

Title	分散リアルタイムシステムにおけるスケジューリングに関する研究
Author(s)	馬場, 久
Citation	
Issue Date	2004-03
Type	Thesis or Dissertation
Text version	author
URL	http://hdl.handle.net/10119/1773
Rights	
Description	Supervisor:田中 清史, 情報科学研究科, 修士

分散リアルタイムシステムにおける スケジューリングに関する研究

馬場 久 (210072)

北陸先端科学技術大学院大学 情報科学研究科

2004 年 2 月 13 日

キーワード: 分散リアルタイム, リアルタイムスケジューリング, タスク割り当て, 負荷分散.

1 はじめに

リアルタイムシステムにおいて, 要求される時間制約内にタスクを処理することが重要である. タスクのデッドライン (deadline, 締切時刻) を守るために従来から様々なスケジューリングアルゴリズムが提案されている.

一方, 近年の急速な技術革新により, かつては実現が困難であった分散リアルタイムシステムの構築が可能となってきた.

従来から研究されてきた単一プロセッサ上でのスケジューリングと比較すると, 複数のプロセッサをターゲットとしたスケジューリングでは最適性を求めることが困難である [3]. 本稿では, 分散システムにおいて, タスクの静的優先度と余裕時間を考慮したタスク割り当ての手法を提案する.

2 リアルタイムスケジューリング

一般に, リアルタイムシステムでは優先度に基づくスケジューリングが行なわれる. このとき, 設計者が事前に静的優先度を決定する方法と, タスクの属性に応じて優先度を決定する方法とがある.

後者の方法については様々なアルゴリズムが提案されている. 単一プロセッサ上のスケジューリングでは, 例えば RM (Rate Monotonic) や EDF (Earliest Deadline First)[2] などが代表的なアルゴリズムである. これら二つを含むいくつかのアルゴリズムは, それぞれ特定の条件下で最適なスケジューリング法であることが証明されている.

一方, マルチプロセッサ上のスケジューリングの場合, オンラインスケジューリングで最適なアルゴリズムは存在しないことが証明されている. また, オフラインスケジュー

リングについても、多くの条件下で NP 完全問題であることが知られている [3]. 加えて、単一プロセッサ上のスケジューリングにはなかったマルチプロセッサ特有の問題がある。したがって、実際には 発見的 (heuristic) 手法が利用される。

単一プロセッサ上のスケジューリングならば比較的 簡単なので、タスクの割り当てと各プロセッサ上でのスケジューリングを別々に扱う方法がある。

3 負荷の算出

タスクを割り当てる際には最も負荷の小さなプロセッサを選択するが、このとき負荷の指標として何を採用するかが重要である。従来は、CPU 使用率、タスクの数などが広く用いられてきた。本稿では、適応型動的優先度法 [1] の概念を取り入れ、負荷の指標に次のパラメータを導入することを提案する。

- タスクの静的優先度
- タスクの余裕時間

負荷 ρ は次式により算出される。

$$\rho_i = \frac{1}{M_i} \sum_{k=1}^N \left(\frac{P_k}{P_{lv}} + \frac{c_k}{X_k} \right)$$

ここで、 M はプロセッサの相対能力、 P はタスクの静的優先度、 P_{lv} は優先度レベル、 c はタスクの残り実行時間、 X はタスクの余裕時間、 N はそのプロセッサに割り当てられているタスク数である。算出した各プロセッサの負荷に基づいてタスク割り当てを行ない、負荷分散によるデッドラインオーバーの減少を図る。

4 評価

全タスクの起動要求を受け付け、プロセッサに割り当てるグローバルスケジューラと、複数のプロセッサから成る分散システムを想定する。本研究ではスケジューリングシミュレータを実装し、様々な条件の下でシミュレーション評価を行なった。主な条件を以下に示す。

負荷の指標

プロセッサを選ぶ際の負荷の指標として、タスク数、CPU 使用率、CPU 負荷率、提案方式から選択する

スケジューリングアルゴリズム (タスクの優先度付け法)

グローバルスケジューラ、また各プロセッサのスケジューラが実行タスクを選ぶアルゴリズムとして、静的優先度に基づく方法・FCFS・EDF・LLF・RM・DM から選択する

各オーバーヘッド

スケジューリングと負荷計算、通信にはそれぞれオーバーヘッドが発生する。それぞれの方式に応じたオーバーヘッドを加味した。オーバーヘッドの有無を選択する

他に、プロセッサの数やその相対能力を選択する。

スケジューリング手法の性能を評価するスケジューリング関数には様々なものがあるが、本シミュレーションでは、デッドラインオーバー数、平均応答時間、そして新たに提案する遅れタスクの静的優先度平均をもって評価を行なった。

5 おわりに

本研究では、分散リアルタイムシステムにおけるオンラインスケジューリングを対象とし、タスク割り当てのための新たな負荷の指標を提案した。負荷の指標に適応型動的優先度法の概念とタスクの余裕時間を導入し、またプロセッサ能力の違いを考慮した。

シミュレータを実装して様々な条件でシミュレーションを行ない、結果の傾向を分析した。プロセッサ間の通信時間が十分に短い場合、提案手法によって、静的優先度が高いタスクのデッドラインオーバーを減少できることが分かった。

今後の課題としては、さらに多くの条件でシミュレーションを行ない、さらに他の評価関数で結果を比較することなどが挙げられる。

参考文献

- [1] 栗谷一路, “リアルタイム OS における優先度を用いた動的スケジューリングの研究,” 北陸先端科学技術大学院大学 情報科学研究科 修士論文, Mar. 2003.
- [2] C. L. Liu and J. W. Layland, “Scheduling Algorithms for Multiprogramming in a Hard-Real-Time Environment,” *Journal of the ACM*, Vol. 20, No. 1, pp. 46–61, 1973.
- [3] J. A. Stankovic, M. Spuri, M. D. Natale, and G. C. Buttazzo, “Implications of Classical Scheduling Results for Real-Time Systems,” *IEEE Computer*, Vol. 28, No. 6, pp. 16–25, June 1995.