

# デッドラインを考慮したメモリ削減スケジューリング LMCLF の改善

組込みデザイン研究室 新井 諒介 (指導教員 中田 明夫)

4.  $\alpha$  の条件を満たす  $\alpha$  を出力

## 1 はじめに

スマートフォン,家電製品,医療機器などの組込みシステムは大量生産されることが多いため,製造コストの削減は重要な課題であり,メモリ消費量を削減することが,組込みシステムの開発目標の一つとして挙げられている[1].そこで我々は,マルチタスクシステムのヒープメモリ消費量を動的解析によって削減するスケジューリングアルゴリズム Least Memory Consumption First (LMCF) スケジューリングを提案した[2]. 先行研究では,消費メモリ増分だけでなく,実時間制約を共に考慮した Least Memory, remaining Computation-time, and Laxity First (LMCLF) スケジューリングが提案されている. 本研究では,優先度をつける段階の計算の手法を改善することで従来手法よりもメモリ消費量を削減しようとするものである.

## 2 提案手法

LMCLF: 消費メモリ増分だけでなく,実時間制約を共に考慮したスケジューリング手法  
ここに本研究の LMCLF のアルゴリズムを示す.

1. 2 ステップ後までのメモリ消費量 4 通りを求める

$m[i][j]$ : 1 ステップ目に実行されるタスクを  $i$ , 2 ステップ目に実行されるタスクを  $j$  としたスケジュールで実行したと仮定した場合のメモリ消費量

2. 4 つの中で最悪メモリ消費量の小さい  $\alpha$  の条件を求める

1 で求めたメモリ消費量が最小のスケジュールを

LMCLF で実現させるような  $\alpha$  の条件

3. 求めた  $\alpha$  の条件を満たす  $\alpha$  が存在するまで以下を繰り返す

3. 1. 次に最悪メモリ消費量の小さい  $\alpha$  の条件に更新する

従来では,その換算レート  $\alpha$  の値が任意で与えられていたが事前に求める必要があった.本研究では,その  $\alpha$  の値の最良な値の導出法を提案する.これを行うことで,  $\alpha$  を事前に定める必要が無く,かつ,  $\alpha$  の値が最良でない場合の従来手法よりもメモリがより削減される(従来手法でも  $\alpha$  がもし最良に設定されていたらメモリ削減量は等しい)。

## 3 実験

実験目的: 最良な換算レート  $\alpha$  の模索のため

実験方法: 現段階では 2 タスク 1 プロセッサの環境下で  
タスクセットの値を変えながらいろいろなパターンで求める.方法としては提案手法の LMCLF のアルゴリズムに沿って行う.

## 参考文献

[1] R. Zurawski, "Embedded Systems Handbook, Second Edition: Embedded Systems Design and Verification", CRC Press, 2009.

[2] Y. Machigashira and A. Nakata, "An improved LLF scheduling for reducing maximum memory consumption by considering laxity time", In Proc. of 12th Int. Symp. on Theoretical Aspects of Software Engineering, pp.144-149, IEEE Computer Society Press, 2018.

[3] M. L. Dertouzos and A. K. Mok, "Multiprocessor On-Line Scheduling of Hard-Real-Time Tasks", IEEE Tran. on Software Engineering, Vol.15, No.12, 1989. [4] C. L. Liu and J. W. Layland, "Scheduling Algorithms for Multiprogramming in a Hard-Real-Time Environment",

Journal of ACM, Vol. 20, No. 1, pp.46–61, 1973. [5] J. Lee, A. Easwaran, and I. Shin, “Laxity Dynamics and LLF Schedulability Analysis on Multiprocessor Platforms”, Real-Time Systems, Vol. 48, Issue 6, pp716–749, 2012.

[6] T. P. Baker, “Comparison of Empirical Success Rates of Global vs. Partitioned Fix-Priority and EDF Scheduling for Hard Real Time”, Technical Report TR-050601, Department of Computer Science, Florida State University, pp.1–14, 2005. [7] J. Lee, “Time-Reversibility for Real-Time Scheduling on Multiprocessor Systems”, IEEE Transactions on Parallel and Distributed Systems, Vol. 28, No. 1,