

帯域予約されたネットワーク上の連続メディア送信

岡村 耕 二[†] 荒木 啓二郎[†]

本論文では、資源予約技術のひとつであるネットワークの帯域予約技術に着目し、帯域が予約されたネットワーク上での連続メディア送信の問題について論じる。帯域予約されたネットワークでは予約した帯域は保証されるが、その一方で予約した帯域は越えることはできないという制限がある。そのため、パケットベースの送信では平均レートに比べて送出レートが高くなるため、送出レートを考慮して予約する帯域を決定する必要がある。しかし、送出レートは送出方法によって変化するため、最適な帯域を決定するのは困難である。そこで、本論文では、品質の指定された連続メディアの送信を送出スケジューリングとあわせてモデル化し、そのモデルに基づいて連続メディアデータの送出レートを最小にする送出スケジューリングを決定する方法を提案する。次に、本論文で提案する送出スケジューリングを、オペレーティングシステムの実時間的なスケジューリングを用いてアプリケーションに提供する方法を提示する。最後に、資源予約できるネットワークを用いて測定を行い、本論文で提案した方法で与えられた連続メディアの品質から送出レートを最小にする送出スケジューリングを求めることができることをその測定値に基づいて示す。

Continuous Media Transmission on Bandwidth Reserved Network

KOJI OKAMURA[†] and KEIJIRO ARAKI[†]

The main subject of this paper is the problem and its solution of continuous media transmission of bandwidth reserved network. On the bandwidth reserved network, although the reserved bandwidth is guaranteed, bandwidth over reservation can never be used. On the packet based communications by Internet-Protocol, the transmission rate is higher than average rate because a packet data pours into network for a short time. Although the reservation bandwidth must be considered with transmission rate on the packet based communications, the transmission rate varies by different transmission scheduling. In this paper, we make a model of continuous media communication with the transmission scheduling. Then we propose a method which can decide the transmission scheduling with lowest rate for given resolution of continuous media based on this model. Then we show how to offer the transmission scheduling to the applications by real time scheduling of operating systems. Finally we demonstrate that the lowest transmission rate and its scheduling can be decided for given resolution of continuous media by measurement value.

1. はじめに

近年のネットワークおよび計算機の高速化によって、計算機ネットワークを利用した音声や動画などを統合したマルチメディア通信を行うことが可能になってきた。遠隔地間でマルチメディアデータの送受信が可能になることによって、地理的に分散している人間の間でのコミュニケーションを支援する実用的な環境の実現に対する期待が高まっている。

現在、インターネット上で音声や動画といった連続メディアをその品質を保証しながら送信するために、その伝送路の資源を予約するプロトコルの研究^{1)~5)}

や、予約されたネットワーク上での連続メディア通信の研究⁶⁾が行われている。また、ATM (Asynchronous Transfer Mode) の帯域予約機能を利用して、ATMの上にインターネットプロトコルを実装する研究も行われており^{7),8)}、ATMをデータリンク層として利用している試験的なインターネットは日本にもすでに存在している⁹⁾。本論文ではATMによる帯域予約技術に着目し、ATMを用いて資源予約することのできるインターネット上での連続メディアの送信について論じる。

インターネットプロトコルを用いた通信では、データの送受信はパケット単位で行われる。そのため、データが伝送路に送出されたときのレートは平均レートよりも高い。本論文では、従来のEthernetやFDDIといった共有型のネットワークを利用する場合には問題

[†] 奈良先端科学技術大学院大学情報科学研究科
Graduate School of Information Science, Nara Institute
of Science and Technology

とならなかった平均レートと送出レートの差による影響が、帯域予約されたネットワークを利用する場合には問題となることに着目している。

共有型のネットワークでは、ある送信者がデータを送出している間は、その送信者がそのネットワークのデータリンクのすべての帯域を占有することができるため、送出レートが平均レートよりも高いことは問題にならない。これに対して、帯域予約が可能なネットワークでは予約した帯域を越えて通信を行うことはできない。そのため、従来の共有型のネットワーク上で開発された通信ソフトウェアのように、送出時に均一なレートで送信する機構がない場合は、帯域予約されたネットワークへの送出時に予約した帯域を越えてしまうためデータの破棄が発生する。ネットワークインタフェースカードのバッファで送出時のデータの破棄を防ぐことは可能であるが、連続メディアデータの大きさがネットワークインタフェースカードのバッファよりも大きい場合は、データの破棄をバッファで防ぐことは不可能となる。

このように、連続メディアを帯域予約して送信する場合、ネットワークの帯域は送出レートで予約する必要があるが、同じ平均レートの連続メディアでも送出スケジューリングが異なると送出レートが異なるため、送出レートを一意に決定することが困難であるという問題がある。そこで、我々は、連続メディアの送出レートを、その送出スケジューリングと組にして定義すれば、送出スケジューリングを決定することで送出レートを正確に決定することができることに着目した。本論文では、連続メディア送信をその送出スケジューリングとあわせてモデル化し、そのモデルに基づいて、連続メディアの送出レートの最小値と、そのときの送出スケジューリングを決定する方法を提案する。

本論文は以下のような構成である。2章では、従来の帯域予約のできないネットワークと帯域予約のできるネットワークの違いから生じる帯域予約されたネットワーク上での連続メディア送信の問題点を述べる。3章では連続メディアの送信をその送出スケジューリングとあわせてモデル化し、そのモデルに基づいて与えられた連続メディアの送出レートを最小にする送出スケジューリングと、その送出レートを求める方法を提案する。4章では、本論文で提案する手法をアプリケーションから利用する方法について説明する。5章では、本手法を用いて具体的に連続メディアの品質から最適な送出スケジューリングとそのときの送出レートを求める例を示す。6章では結論を述べる。

2. 帯域予約可能なネットワークの利用における問題点

本章では帯域予約が可能であるネットワーク上で、連続メディアを送信する場合の問題点を述べる。

現在利用されている帯域予約のできないネットワークでは、複数のアプリケーションがネットワークの帯域を共有して通信を行っている。そのため、あるアプリケーションがネットワークの帯域を占有することを防ぐために、大きなトラフィックを発生させる可能性があるアプリケーションには、データ送信の平均レートを制御する機能が必要である。

現在のインターネット上には Mbone と呼ばれるマルチキャスト技術の実験用のネットワークが存在している¹⁰⁾。Mbone ではマルチキャスト技術と並行して、このマルチキャストネットワーク上で音声や動画といった連続メディアを扱うことのできる多くのアプリケーションも研究開発されている。そのような連続メディアを扱うアプリケーションには、送信の平均レートを制御する機能によって、Mbone 上で送信されている連続メディアでインターネットの帯域が占有されない工夫が施されている。Mbone で開発された連続メディアを扱うアプリケーションは、インターネット上でのテレビ会議などを可能にするため、インターネット上でのヒューマンコミュニケーションを行うためのツールとして注目されている¹¹⁾。

しかし、そのようなアプリケーションを帯域予約可能なネットワークで使用する場合、その平均レートで帯域を予約したネットワーク上で連続メディアデータを送信するとデータの破棄が発生する。我々は、Mbone で開発された動画の送受信ができるツールを用いて、資源予約されたネットワーク上でテレビ会議を試みたが、動画データがネットワークで破棄されることなしに送信するためには、そのツールで設定する平均レートよりもかなり高い帯域を予約する必要がある。

連続メディアアプリケーションが平均レートを制御しているにもかかわらず、帯域予約されたネットワークでデータの破棄が発生するのは、送出レートが平均レートに比べ高いからである。インターネットプロトコルを利用したパケットベースの送信では、パケット単位でデータを送出するため、送出レートは平均レートよりも高い。しかし、共有型ネットワーク上では、送信者はデータの送出時にそのネットワークのデータリンクのすべての帯域を瞬間的に占有することができるため、送出レートの影響は問題にならない。これに対して、帯域予約しているネットワークで、送出レート

が予約している帯域を越えてしまうと、データの破棄が発生する。このとき、アプリケーションの平均レート制御の機能は、単位時間あたりに送信するデータの量の制御が行えるだけであるため、このような送出時に発生する高レートをおさえることはできない。そのため、帯域予約されたネットワークでは平均レートの制御機能の効果は期待できない。

また、ネットワークインタフェースカードなどのバッファはこの送出時に発生する高レートの影響をある程度吸収することができるが、ネットワークインタフェースに標準的に装備されているバッファの大きさは、通常はそのインタフェース全体の容量の単位時間分であり、帯域予約したアプリケーションが利用できるバッファとしては、予約した帯域の単位時間分の大きさのみが割り当てられる。そのため、送出データがそのバッファよりも大きい場合は、バッファオーバーフローが発生しデータの破棄が発生する。

結局、帯域予約されたネットワーク上で連続メディアを送信するためには、平均レートではなく送出レートで帯域を予約する必要があることが分かる。しかし、同じ平均レートの連続メディアでも送出スケジューリングが異なれば送出レートも異なるため、送出レートを一意に決定できる送出スケジューリングを定義する必要がある。そこで、本論文では連続メディア送信をその送出スケジューリングとあわせてモデル化し、そのモデルに基づいて送出レートを最小にする送出スケジューリングを決定する方法を提案する。

3. 送出スケジューリングと送出レート

本論文は、アプリケーションとして、連続メディアを扱うアプリケーションを想定している。そのため、以降では特に明記しない限り、連続メディアを扱うアプリケーションを単にアプリケーションと表記する。

本章では、まず、帯域予約の可能なネットワークの説明を行い、データの送出スケジューリングによって送出レートが異なることを示す。次に連続メディアの送信を送出スケジューリングとあわせてモデル化し、そのモデルに基づいて送出レートを最小にする送出スケジューリングを求める方法を提案する。最後に本手法の現実性について述べる。

3.1 帯域予約の可能なネットワーク

連続メディア送信について議論する前に、帯域予約の可能なネットワークについて説明する。ネットワークの帯域予約はアドミッション・コントロールとモニタリングによって実現することができる。これらの機能を簡単に説明する。詳細は文献[12]を参照されたい。

アドミッション・コントロール

アドミッション・コントロールは、つねに予約している帯域の総和がそのネットワークの容量を超えないことを保証する。もし、新規の予約によって、予約している帯域の総和がネットワークの容量の帯域を越えてしまう場合はその予約を拒否するか、現在予約されている帯域と折衝を行う。

モニタリング

モニタリングは、アプリケーションの使用レートが予約した帯域を越えていないかを監視する。モニタリングは予約した帯域を越えたデータを破棄、または遅延させることによって、アドミッションコントロールで予約した帯域の保証を行う。

モニタリングは、たとえばATMスイッチではUPC (Usage Parameter Control) 機能によって実現されている¹³⁾。ネットワークはUPCによって、帯域分割され、各アプリケーションからは自らが予約した帯域のみが利用可能となる。本論文では、このようにして帯域予約されたネットワーク上での連続メディア送信について議論する。

3.2 連続メディアデータの送信

連続メディアの品質は時間的解像度と空間的解像度で定義できる¹⁵⁾。連続メディア処理では空間的解像度に基づいた連続メディアデータが時間的解像度に基づいた周期時間で連続的に処理されている。本論文では、時間的解像度を周期時間、空間的解像度をその周期時間内で処理されるデータ量に対応させて議論する。連続メディアの送信を行う場合、アプリケーションから渡されてきたデータをオペレーティングシステム（本論文では以降OSと表記する）がシステムコールで送信する。帯域予約の可能なネットワークでは、使用レートの監視のためにスライディング・ウィンドウや、リーキーバケットと呼ばれる手法を用いている^{13),14)}。これらの手法では、ある決められた単位時間で送信されたデータ量を測定することにより帯域を監視している。一般的に、監視の単位時間は連続メディアの周期時間に比べて非常に小さい。そのため、1周期分のデータを短時間でまとめて送出したときに発生する高レートは、トラフィックモニタによって予約した帯域を越えたレートと判断される可能性がある。このデータの破棄はネットワークインタフェースカードのシェーピング機能で防ぐことができる。シェーピング機能とはネットワークインタフェースカードでアプリケーションから渡されたデータをいったんバッファに貯めて、予約したレートで送出する機能である¹³⁾。しかし、現在のすべてのATM用のネットワークイン

タフェースカードにシェーピング機能が付いているとは限らず、また、シェーピング機能がついている場合でも、そのバッファはインタフェースの全帯域の単位時間分の容量だけが用意されているため、帯域予約したアプリケーションが使用できるバッファの大きさは、予約した帯域の単位時間分の容量のみである。つまりバッファは、予約した帯域の単位時間分の容量を超えるデータ量には対処することができないため、たとえば動画の1画面分のすべてのデータといった大量のデータが一度にネットワークインタフェースカードに渡されると、バッファはオーバーフローし、結局データ破棄が発生することになる。

以上のことから、連続メディアのような大量のデータを送信する場合、個々のシステムコールで渡されるデータ量とシステムコールの間隔から計算される送出レートが最小になるように送出スケジューリングを行い、そのときの送出レートを予約する必要があることが分かる。

さて、同じ量のデータを複数のシステムコールに分割して送信するとき、その送出スケジューリングは、1回のシステムコールで送信するデータの大きさと、システムコール間隔によって定義することができる。しかし、データ量とシステムコールの間隔によって送出レートは異なる。

次の3.3節では、連続メディア送信を、そのデータの送出スケジューリングとあわせてモデル化し、与えられた連続メディアの品質から、その連続メディアの最小送出レートおよびその送出スケジューリングを決定する方法を提案する。

3.3 連続メディア送信モデルと送出スケジューリング

周期時間 P (sec)、1周期時間で処理する連続メディアの量が D (byte) である連続メディアの送信を行うとき、 D の送出スケジューリングによって送出レートは変化する。連続メディアの送信を、その送出スケジューリングとあわせてモデル化した図を図1に示す。連続メディアの品質から、周期時間 P および、周期時間内に処理するデータ量 D は一意に決まり、1回のシステムコールで送信するデータ量 d (byte) と、システムコール間隔 t (sec) によって、送出スケジューリングは一意に決まる。

1回のシステムコールで送信するデータ量が少なく、またシステムコールの間隔が広ければ、送出レートは低くなる。しかし、システムコールで送信するデータ量を減らすと、システムコールの回数が増え、システムコール間隔が短くなる。一方、1周期時間で処理す

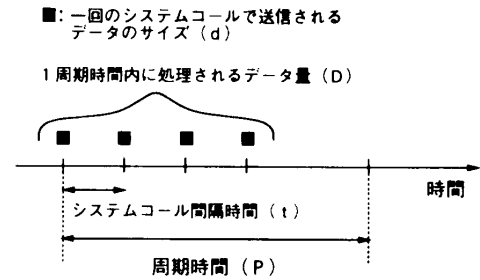


図1 連続メディア送信のモデル

Fig. 1 Model of continuous media transmission.

るデータ量 D の送信は周期時間以内に行う必要があるため、一定のシステムコールの回数以上には分割することができない。これらのトレードオフを考慮して、送出レートを最小にする1回のシステムコールで送信するデータ量 d とシステムコール間隔 t を決定する必要がある。

本論文では、あるデータ量 d をあるシステムコール間隔 t で送信しているときの送出レート $B(d, t)$ が分かれば、 $B(d, t)$ を用いて周期時間 P 、1周期時間内に処理されるデータ量が D である品質の連続メディアデータの送出レートを最小にする送出スケジューリングを決定することができることを示す。

システムコールの呼び出しに必要な時間は、そのシステムコールが送信するデータ量 d に依存するので $T_s(d)$ と表す。周期時間 P をシステムコールを時間間隔 t で割った値と1周期内に処理されるデータ量 D を1回のシステムコールで送信される d で割った値は、どちらも1周期内でシステムコールが呼ばれる回数を表しているため、式(1)が成り立つ。

$$\frac{P}{t} = \frac{D}{d} \quad (1)$$

式(1)からシステムコール間隔時間 t を P 、 D 、および d を用いて式(2)のように表すことができる。

$$t = P \times \frac{d}{D} \quad (2)$$

ただし、1回のシステムコールで送信するデータ量が d のときにシステムコールを呼び出すのにかかる時間は $T_s(d)$ であり、それがシステムコールの最小時間間隔であるため、式(3)の不等式が成り立つ。

$$T_s(d) \leq t = P \times \frac{d}{D} \quad (3)$$

よって、与えられた連続メディアの送出レートの最小値を求めるためには $B(d, P \times \frac{d}{D})$ ($T_s(d) \leq P \times \frac{d}{D}$) の最小値を求めればよいことが分かる。このときの送出スケジューリングは、1回のシステムコールで送信するデータインターネットが d 、システムコール間隔

が t となる。

3.4 実現性に関する考察

本論文で提案した、品質の指定された連続メディアデータ送信の最小送出レートとその送出スケジューリングを求める方法を実現するためには、1回のシステムコールで送信されるデータ量 d とシステムコール間隔 t から決まる送出時の帯域 $B(d, t)$ が既知であるという仮定の正当性と、そのような $B(d, t)$ を求める方法が問題となる。

まず、システムコールで送信されるデータ量 d とシステムコール間隔 t の組合せで決まる送出レートがネットワークインタフェースカードのハードウェアの性能で決定していれば、それよりも低いレートで送出されることはないため、その値を送出レートとして定義することができる。ネットワークインタフェースカードのバッファをオーバーフローさせないようにすれば、この送出レートを求めることができる。

次に、 $B(d, t)$ の d と t の有効な範囲については、OS の送信用のシステムコールで送信できるデータ量の最大値が存在するので、 d の最大値はその OS の1回のシステムコールで送信できる最大値となる。さらに、文献によれば、人間が許容できる遅延時間は数百 msec である¹⁷⁾という報告がなされているので、システムコール間隔は t は1秒以下で十分である。つまり、 d と t はそれぞれ有限範囲内であるのでその範囲で $B(d, t)$ を測定しておけばよいことが分かる。

最後に、測定によって得られる $B(d, t)$ の精度について述べる。OS がデータを送信処理した後はネットワークインタフェースカードのハードウェアがネットワークへ送出し、また、帯域の監視も ATM スイッチ等のハードウェアで行われているため、測定によって得られる値の精度はソフトウェアで行われている OS の送信処理のスケジューリングの精度に依存する。そこで、本論文では資源予約の機能を持ったマルチメディア OS を利用することを前提とする。資源予約機能を持った OS は、アプリケーションから要求されるサービスの品質 (Quality of Service, 以降では QoS と表記する) を満たしてスケジューリングを行うことができる¹⁶⁾。資源予約を行った場合の負荷の影響は、資源予約を要求するときの QoS パラメータのひとつである「揺らぎ時間」に反映することができる。そのため、予測されないような負荷の影響は受けないことを仮定できる。QoS については次の4章で説明する。そこで、 $B(d, t)$ のシステムコール間隔 t の下限を、負荷の影響を考慮した QoS を指定した場合の時間間隔に設定することによって期待したスケジューリングの精

度を得ることができる。

4. アプリケーションからの送出スケジューリングの利用

本章では、本論文で提案する送出スケジューリングの利用方法について、連続メディアを実時間的に入力し送信するアプリケーションを例にして説明する。まず、アプリケーションでの連続処理を入力処理、符号化処理および送信処理に分けてそれぞれの処理の時間的な制約を説明した後、連続メディア処理に対する QoS について説明する。次に連続メディアの送信処理に本論文で提案する連続メディアの送出スケジューリングを利用する方法について説明し、最後に帯域が動的に変化する連続メディア処理への適用方法について述べる。

4.1 連続メディア処理の時間的制約

カメラやマイクなどの連続メディア用の入力装置を備えた計算機上で、実時間的に連続メディアデータを入力して送信するアプリケーションでは入力処理、符号化処理および送信処理が連続的に行われる¹⁸⁾。それぞれの処理について説明する。

入力処理

入力処理では、1周期分の連続メディアデータを取り込む。通常1周期は、動画の場合は、画面1フレームに対応する。音声の場合、その標本間隔は非常に短いため、単位時間分のデータを取り込む必要がある。なお、現在、インターネットで音声を送信するために、音声をキャプチャする単位時間は標準化されている¹⁹⁾。

符号化処理

符号化処理では、入力処理によって取り込んだ連続メディアデータを指定された形式に符号化する。

送信処理

送信処理では、符号化された1周期分のデータを送信する。

アプリケーションは連続メディア処理の各周期で、入力処理、符号化処理および送信処理を、図2に示すような順番で逐次的に開始する。連続メディアデータ

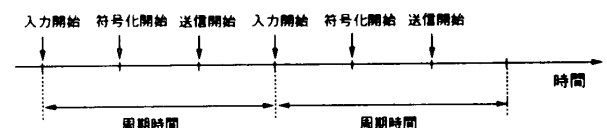


図2 連続メディア処理におけるアプリケーションによる処理の開始

Fig. 2 Start of processing by application in continuous media.

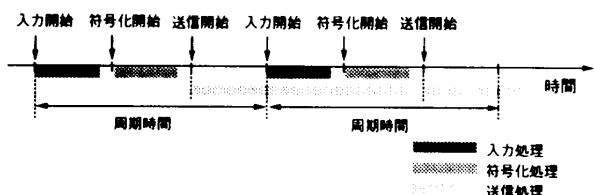


図3 連続メディア処理におけるOSによる処理のスケジューリング

Fig. 3 Scheduling of processing by OS in continuous media.

をなめらかに処理するためには、ある周期内での処理を次の周期の処理に影響を与えずに終了させる必要がある。つまり、図2のようにアプリケーションによって開始される入力処理、符号化処理および送信処理にはそれぞれ時間的な制約がある。

まず、なめらかな連続メディアデータを生成するためには、各周期の入力処理の開始時刻の間隔をつねに周期時間と等しくする必要がある。次に連続メディアの品質に基づいてその帯域を予約しているネットワークでは、ある周期の連続メディアデータの送信を次の周期の連続メディアデータの送信と同時にすることはできない。そのため、ある周期の連続メディアデータの送信処理は次の周期の連続メディアデータの送信処理が開始する前に終了させる必要がある。また、入力処理の終了後開始される符号化処理は、送信処理のための送信時間が確保できるように処理を終了させる必要がある。

以上説明した入力処理、符号化処理、送信処理の各処理の時間的な制約の条件を考慮すると、ある周期の送信処理とその次の周期の入力処理および符号化処理とは、図3で示されるように並列に処理できることが分かる。

4.2 連続メディア処理のQoS

4.1節で説明した、入力処理、符号化処理および送信処理の時間的な制約は、処理の対象である連続メディアデータの特性に起因している。これに対して、アプリケーションを使用しているユーザからの連続メディア処理への要求がある。このユーザからの要求はアプリケーションを通じて、QoSパラメータで表現することができる。典型的なQoSパラメータとして、連続メディアデータの品質、遅延時間、揺らぎ時間、および信頼性をあげることができる。

連続メディアデータの品質

連続メディアの品質は空間的解像度と時間的解像度で定義することができる。3.2節で説明したように、本論文では時間的解像度を周期時間に、空間的解像度を周期時間内で処理されるデータ量に

対応させている。

遅延時間

遅延時間はある連続メディアデータの入力開始から送信終了までの時間である。

揺らぎ時間

アプリケーションの種類によっては、連続メディアの各処理を厳密に指定した時間に終了させる必要がないものもある。揺らぎ時間は、その許容範囲の時間を表す。また適切な揺らぎ時間を設けることによって、負荷の変動による影響を受けにくくすることができる。

信頼性

ある連続メディア処理において、指定された時間的な制約を守ることのできなかった周期の処理は行われない。そのため、その部分は再生されず、一時的に動画や音声の乱れが生じる。信頼性は単位時間中の連続メディアの周期処理の中で失敗した周期の割合で、その連続メディアの処理の乱れを表す。

連続メディアデータの周期時間内で処理されるデータ量によって入力処理、符号化処理、送信処理のそれぞれに必要な処理時間を決定することができる。

これらのアプリケーションから要求されるQoSパラメータの間には依存関係がある。たとえば、揺らぎ時間を大きくすると、負荷の変動によって発生する遅延時間の影響を受け難くなるため、信頼性は高くなるけれども、遅延時間は長くなる。また、遅延時間を短くすると各処理の揺らぎ時間は短くなり、負荷の変動によって発生する遅延時間の影響を受けやすくなるために信頼性が低くなる。

4.3 送信スケジューリングの利用方法

図2で示される各処理の開始はアプリケーションによって行われ、図3で示される処理のスケジューリングはOSによって行われる。連続メディア処理の送信処理に本論文で提案する送信スケジューリングを利用するためにQoSとして連続メディアの空間的解像度と遅延時間を指定すればよいことを説明する。

1周期分のデータの送信処理時間の最大値は、アプリケーションから要求される遅延時間から入力処理、符号化処理時間を引いた時間となる。なおOSが入力処理、符号化処理および送信処理を図3のように並列に処理することができれば、アプリケーションは遅延時間の最大値として周期時間の倍の時間を指定することが可能である。次に、入力処理、符号化処理時間は連続メディアの空間的解像度である1周期で処理されるデータ量から一意に定まる。以上のことから、連

続メディア処理において、空間的解像度と遅延時間が QoS として指定されれば、この連続メディアデータの送信処理の周期時間および 1 周期時間で処理されるデータ量が一意に定まるため、本論文で提案している送信スケジューリングを送信処理に利用することができる。

4.4 帯域が動的に変化する連続メディアの送信

動画のような連続メディアを符号化するときは圧縮が行われる。このとき、圧縮率は周期ごとのデータに依存するために、各周期で送信するデータ量は異なっている。このように、送信レートが動的に変化する場合、ATM ではそのピークレートを指定してネットワークの帯域を予約するサービスである VBR (Variable Bit Rate)¹³⁾ を利用することができる¹⁴⁾。VBR サービスでは、送信者がピークレート以下のレートで送信している場合は、その利用されていない帯域を他の送信者が利用することができるため、ネットワークの帯域を有効に利用することができる。

VBR サービスを利用している場合でも、それぞれの周期で送出レートを最小にする必要がある。送信レートが動的に変化する場合、周期ごとに処理するデータ量は変化するけれども、周期時間は一定である。そのため、ピークレートを発生させるデータ量の送出レートを最小にするスケジューリングは、すべての周期の送出レートを最小にすることができる。そこで、送信レートが動的に変化する場合は、ピークレート時のデータの送出を最小にする送出スケジューリングを行えばよい。

5. 計測値によるスケジューリング方法の決定例

本章では、1 回のシステムコールで送出するデータ量とシステムコール間隔を変化させたときに、それぞれ送出に必要な帯域を測定によって求め、その測定値を用いて、与えられた品質の連続メディアの送出レートの最小値と、そのときの送出スケジューリングを求める例を示す。

5.1 測定環境

図 4 に測定環境を示す。以下にその物理的なハードウェア環境と測定に用いたソフトウェアを説明する。

ハードウェア環境

測定は、2 台の性能の同じワークステーション (Sun Microsystems 製 SPARC station 20, OS は Solaris 2.4) を ATM スイッチ (NEC 製 ATOMIS5) で接続して行った。それぞれのワークステーションには ATM ネットワークインタ

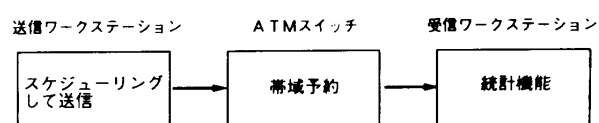


図 4 測定環境

Fig. 4 Measurement environment.

フェースカード (Efficient Networks 社製) を挿入し、IP アドレスと ATM の VC (Virtual Circuit) の対応づけには RFC1577⁷⁾ を用い、またパケットのカプセル化方式は LLC/SNAP²⁰⁾ を用いた。ATM スイッチでは送信側のワークステーションと受信側のワークステーションを CBR¹³⁾ (Constant Bit Rate) で接続し、帯域は送信側ワークステーションのデータがスイッチに流れ込む部分で、ATM スイッチの UPC 機能で制御した。なお、本論文で測定に使用した OS には CPU を予約する機能はなかったため、CPU を予約することと同じ状況を作り出すために、関係のない処理の負荷やネットワークのトラフィックを遮断した。また 4 章で述べた適切なゆらぎ時間を設定することによって負荷の変動による影響を受けにくくした。

測定用ソフトウェア

測定のために、送信側で 1 回のシステムコールで送信するデータ量とシステムコールの時間間隔が自由に設定できるソフトウェアおよび、受信側で受信したデータの統計をとる機能を持ったソフトウェアを作成した。トランスポートプロトコルとしては UDP を用いた。なお、現在ではまだ OS に送出スケジューリングの機能を組み込んでいないため、この計測用の送信ソフトウェアが送出スケジューリングも行っている。

5.2 送信スケジューリングとそのときの帯域

まず、1 回のシステムコールで送信するデータ量とシステムコールの間隔を変化させた場合、その送出に必要な帯域を測定した。送出に必要な帯域は測定用のソフトウェアが IP レベルで、100% 送信できる最小の帯域を求めた。この最小の帯域は、ATM スイッチの UPC を最初は十分な帯域に設定しておき、受信側のワークステーションでの IP パケットの破棄が始まるまで、徐々に UPC の設定帯域を落としていくことによって決定した。

本測定に用いた OS が 1 回のシステムコールで送出できる UDP のパケットサイズが 64 KByte であったため、測定値として、1 回のシステムコールで送出するデータの量は 64 Kbyte, 32 Kbyte, 16 Kbyte, 8 Kbyte,

4 Kbyte と変化させ、それぞれ、システムコールの間隔を 10 msec, 25 msec, 50 msec, 100 msec と変化させたときの送出に必要な最小の帯域を求めた。この時測定結果を図 5 に示す。図 5 は、横軸がシステムコールの間隔を、縦軸が送出の最小帯域を示す。

次に、測定に用いた OS のデータ量と、そのデータをシステムコールが送信するのに要する応答時間を図 6 に示す。図 6 は、横軸が 1 回のシステムコール

で送信したデータの量を表し、縦軸がそのときに要した応答時間を表している。

5.3 送出スケジューリングの決定例

品質が指定された連続メディアの送出スケジューリングおよび、そのときの送出レートを、5.2 節で求めた図 5 および、図 6 のデータに基づいて、本論文で提案している手法を使って求める例を示す。

●周期時間が 50 msec, 1 周期時間内で処理されるデータ量が 64 Kbyte である品質の連続メディア送出に必要な帯域を求める。

1 回のシステムコールで送信されるデータ量が d のとき、システムコール間隔 t は、式 (4) のようになる。

$$t = 50 \times \frac{d}{64} \quad (4)$$

このとき、送信するデータ量 d と時間間隔 t の送出に必要な帯域 $B(d, t) = B(d, 50 \times \frac{d}{64})$ の値を、図 7 中の太線で表す。すると図 7 から、その最小値は、 $d = 32$ Kbyte, $t = 25$ msec のときで、その値は 10.3 Mbps であることが分かる。また、32 Kbyte のデータを送信するシステムコールにかかる時間は図 6 から 3 msec であるので、25 msec での時間間隔でのスケジューリングは可能である。

結局、指定された連続メディアは、1 回のシステムコールで送信するデータ量を 32 Kbyte、システムコール間隔を 25 msec で送出するスケジューリングを行えば、最小送出レート 10.3 Mbps で送信できることが分かった。

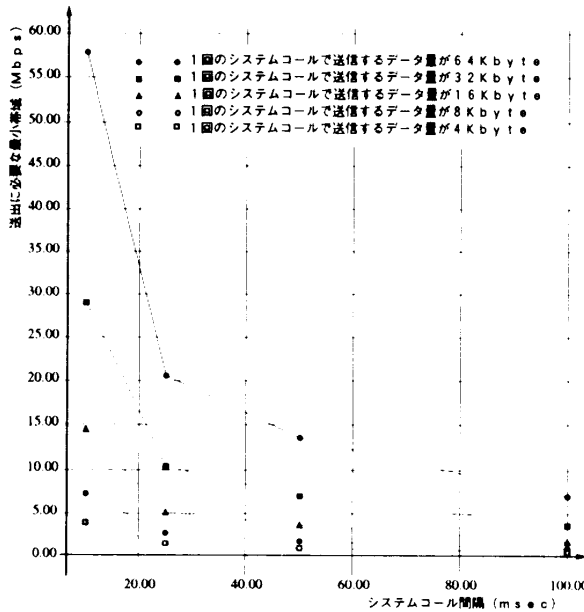


図 5 送出データ量と送出時間間隔で決まる送出帯域
Fig. 5 Bandwidth for transmission data and interval time.

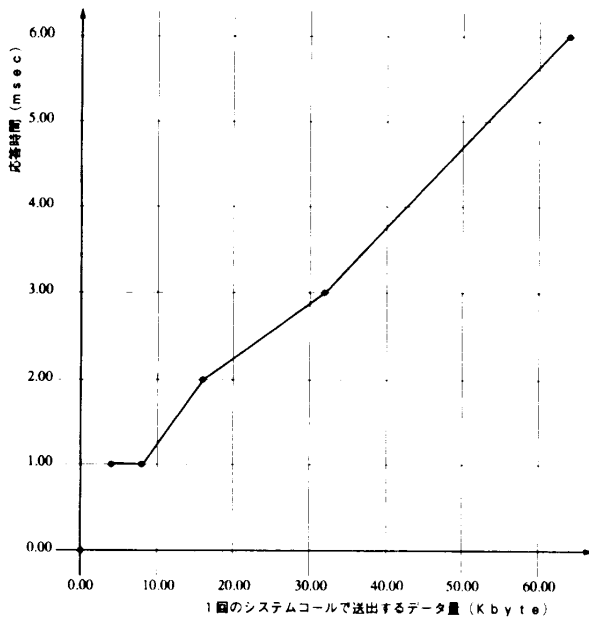


図 6 システムコールの応答時間
Fig. 6 Response time of systemcalls.

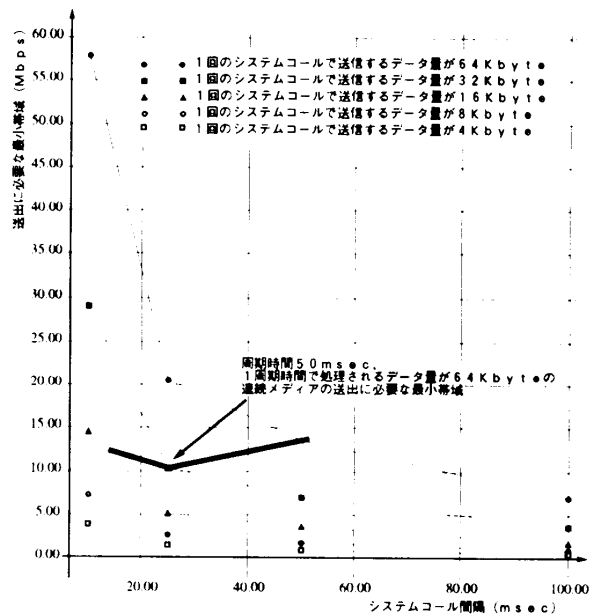


図 7 連続メディアの送出に必要な帯域
Fig. 7 Bandwidth for continuous media transmission.

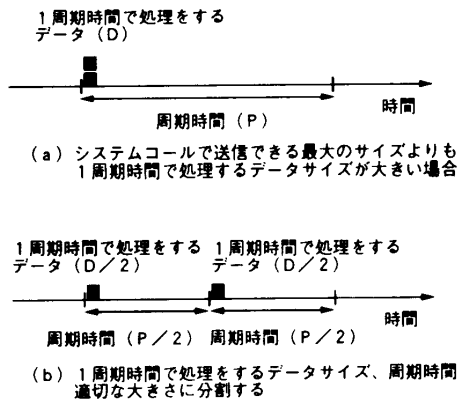


図 8 複数のシステムコールを必要とする連続メディア送信
 Fig. 8 Continuous media transmission by multiple systemcalls.

● 周期時間が 100 msec、1 周期時間内で処理されるデータ量が 128 Kbyte である品質の連続メディア送出に必要な帯域を求める。

1 周期あたりのデータ量が、本測定で用いた OS の 1 度のシステムコールで送信できるデータの最大値である 64 Kbyte 以上の品質の連続メディアが与えられた場合、その 1 周期あたりのデータを分割する必要がある。本論文では最小の帯域を求めることに着目しているため、図 8 で示されるように元の連続メディア (a) を、周期時間、データ量をそれぞれ半分にした 2 つの連続メディア (b) に分割することが最も送出レートの低くなる分割となる。

この連続メディア送信は周期時間が 50 msec、1 周期時間内で処理されるデータ量が 64 Kbyte の品質の連続メディア送信と同じであると考えことができ、先の結果から、1 回のシステムコールで送信するデータ量が 32 Kbyte、システムコール間隔が 25 msec である送出スケジューリングと、そのときの最小送出レートとして 10.3 Mbps を求めることができる。

本論文で得られた結果を評価するために、指定した品質の連続メディアと同じデータを送信するソフトウェア、およびデータを受信するソフトウェアを試作して評価に用いた。送信側のソフトウェアは連続メディアの品質として周期時間と 1 周期時間内で送信するデータ量の指定が可能で、送出スケジューリングとして、1 回のシステムコールで送信するデータ量およびシステムコール間隔の指定が可能である。また、受信側のソフトウェアでは、受信したデータの統計情報をとることができる。評価用のソフトウェアを用いて、図 4 のハードウェア環境で評価を行った。評価では、説明に用いた 2 種類の連続メディアの送信を、本論文で求めた帯域で予約したネットワーク上で、本論文で求めた

送信スケジューリングに基づいて送信を行った。受信側の統計情報では送信したすべての連続メディアデータが破棄されずに受信できたことを確認した。

6. おわりに

本論文では、従来の共有型ネットワークでは問題にならなかった連続メディアの送出レートが平均レートよりも高いことが、帯域予約されたネットワークでは、帯域の利用の面で問題となることに着目した。その問題を解決するために、連続メディア送信をデータ送出スケジューリングとあわせてモデル化し、そのモデルに基づいて、送出レートを最小にする送出スケジューリングを求める方法を提案した。また、その送出スケジューリングを利用するために OS に要求する QoS について述べた。最後に、帯域予約のできるネットワーク上で測定を行い、与えられた品質の連続メディアを、最小の送出レートで送信する送出スケジューリングを求める例を示し、評価によってその正当性を示した。現在、我々は、ATM によって帯域予約されたネットワークで、実時間的に入力した動画と音声の送信を、本論文で提案した送出スケジューリングを利用して送信するアプリケーションを開発中であり、いくつかの実用的な実験も行っている²¹⁾。また、送出スケジューリングは現在アプリケーションで行っているが、将来的には OS に組み込む予定である。

現在、マルチメディア通信のためのネットワークの基盤は実用段階に近づいている。このような次世代型のネットワークでは、資源予約技術によって、アプリケーションに対してサービスの品質を保証することができる点は大いに期待できる。しかしながらその一方で、従来のネットワークと比べて資源に対する利用上の制限が発生するため、今までとはまったく異なる新たな問題が発生することが予想される。本論文では帯域予約に着目したが、今後は他の資源が予約されることによって発生する問題に取り組んでいく予定である。

我々は、今後も、実際に帯域や遅延時間等の資源予約のできるネットワーク上で、新たに発生する問題を解決しながら、次世代型のネットワーク上で、音声、画像通信や、それらのマルチメディア通信を用いた遠隔ヒューマンコミュニケーション応用といった様々な実験を行い、本格的なマルチメディア通信のために必要な新しい技術を開発していく予定である。

謝辞 日頃から熱心に議論していただく On-Line University Project のメンバおよび、奈良先端科学技術大学院大学平原正樹助教授、荒木研究室、福田研究室の諸先生方および学生諸君に感謝する。

参 考 文 献

- 1) Topocia, C. (Ed.): Experimental Internet Stream Protocol, Version 2 (ST-II), RFC1190 (Oct. 1990).
- 2) Partridge, C. and Pink, S.: An Implementation of the Revised Internet Stream Protocol (ST-2), *Journal of Internetworking: Research and Experience* (Mar. 1992).
- 3) Delgrossi, L. and Berger, L.: Internet Stream Protocol, Version 2 (ST2), Protocol Specification - Version ST2+, RFC1819 (Aug. 1995).
- 4) Zhang, L., Deering, S., Estrin, D., Shenker, S. and Zappala, D.: RSVP: A New Resource ReSerVation Protocol, *IEEE Network*, pp.8-19 (Sep. 1993).
- 5) Zhang, L., Estrin, D., Herzog, S. and Jamin, S.: Resource ReSerVation Protocol (RSVP) - Version 1, Function Specification, Internet Draft (July 1995).
- 6) Schulzrinne, H. and Casner, S.: RTP: A Transport Protocol for Real-time Application, RFC1889 (Jan. 1996).
- 7) Laubach, M.: Classical IP and ARP over ATM, RFC1577 (Jan. 1994).
- 8) Cole, R.G., Shur, D.H. and Villamizar, C.: IP over ATM: A Framework Document, Internet Draft (Sep. 1995).
- 9) 村岡: On-Line University プロジェクトについて, 第2回 JAIN Consortium Symposium 論文集, pp.1-10 (Dec. 1994).
- 10) Eriksson, H.: MBONE, The Multicast Backbone, *CACM*, Vol.37, No.8, pp.54-60 (Aug. 1994).
- 11) 荒木, 岡村, 佐伯, 三浦, 落水, 篠田, 海谷: ネットワークを介した協調活動の実現にむけて, 情報処理学会サマワーショップ・イン・立山 (July 1995).
- 12) Campbell, A., Coulson, G. and Hutchison, D.: A Quality of Service Architecture, *Proc. ACM SIGCOMM '93*, Vol.23, No.4, pp.6-27, (Sep. 1993).
- 13) The ATM Forum: *ATM User-Network Interface Specification, Version 3.1*, Prentice Hall (1995).
- 14) Reibman, A.R. and Berger, A.: On VBR Video Teleconferencing over ATM Networks, *Proc. GLOBECOM*, pp.314-319 (Dec. 1992).
- 15) Tokuda, H., Tobe, Y., Chou, S.T.-C. and Noura, J.M.F.: Continuous Media Communication with Dynamic QOS Control Using ARTS with and FDDI Network, *Proc. ACM SIGCOMM '92*, Vol.22, No.4, 88-98 (Oct. 1992).
- 16) Mercer, C.W., et al.: Processor Capacity Reserves: Operating System Support for Multimedia Application, *Proc. Intl. Conf. on Multimedia Computing and Systems*, pp.90-99 (1994).
- 17) 市野: オーディオ機器, コロナ社, (1991).
- 18) 安田: マルチメディア符号化の国際標準, 丸善 (1993).
- 19) Schulzrinne, H.: RTP Profile for Audio and Video Conference with Minimal Control, RFC1890 (Jan. 1996).
- 20) Heinanen, J.: Multiprotocol Encapsulation over ATM Adaptation Layer 5, RFC1483 (July 1993).
- 21) 中村, 池田: On-Line University Project 高度知識共有 (教育) 研究部会 (2), *bit*, Vol.28, No.3, pp.62-63 (1996).

(平成7年10月2日受付)

(平成8年3月12日採録)

岡村 耕二 (正会員)



1965年生。1988年3月九州大学工学部情報工学科卒業。1990年3月同大学院修士課程修了。同年4月三菱電機(株)入社。1993年4月奈良先端科学技術大学院大学情報科学

研究科助手。1996年4月神戸大学総合情報処理センター助手。工学修士。分散並列処理, インターネット上でのマルチメディア処理, マルチメディアを利用したヒューマンコミュニケーションなどの研究に従事。ソフトウェア科学会会員, Jain Consortium 幹事。

荒木啓二郎 (正会員)



1954年生。1976年3月九州大学工学部情報工学科卒業。1978年3月同大学院修士課程修了。同年4月九州大学工学部助手, 1984年8月同助教授。1993年4月奈良先端科学

技術大学院大学情報科学研究科教授。1996年4月九州大学大学院システム情報科学研究科教授。工学博士。形式的仕様記述, プログラミング言語, 並列/分散処理などの研究に従事。IEEE Computer Society Certificate of Appreciation Award など受賞。情報処理学会調査研究運営委員会委員, ソフトウェア技術者協会幹事, 博多祇園山笠西流赤手拭など。