を加味したグローバル・スケジューラとローカル・スケジューラの二つにより構成される。

グローバル・スケジューラ グローバルスケジューラはプロセスアトリビュートに基づき同期性の高いプロセスがPE空間内で同一時刻のタイムスライスに割り当てられるようスケジューリングを行なう。各タイムスライスに割り当てられた1つのプロセスをそのタイムスライスにおけるプライマリ・プロセスと呼ぶ。

ローカル・スケジューラ プライマリプロセスは同期や通信により待ち状態になることがある。このアイドル時間を利用するのがローカル・スケジューラである。ローカル・スケジューラには空間的な同期性の不要なアクティビティが登録され、プライマリ・プロセスが待ち状態になったら実行の割り当てを行なう。プライマリ・プロセスの待ち状態が解消後、直ちにプライマリ・プロセスのアクティビティが実行を再開する。

6 空間実行制御

PE空間の有効な利用を図るためプロセスやアクティビティの動的な再配置を行なう。プロセスの平行移動とプロセス内部のアクティビティの移動、即ちプロセスの変形を行なう。

プロセスの平行移動 実行中のプロセスをプロセス内部のアクティビティの位置関係を保存したままPE空間内を平行に移動させる。新しいプロセスの発生やプロセスの大幅な変形が行なわれてPE空間に大きな領域が必要となった際に行なう。プロセス内でのアクティビティの相対的な位置関係は保存されるためデータパラレルのようなトポロジ依存性の強いプログラムに対しても行うことが可能である。

プロセスの変形 プロセス内の一部のアクティビティの位置を少し移動させる。比較的小さな空間が必要となった時やコントロールパラレルなどのトポロジ依存性の低いプログラムを、アイドル時間の多いプライマリプロセスの動いているPE上で動作させる際に行なう。

再配置の時期と位置はプロセスの中のAASMからの要求と、スケジューラが管理するランキューの長さの空間分布により決定する。

7 おわりに

ここでは開発中の「超流動OS」の基本構想と実行制御の基本設計について述べた。現在、本システムは実際の超並列システム上での動作を実現すべく詳細な設計を行なっている。

謝辞

本研究の一部はRWC計画の一環として「超並列システムアーキテクチャに関する研究」で行なわれたものである。関係各位に感謝いたします。

参考文献

[1] 平野, 田沼, 須崎, 浜崎, 塚本. 超並列システム用オペレーティングシステム「超流動OS」の構想. 情報処理学会オペレーティング・システム研究会資料,1993.

[2] 平野, 田沼, 須崎. 超並列システム用OS「超流動OS」における仮想記憶管理の方式. 情報処理学会第46回全国大会.1993

[3] 須崎, 栗田, 田沼, 平野. 超並列システム用OS「超流動OS」におけるアプリケーションソフトウェアの動的最適化. 情報処理学会第46回全国大会.1993