

リアルタイムシステムにおける QoS 公平化適応制御器の実装

学籍番号：90153148 潮 研究室 満岡咲枝

1 序論

リアルタイムシステムにおいて、過負荷状態はシステム性能の低下を引き起こすため、タスクに割り当てられたサービス品質 (Quality of Service, QoS) に応じてタスクの CPU 利用率を下げ、ジョブをリリースすることで回避する必要がある。しかし、リソースの配分によってはタスクの QoS レベルに偏りが生じる。本報告では QoS レベルの公平化を達成しながらシステムの過負荷状態を回避する QoS 公平化適応制御器を、リアルタイムシステム環境上で実装し、その手法の有効性を実証する。

2 QoS レベルの公平化

本報告ではフィードバック制御理論を用いて QoS レベルを公平化する QoS 公平化適応制御を用いる。互いに独立な周期タスク集合 $\{T_1, T_2, \dots, T_N\}$ を考えるとき、総 CPU 利用率を R に保ちながらすべてのタスクの QoS レベル Q_i が等しくなるようにタスクに配分する CPU 利用率 r_i を制御するのが QoS 公平化適応制御の目的である。制御器の l 回目の起動時刻を t_l とおき、時刻 t_l におけるすべてのタスク間での QoS レベルの平均値を $\bar{Q}(t_l)$ とすると、 $r_i(t_l)$ は次式によって決定される。

$$r_i(t_l) = r_i(t_{l-1}) + \alpha(\bar{Q}(t_{l-1}) - Q_i(t_{l-1})) \quad (1)$$

ここで α はフィードバックゲインであり、この α に関して次の定理が成り立つ。

定理 1 (QoS レベル公平化可能条件)

$$\alpha \leq \frac{1}{\max_i D_i}$$

であるとき、タスク集合 $\{T_1, \dots, T_N\}$ の QoS レベルの公平化が可能である。ただし、 D_i は r_i に対する Q_i の最大変化率である。

3 QoS 公平化適応制御システムの実装

QoS 公平化適応制御を用いて、異なる QoS 関数を持つ複数の画像トレースタスクのリソース配分を行う。リアルタイム性を実現するため、本報告ではリアルタイム OS として CPU 予約機能を持つ TimeSys Linux を採用した。CPU 予約を用いることで、他のタスクの状態に関係なく一定の CPU の割り当てが保証される。QoS 公平化適応制御機構は図 1 のように構成した。

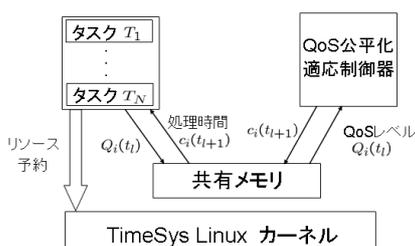


図 1 QoS 公平化適応制御システム

制御器は QoS レベルを受け取ると制御則 (1) に従って各タスクに配分する CPU 利用率 $r_i(t_{l+1})$ を計算する。タスク T_i の周期を τ_i とおくと、リソースとして CPU の処理時間 $c_i(t_{l+1}) = r_i(t_{l+1})\tau_i$ を配分していることになる。

各タスクは、画像内を 1 フレーム毎に移動する点を前フレームの位置から時計回りに探索するジョブを周期的にリリースし、予約で与えられた処理時間 $c_i(t_l)$ 以内にトレースを終了することができた割合を QoS レベル $Q_i(t_l)$ として評価する。トレース対象となる点の 1 フレーム毎の移動の最大変位を d とし、 d を異なる値に設定することでタスク毎に異なる QoS 関数を持たせることが可能となる。 $d(T_1) = 20, d(T_2) = 40, d(T_3) = 60, d(T_4) = 80, d(T_5) = 100$ と変化させたときの各タスクの予約処理時間に対する QoS レベルの変化と、それから得られる QoS 関数は図 2 のようになった。したがって $\max_i D_i = D_1 = 8.08384 \approx 8.1$ となる。

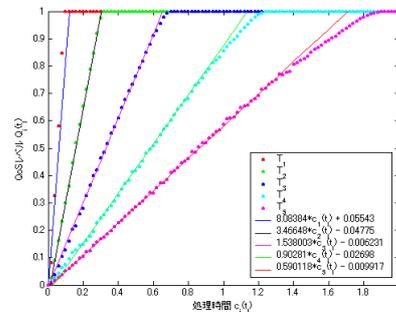


図 2 QoS 関数

4 実験

タスク数 $N = 5$ 、総 CPU 利用率 $R = 0.1$ 、ゲインパラメータ $\alpha = 1/8.1 \approx 0.12$ として実験を行うと、制御器起動回数 l に対する QoS レベル $Q_i(t_l)$ の変化は図 3 のようになり、すべてのタスクの QoS レベルが同じ値に収束していること、つまり QoS レベルの公平化が達成されていることが確認できた。

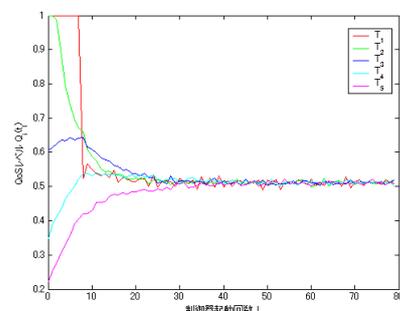


図 3 QoS レベルの時間応答

5 結論

本報告では、QoS 公平化適応制御をリアルタイムシステム環境上で実装し、その手法の有効性を実証した。