

プロセスのスケジューリングアルゴリズムの性能について

日大生産工(院) ○山本 翔
日大生産工 松田 聖

1 はじめに

近年、多くの学術機関や企業を中心にマルチユーザーシステムが普及している。マルチユーザーシステムとは複数のユーザーが同時にログインして環境を利用することを前提としたシステムである。このようなシステムで実装されているプロセススケジューリングはシングルユーザーシステムと変わらない。その理由は、このプロセススケジューリングがプロセスの公平性を重視した設計になっているため、利用するユーザー数に関係なくスケジューリングを行うことができからである。その一方で、様々なレベルのユーザーが存在するようになり、現在では多種多様のソフトウェアが利用されている。このような環境において、CPUの占有時間のみで公平性を判断していても良いスケジューリングは行えない。つまり従来のスケジューリング方式に加えて、新たにユーザー間のCPU占有率の公平性を保証するスケジューリングを導入する必要がある。

本研究ではCPUの占有時間をユーザー単位で平均化することができるフェアシェアスケジューリング方式の性能について調べる。具体的にはLinuxカーネルの一部を変更することで、既存の機構や機能をできるだけ変えることなく、フェアシェアスケジューリングを実現する。そして従来のスケジューリングと性能を比較し、既存のスケジューリングに対する改善の可能性を検証する。

2 Linux

1991年、Linus Torvalds氏によって開発された、UNIX互換のOS。フリーソフトウェアとして開発が行われており、現在では世界的に利用されるOSとなった。本来、OSの基盤となる「カーネル」のみを指す呼称であるが、今日ではLinuxディストリビューションを指して、単にLinuxと呼ぶこともある。

2.1 Linuxカーネル

カーネルはC言語やアセンブリ言語で書かれており、それらのソースコードをコンパイルするためのコンパイラにGCCを使用している。Linuxカーネルはマルチタスクの仕組みを提供しており、複数のプロセスを細かく切り替えながら動作させることで、いかにも同時動作しているような環境を作り出している。マルチタスク環境を提供するOSであれば、ほぼ同様の仕組みで実現されている。この処理を行う機能をプロセススケジューラと呼ぶ。

3 RRスケジューリング

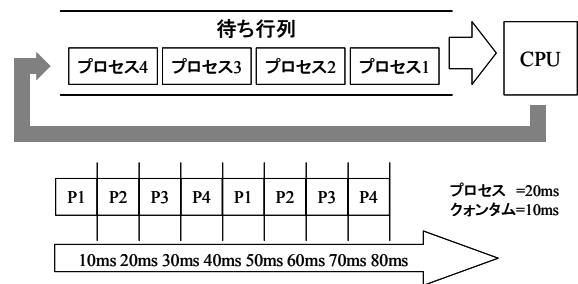


図1 RRスケジューリング

図1はRRスケジューリングの待ち行列と実行順序を示している。RRスケジューリングは最も基本的なスケジューリングアルゴリズムである。このスケジューリングでは、まず各プロセスにクォンタム(一定の時間)が与えられる。そのプロセスがクォンタムの終了時にまだ実行中であれば、CPUは別のプロセスに与えられる。この時CPUを横取りされたプロセスは、プロセス待ち行列最後尾に置かれる。

実際のLinuxでは、優先順位に基づくスケジューリングと組み合わせることで、待ち時間の長いタスクを実行させることを可能にしている。

4 マルチユーザーシステムにおける問題点

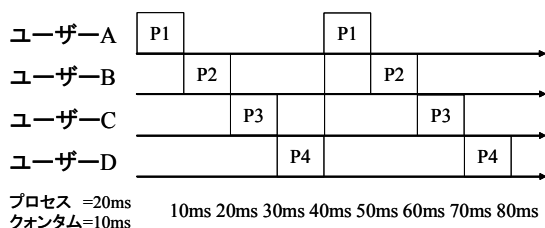


図2 CPU占有の遷移図(user4:process4)

図2は四人のユーザーがそれぞれ同じプロセスを実行したときの各ユーザーがCPUを占有する遷移を示している。このときの平均ターンアラウンド時間は65(ms)である。

図2に新しくユーザーEが他と同様のプロセスを実行したときのCPU占有の遷移を図3に示す。

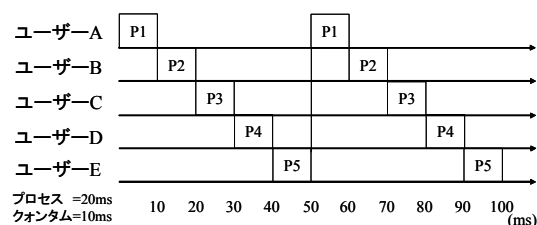


図3 CPU占有の遷移図(user5:process5)

このとき求められる平均ターンアラウンド時間は80(ms)であり、ユーザーが増えることでプロセスが増えるため各ユーザーのターンアラウンド時間も増加していることがわかる。

ここでユーザーが多様化している状況を想定するために、プロセス5(P5)をユーザーDが実行したとする。そのときのCPU占有遷移の様子を図5に示す。

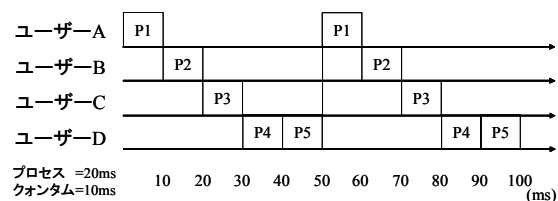


図4 CPU占有の遷移図(user4:process5)

このときに求められる平均ターンアラウンド時間は77.5(ms)であり、5ユーザーの時よりも短い。しかし、4プロセスの時と比べ場合、同じプロセスを実行しているにもかかわらず、ユーザーA, B, Cのターンアラウンド時間が長くなっている。このことから、プロセスを重視する現在のスケジューリングでは、他ユーザーよりも多くのプロセスを持っているユーザーの方がCPU占有の機会を多く得てしまう。

5 提案するスケジューリング

5.1 フェアシェアスケジューリング

ユーザー間でCPU占有の機会を公平にするために、フェアシェアスケジューリングを考える。フェアシェアスケジューリングとは、ユーザー毎のCPUの占有率を公平に設定し、それに従ってスケジューリングする方式である。子のスケジューリングにおいて同じユーザーを所有者とするプロセスが複数存在する場合は、そのプロセス群にユーザーのCPU時間を分配する。

5.2 フェアシェアスケジューリングの検証

問題となる4ユーザー:5プロセスの状態に、フェアシェアスケジューリングを適用する。

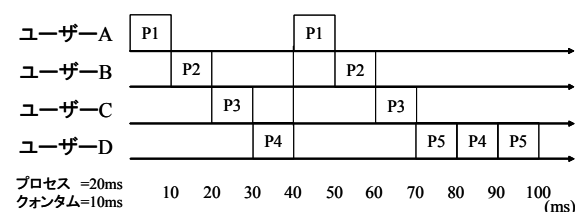


図5 フェアシェアスケジューリング

図5でフェアシェアスケジューリングにおけるCPU占有の遷移の様子を表している。4ユーザー:5プロセスのときのRRスケジューリングの様に似ているがユーザーDのプロセスに対するCPUの占有の遷移が変わったことに気づく。このときの平均ターンアラウンド時間は70(ms)とプロセスを基準に考えたスケジューリングのときよりも短くなった。なによりユーザーA, B, C, それぞれに対するターンアラウンド時間が、4ユーザー:4プロセスの時と同じになったことから有用性があると考えられる。

6 まとめ

本研究ではプロセスのスケジューリングアルゴリズムの性能について、従来のスケジューリングとは異なる、ユーザーを基準としたスケジューリングについて有用性について検証している。今後はこのスケジューリングを実装したLinux上での検証を考えている。

「参考文献」

- 1) ゲーリー・ナット, 「実習Linuxカーネル, 理論と実習 カーネルを効率的に理解するための実習書」, ピアソン・エデュケーション, (2001)
- 2) 大久保 英嗣, 「オペレーティングシステムの基礎」, サイエンス社, (1997)