

3N-5 動的 QoS 制御に基づく資源管理システム『堤』における資源確保手法

田中 大介[†] 芝 公仁[†] 大久保 英嗣^{††}

[†]立命館大学大学院理工学研究科 ^{††}立命館大学工学部情報学科

1 はじめに

近年、計算機の高性能化に伴い、動画や音声等の連続メディアデータを扱うマルチメディアアプリケーション (MM-AP) が多く開発されている。さらに、インターネットが爆発的に普及し、ストリーミング技術やネットワークの帯域制御技術に大きな関心が集まっている。計算機上で連続メディアデータを扱うためには、システムが各メディアのもつ時間的制約を満たしながら処理を行う必要がある。そのためには、処理のために十分な資源 (CPU, ネットワーク, メモリ等) が確保されなければならない。従来のシステムのように、best-effort な資源管理では、システム負荷時に AP に対し十分な資源が提供されない。しかし、MM-AP の時間制約がある連続メディアデータの再生処理では、与えられた資源量に応じて動的に処理のサービスの質 (QoS: Quality of Service) を変更していく必要がある。それらの問題を考慮し、現在我々は、MM-AP に有効な動的 QoS 制御に基づく資源管理システム『堤』の開発を行っている [1]。

以下、本稿では、2 章で『堤』の概要、3 章でリソースリザーバで行われる資源確保の手法について述べ、最後に 4 章でまとめと今後の課題を述べる。

2 資源管理システム『堤』

『堤』の構成を図 1 に示す。『堤』は、ユーザ、MM-AP 及びシステムの 3 層に QoS パラメータを与え、その相互変換と QoS 交渉で動的 QoS 制御を実現している。QoS 交渉では、ユーザの意図する QoS を MM-AP に提供する資源量に反映させるために、MM-AP の優先度を用いた市場モデルを導入している。QoS 交渉で得られた資源量は、『堤』の資源確保機構である“リソースリザーバ”によって、CPU, ネットワーク帯域, メモリの各資源において予約される。

また、『堤』は、柔軟な資源管理を可能にするため、MM-AP に QoS 変更の指示を促すことができる“堤ライブラリ”を用意する。堤ライブラリは、MM-AP に『堤』の API として動作する。MM-AP は、ユーザや『堤』から QoS 変更の要求を受け取り、メディアスケールリングによって QoS を変更する。

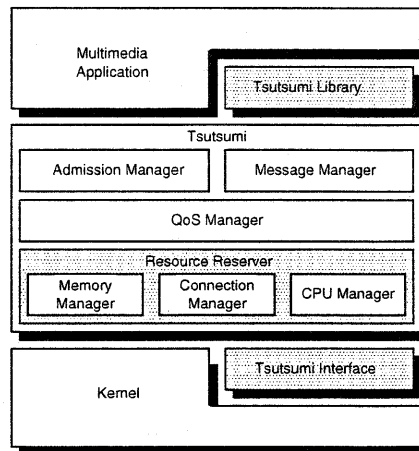


図 1 『堤』の構成

3 資源確保

『堤』では、MM-AP が必要とする資源量を QoS 交渉によって算出する。その後、リソースリザーバによって、MM-AP が利用できる資源を予約し、利用するのに必要な情報を通知して MM-AP に資源提供を行う。『堤』で提供される資源は、CPU, メモリ, ネットワーク帯域の 3 つであり、各資源を確保するマネージャが存在する。

3.1 CPU 資源

MM-AP に提供される CPU 資源は、優先度及びデッドラインによるリアルタイムスケジューリングで確保された CPU 利用率である。この CPU 利用率は、MM-AP の連続メディアデータの再生タスクに与えられるべきである。再生タスクの CPU 利用率を動的に変更するためには、MM-AP 自体によってタスクの負荷を変化させなければならない [3]。したがって、MM-AP は、各連続メディアセッションに周期タスクを生成し、与えられた CPU 利用率で動作するように負荷を定める必要がある。

3.2 CPU 利用率の予測

『堤』では、セッションによる CPU 資源の利用と動的変更のために、QoS と CPU 利用率の変換テーブルを用意してある。一般に、あるタスクの CPU 利用率は、実行時間と起動時間の商で求めることができる。したがって、QoS が変更された時点を生起時間として算出した CPU 利用率とその時の QoS パラメータを変換テーブルに格納する。しかし、選択されたことのない QoS に対応した CPU 利用率は未知数であるため、『堤』は、柔軟な QoS 制御を行うことができない。

そこで、選択中の QoS と CPU 利用率から変換テー

A Resource Reservation Method for Resource Management System “Tsutsumi” based on Dynamic QoS Control

Daisuke TANAKA[†], Masahito SHIBA[†], and Eiji OKUBO^{††}

[†]Graduate School of Science and Engineering, Ritsumeikan University

^{††}Department of Computer Science, Faculty of Science and Engineering, Ritsumeikan University

ブルの CPU 利用率フィールドを予測し完成させることを考える。『堤』では、CPU 利用率 U (%) は、連続メディアデータのビットレート $Bitrate$ (bit/s) に比例すると仮定した。このとき、式 (1) のように比例定数 D を導入する。定数 D は、MM-AP の特性から、計算機の処理能力及び MM-AP の連続メディアデータの処理手順の 2 つの要素から一意に決まると予想される。ある QoS における $Bitrate_n$ は、QoS パラメータである時間的解像度と空間的解像度の積で求めることができる。よって、式 (2) のように任意の QoS における U_n を求めることができる。

$$U = Bitrate \times D \quad (1)$$

$$U_n = Bitrate_n \times D \quad (2)$$

この予測方法は、判断材料が非常に少ないため、誤差が大きいことが分かっている。そのため、予測された CPU 利用率は、QoS 変更時に計測されたものと置き換えられるべきである。

3.3 メモリ資源

メモリ資源は、CPU 資源や帯域幅資源に比べ、時間に影響されない固定的な資源である。MM-AP では、再生処理のためにメモリをバッファとして確保する。『堤』では、メモリ資源の提供及び操作のために、各 MM-AP のセッションとの共有メモリを利用している。共有メモリは、スワッピングによる遅延を回避すべきである。そのため、『堤』では、`mlock()` システムコールを使って領域のスワッピングを禁止している。MM-AP は、バッファ領域にアクセスするために、堤ライブラリの一部である表 1 の API を用いる。

3.4 帯域幅資源

帯域制御は、帯域確保とデータの同期が必要となり、端末またはルータが帯域制御をしなければならない。帯域確保は、CPU 資源のようにパケット送信のスケジューリングで実現できる。例えば、Linux Kernel では、プロトコル毎に送信時の帯域幅を制限する機能があり、管理者の決めたポリシーでパケットを送信できる。『堤』は、Fast Ethernet 上での End-To-End の帯域制御を目的とし、送信パケットの制限、帯域確保に必要な交渉プロトコル、ストリームの同期プロトコルによって帯域幅の管理を行う (図 2)。交渉プロトコルは、受信側主導型と送信側主導型があり、ストリームの用途によってどちらかに決めることができる。このプロトコルによって、必要な帯域幅やコネクションのポートが通知され、交渉の結果で利用できる帯域幅を決定することができる。また、同期プロトコルは、ストリーム同士のリンクを考慮した同期や送受信データの同期を実現する。送受信間で相対的な時間系を用い、ストリームデータに時間情報を付加することで、同期を行う。

3.5 堤インタフェース

現在我々は、これらの資源確保手法を評価するために、『堤』を Linux 上に実装している。しかし、Linux Kernel

表 1 メモリ利用の API

| Interface | 説明 |
|-------------------------|---------------|
| MemControl (id,arg,...) | メモリ資源を操作する |
| MemMap (*addr,len,*id) | メモリ領域をマップする |
| MemFree (id) | メモリ領域を解放する |
| MemResize (id,len,opt) | 領域の大きさを変更する |
| MemHandler (opt) | 『堤』からの指示を取得する |

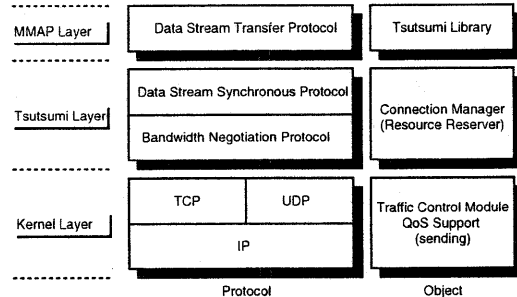


図 2 帯域制御プロトコル

で提供される機能だけでは、各資源の確保と提供を実現できない。そこで、『堤』に必要な機能を Linux Kernel に追加し、それらを“堤インタフェース”としてまとめている。堤インタフェースのシステムコールは、すべてリソースリザーバから呼び出される。追加すべき主な機能は、EDF スケジューラ及び周期タスクの登録、CPU リザーブ機能、任意のポートにおけるネットワーク帯域幅の監視などである。CPU リザーブ機能は、RT-Mach で実装されている機能であり、あるタスクの CPU 利用率の上限を定めてそれを保証する [2][4]。

4 おわりに

本稿では、資源管理システム『堤』の概要と資源確保の手法について述べた。『堤』は、QoS 交渉と資源確保の機能を用いて、連続メディアセッションにおける QoS パラメータからそれに見合った資源量を確保することができる。今後は、CPU リザーブ機能と帯域制御プロトコルの詳細設計を行い、『堤』全体の評価をする予定である。

参考文献

- [1] 田中 大介, 芝 公仁, 大久保 英嗣: “動的 QoS 制御に基づく資源管理システム『堤』の設計と実装,” 第 60 回 (平成 12 年前期) 情報処理学会 全国大会講演論文集 (1), pp. 141-142 (2000).
- [2] 河内谷 清久仁: “マルチメディア処理の動的 QoS 制御のためのフレームワーク,” 電子情報通信学会論文誌 B-I, Vol. J80-B-1, No.6, pp. 465-471 (1997).
- [3] 酒井 靖夫, 大久保 英嗣: “マルチメディア端末における適応的映像フロー制御方式,” 情報処理学会論文誌, Vol.41, No.9, pp.2455-2466 (2000).
- [4] C.W. Mercer, S. Savage, and H. Tokuda: “Processor Capacity Reservers: Operating System Support for Multimedia Applications,” Proc. IEEE ICMCS '94, pp.90-99 (1994).