

# マイクロカーネル Lavender における多段階スケジューリング機構

5 Z-4

毛利 公一† 佐脇 秀登† 芝 公仁† 豊岡 明† 大久保 英嗣††  
 †立命館大学大学院理工学研究科 ††立命館大学理工学部情報学科

## 1 はじめに

現在、マイクロカーネルが持つ柔軟性と拡張性が注目されている。従来のカーネルは、カーネルのサイズが大きく、カーネル内の機能を容易に変更できないなどの問題点を持つ。また、現在では、アプリケーションの多様化により、個々のアプリケーションが OS に要求する機能も多様化している。

これらを背景に、我々はユーザカスタマイズ可能なマイクロカーネル Lavender を構築している。Lavender のスケジューラは、アプリケーションに応じた柔軟なスケジューリングアルゴリズムを提供するための機構として、多段階スケジューリング方式を採用している。本スケジューラは、マスタスケジューラとユーザスケジューラとから構成される。ユーザは、ユーザスケジューラとして様々なスケジューリングアルゴリズムを構築することができる。

以下、本論文では 2 章で Lavender の概要、3 章で多段階スケジューリング機構について述べ、4 章でスケジューラの実装と評価について述べる。

## 2 Lavender の概要

Lavender は、メモリ管理、プロセス管理、スケジューリング、割り込み管理、プロセス間通信の機能を持つ。また、Lavender は以下に示す特徴を持つ。

- ・ユーザカスタマイズ可能
- ・ポリシーとメカニズムの分離
- ・階層化インタフェース
- ・クロスアドレススペースコールのオーバーヘッド軽減

Lavender は、サーバによるユーザカスタマイズ性を可能にするために、ポリシーとメカニズムの分離の概念を採用している。ポリシーとは、ある処理を遂行するための手法や手順を決定するものである。メカニズムとは、ポリシーの決定に従って実際に処理を行うものである。Lavender では、ポリシーとメカニズムの分離を階層化インタフェースによって実現している。階層化インタフェースでは、Lavender の持つ機能をニュークリアス層、カーネル層、システム層の 3 つの層に分類している。

Lavender では、プロセスグループ機能及びレジデントアドレス空間を用いることによって、プロセス間通信

Multi Stage Scheduler in Lavender Micro Kernel  
 Koichi Mouri, Hideto Sawaki, Hirohito Shiba, Akira Toyooka and Eiji Okubo  
 Department of Computer Science, Faculty of Science and Engineering, Ritsumeikan University

やプロセス間手続き呼び出しなどの際に発生する、データのコピーやアドレス空間の切り替え、コンテキストの切り替えによるオーバーヘッドを軽減させている [1]。

## 3 多段階スケジューリング機構

Lavender のスケジューラは、アプリケーションに応じた柔軟なスケジューリングアルゴリズムを提供するための機構として、多段階スケジューリング方式を採用している。

### 3.1 プロセス・スレッドモデル

Lavender はプロセス・スレッドモデルを採用している。

- ・プロセス … 1 つ以上の仮想アドレス空間に属し、1 つ以上のセグメントに分割された実行環境である。
- ・スレッド … レジスタセットとスタックを持ち、プロセスが提供する環境の中で動作する実行実体である。スレッドはスケジューリングの単位である。スレッドは 1 つのプロセス内に複数あってもよい。

### 3.2 スケジューリング機構の概要

Lavender のスケジューリング機構は、マスタスケジューラとユーザスケジューラから構成される (図 1 参照)。

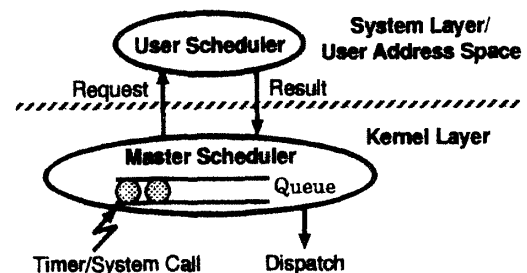


図 1: スケジューリング機構の概要

#### (1) マスタスケジューラ

マスタスケジューラは、カーネル層に属し、システムに唯一である。Lavender のスケジューリング機構においてメカニズムを実現する部分であり、スケジューリングアルゴリズムなどのポリシーを持たない。スケジューラによってその内容が設定される 1 つのキューを持つ。マスタスケジューラは、タイマ割り込み、システムコール、及びカーネル内からの関数呼び出しによって起動される。マスタスケジューラのインタフェースを以下に示す。

- ・初期化
- ・キューの操作
- ・スケジューリング情報の提供
- ・スケジューラへの処理要求の受け付け
- ・スレッドへのCPUの割り当て

マスタスケジューラの持つキューは、ユーザスケジューラによって操作される。キューではスレッドの割り当て順序を管理するとともに、スレッドのスケジューリング情報へのポインタを管理する。マスタスケジューラは、キューで示される順序とスレッドのスケジューリング情報を参照し、それによってスレッドにCPUを割り当てる処理を行う。マスタスケジューラを持つスケジューリング情報は、スレッドのID、属性、実行開始時間、タイムスライスである。

## (2) ユーザスケジューラ

ユーザスケジューラは、Lavenderのスケジューリング機構においてポリシーを実現する部分であり、デフォルトスケジューリングサーバとしてシステム層に属するか、またはユーザプロセスとして実現される。ユーザスケジューラでは、スケジューリングアルゴリズムが実装され、実際にスケジューリングを行う。ユーザスケジューラは、マスタスケジューラによって呼び出され実行される。このとき、マスタスケジューラから処理要求を格納した要求パケットを受け取る。ユーザスケジューラはその要求パケットに従い処理を行う。要求パケットの内部には、処理要求コードとパラメータが格納されている。スケジューリングを行う際には、必要な情報をカーネルから取得し、スケジューリングの結果はマスタスケジューラを持つキューに反映させる。

## 4 スケジューラの実装と評価

Lavenderでは、前章までに述べた機構を用い、システム層のユーザスケジューラとしてラウンドロビンとレートモニタリングスケジューリング(RMS)が実現されている。本章では特にRMSの実装について述べる。RMSを実現したユーザスケジューラは、初期化、スケジューリング、スレッドの追加・削除・状態変更の機能を持つ。

新たにスレッドが起動されると、スレッドの登録要求がユーザスケジューラに対して行われる。登録要求時には、パラメータとして、スレッドの起動時刻、周期、処理時間がユーザスケジューラに渡される。ユーザスケジューラは、これらのパラメータを基にスケジューリング可能かどうかを調べる。スケジューリング可能であれば実行可能キューに追加し、スケジューリング不可能であればスレッドの起動処理は中断される。

ユーザスケジューラは、スケジューリング要求が来ると、まず、既に終了しているスレッドを調べる。前回起動時刻と周期の和が現在時刻と等しいか大きいスレッドが存在すれば、そのスレッドを起動し、実行可能キュー

に追加する。次に、実行可能キューを調べる。実行可能キューが空であればアイドル状態となる。実行可能キューにスレッドが存在する場合は優先順位順にキューをソートした後に、最も優先順位の高いスレッドに対してCPUを割り当てる。

上記ユーザスケジューラを、PC/AT互換機(Pentium MMX 200MHz)上で動作させ、評価を行った。以下に示す条件で、スレッド数を1から16まで変化させ、周期スレッドの起動遅延を計測した(図2参照)。

- ・タイマ割り込みの間隔は100 $\mu$ s
- ・マスタスケジューラの起動間隔は10ms
- ・スレッドの周期は200ms, 処理時間は10ms

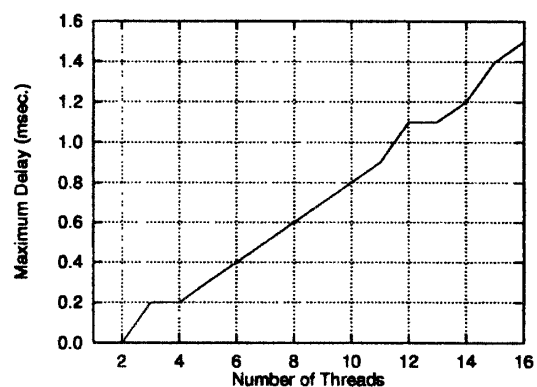


図2: 周期スレッドの起動遅延

図2に示す結果は、スレッドの周期と処理時間を変えて測定した場合にも同様の結果が得られた。スレッドの起動遅延は、ユーザスケジューラのオーバヘッドによって発生する。このオーバヘッドには、スケジューラの呼び出しなどにかかる一定のオーバヘッドと、既に終了したスレッドの周期を調べるためのオーバヘッドや、実行可能キューを優先順位でソートするオーバヘッドなど、スレッド数に関するオーバヘッドがある。これらのオーバヘッドの削減は今後の課題である。

## 5 おわりに

本論文では、マイクロカーネルLavenderにおけるスケジューラの構成と、RMSの実装について述べた。多段階スケジューリング機構を用いることで、リアルタイム・非リアルタイムを問わず、ユーザが容易にスケジューリングアルゴリズムを変更・構築することが可能となる。

今後は、複数のスケジューリングアルゴリズムを同時に動作させるための機構の実装・評価を行う予定である。

## 参考文献

- [1] 芝 公仁, 佐脇 秀登, 豊岡 明, 毛利 公一, 大久保 英嗣: “マイクロカーネル Lavender の構成”, 情報処理学会研究報告 97-OS-75, pp. 7-12, 情報処理学会 (1997).