

シナリオに基づいたマルチメディア転送プロトコル

清水省悟 柴田義孝

東洋大学 工学部 情報工学科

{simi, shibata}@yosemite.sb.cs.toyo.ac.jp

マルチメディア情報をネットワークを越えて提供するには、口の動きに声を合わせるといったビデオとオーディオ間のLip同期や、複数のメディアを同時に提供するというメディア間のシーン同期制御が必要になる。また、ユーザ端末に合わせたメディアのフォーマット変換や、メディアソースがJPEG, MPEGなどにより圧縮されている場合はデータの伸張が必要になる。そこで本稿では、シナリオに基づいた動的メディア及び静的メディアの転送プロトコルの為のメディア変換時間、及びネットワーク転送時間を考慮したデータ転送方法、さらにペトリネットを使用したメディア同期方法の設計、実装について述べる。

Multimedia Transmission Protocol based on Scenario

Shogo SHIMIZU, Yoshitaka SHIBATA

Department of Information and Computer Sciences
Toyo University

{simi, shibata}@yosemite.sb.cs.toyo.ac.jp

In order to provide multimedia information over a network, lip synchronization and scene synchronization are required to synchronize between audio and video more precisely as well as between continuous media and static media, such as image and text data. Media format conversion of media data, such as JPEG and MPEG compression/decompression and color depth or audio sampling frequency are also required. In this research, we proposed a media transmission protocol which considers lip and scene synchronization functions and media format conversion, and a scheduled data transmission functions based on the scenario in application services.

1 はじめに

情報システムの発達により、画像や文字、音声といった複数の異なるメディアを同時に表示可能なコンピュータや、百数十メガビット毎秒を越える転送速度を持つ高速ネットワークが実用化されている。これにより、分散環境でマルチメディア情報をユーザに提供するサービスの提供が可能になった。この結果、マルチメディア オンデマンド サービスを利用した“電子新聞ネットワーク サービス”も実現可能になってきた。この電子新聞ネットワーク サービスでは、ネットワーク上に各新聞社が分散しており、マルチメディア端末を使用してユーザはそのネット

ワークに接続しさえすれば、何時でも何処からでも新聞社が提供している新聞を読むことが可能となる。その新聞には写真や文字だけでなく、新聞の中に映像や音が流れ、インタラクティブな操作によってユーザの欲する情報を提供可能とする。このようなサービスでは、複数のメディアが、そそれを提供するための時空間情報であるシナリオに基づいてユーザに提供される。しかし、ネットワークを越えてそれらのメディアを意味的に統合された一つの情報として提供するためには、シナリオに従うメディアデータの転送や、その提供の際に必要なメディアフォーマット変換などを行わなければならない。

そこで本研究では、アプリケーションがシナリオに基づいてメディアを提供するために、提供時刻までにはメディアデータの転送が完了するような各メディアデータの送信時間を予め計算してスケジューリングを行って、それに基づいて転送を行う。また、受信側においては、受信されたメディアデータをシナリオ通りに提供するために、ペトリネット表現されたメディア同期処理をしている。そして、動画や音声といった連続メディアのみならず、画像や文字といったバーストメディアより構成されるマルチメディア情報を、シナリオに基づき時間的に統合して提供するための同期転送プロトコルの設計、実装及び性能評価を行なった。

2 プロトコルアーキテクチャ

2.1 システムアーキテクチャ

マルチメディア情報にはそれを構成する複数のメディア間の時間関係及び空間関係を示す情報が存在し、これをシナリオと定義する。アプリケーションは、そのシナリオを管理し、シナリオ通りに複数のメディアを提供する。

そして、このシナリオに基づいた転送を行うためのプロトコルとして、図1に示すような階層構造をするクライアント・サーバ方式により構成した。

Synchronization Layer(以下、Sync 層)では、動画フレームと対応する音響セグメント間の同期処理及び、任意時刻に表示される静止画やテキストの出力時間同期などを行なう。Transform Layer(以下、Trans 層)では圧縮・伸張、端末属性に合わせたメディアのフォーマット変換を行う。Media Flow Control Layer(以下、MFC 層)ではパケットロスを検討したネットワーク転送を行う。Control and QoS Management Layerでは、インタラクティブな操作に対応するための制御や、ユーザの要求に応じた QoS(Quality of Service) の管理を行う [4]。

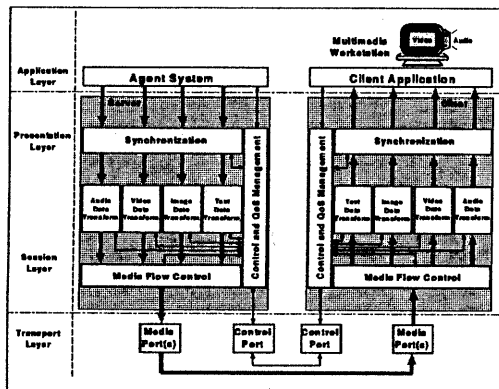


図1: システムの階層構造

2.2 メディア属性

Sync 層での同期処理や、Trans 層でのフォーマット変換などを行うために、アプリケーションは転送プロトコル内の各層に対して、転送するメディアデータのフォーマットやシナリオ情報を予め通知しなければならない。この情報をメディア属性と定義し、アプリケーションではメディアデータの転送前に、アプリケーションが管理しているシナリオをメディア属性として、コントローラに通知することにより各層の処理が行われる。メディア属性の内容を以下に示す。

- **メディア型** : Audio, Video, Image, Text といったメディアの種類を示す。
- **時間属性** : メディア開始時刻、継続時間、同期クラス、出力重視度
- **ソース属性** : サーバ側のアプリケーションが同期層へ渡す各メディアデータのフォーマット。フォーマットとは、例えばオーディオであれば変調方式、サンプリング周波数などのメディア固有のフォーマットを表す。
- **出力属性** : クライアント側の同期層からアプリケーションに渡すメディアフォーマット

メディア型で指定されたメディアデータがソース属性で指定されたフォーマットで、サーバアプリケーションから Sync 層へ送信される。プロトコル内部では、そのメディアデータを出力属性に適合させたフォーマット変換を行った後、クライアントアプリケーションに渡す。

2.3 サービスフロー

本プロトコルが提供するサービスは、図2で示すように4つの Phase に処理を分離される。

1. **Open Phase** : 情報サービスの開始として、ユーザの認証とクライアントでの出力フォーマット属性の通知を行い、クライアントとサーバの接続を行う。
2. **Request Phase** : マルチメディア情報の要求を行う。クライアントアプリケーションの要求にはマルチメディア情報のシナリオのタイトルをシステムに通知し、サーバのアプリケーションによってシナリオの検索が行われる。このシナリオをメディア属性としてシステムに通知する。

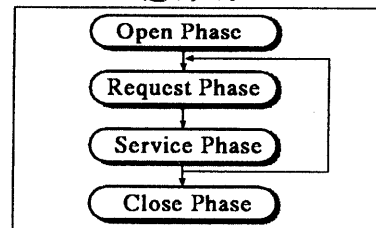


図2: サービスフロー

3. **Service Phase** : マルチメディア情報の転送を行う。システム内部では Request Phase で得られたメディア属性に従いスケジューリングが行われ、フォーマット変換を行いながら、スケジューリングに基づいたメディア転送が行う。
4. **Close Phase** : サービスの終了を行う。

3 Synchronization Data Unit

Sync 層では複数のメディアを統一した形でメディア同期処理を行うために、無圧縮ビデオでは 1 フレーム、MPEG データは 1GoP(Group of Picture)[5]、オーディオは 1 セグメント、静止画は 1 イメージを一つの単位として考え、それを Synchronization Data Unit(SDU) と定義する。以下に SDU の構造を示し、始めの 4 項目がヘッダ情報となる。

- 全長: バイトを単位としたデータの全長を示す。
- メディア型: オーディオ、ビデオ、イメージ、テキストといったメディアの種類を示す。
- メディア ID: 同じメディア型の中で複数存在するメディアデータから、ユニークに決定するための番号。
- シーケンス番号: 連続メディアのみ有効のフィールドであり、メディア ID で指定されたメディアデータ中の何番目のデータであるかをフレーム、GoP、セグメント単位で表す。
- メディアデータ: 上位プロトコルから渡されるメディアデータ。

図 3 はビデオフレームを 1SDU とした例である。全長は、フレームサイズ (160(width) × 120(height) × 1(depth)) に、ヘッダ長 (16) を加えた 19216[bytes] となる。メディア型は VIDEO、識別のために使用されるメディア ID として、ここでは 123、シーケンス番号はフレーム番号が代入される。

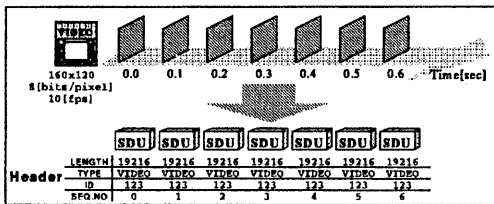


図 3: Synchronization Data Unit

4 スケジューリング

アプリケーションがメディアをシナリオに示されている時間通りに提供するには、プロトコル内部でメディア属性に記されているメディア出力時間を、実時間と照らし合わせながら転送する方式が必要となる。しかし、メ

ディアデータはネットワークを越えて転送され、プロトコル内部でフォーマット変換を行っているために、メディア出力時間だけを考慮するだけでは、シナリオに基づいた転送は不可能である。

そこで本システムでは、メディアデータの End-to-End 遅延をネットワーク転送時間だけでなく、圧縮伸長、端末属性に合わせたフォーマット変換などに要する時間も考慮し、メディア提供時刻よりもその遅延時間分先送りするといった転送方式を行なっている。この送信開始時刻を確定する計算をスケジューリング (Scheduling) と定義し、サーバ側の Sync 層ではこの開始時刻に基づいてメディアデータの転送を行っている。

$$\begin{aligned}
 &t_{prev} = \infty \\
 &\text{for } n = \text{the number of SDU to } 1 \\
 &\quad t_{delay_n} = S_n / C + T_d + T_n \\
 &\quad t_{send_n} = \min[t_{prev}, t_{output_n}] - t_{delay_n} \\
 &\quad t_{prev} = t_{send_n} \\
 &\text{end}
 \end{aligned}$$

スケジューリングは、上式のアルゴリズムにより計算式される。ある SDU_n のメディア変換に必要な時間を T_n 、転送データサイズを S_n 、確保されたネットワークの転送帯域を C 、その時のディレイを T_d 、送信開始時刻を t_{send_n} とすると、End-to-End 間の総合遅延 (t_{delay_n}) は $S_n / C + T_d + T_n$ となる。その SDU が時刻 t_{output_n} で出力する場合、時刻 $t_{output_n} - t_{delay_n}$ に送信を開始すれば出力時刻に間に合うが、全てのメディアで一つのストリームを使用する場合には、図 4 のように、前後の SDU の送信時間が重ならないように SDU の送信時刻をシフトする必要がある。このようにシナリオに基づいたメディア統合転送は、サーバ側の Sync 層が送信開始時刻 t_{send_n} に従った SDU の送信により行われる。

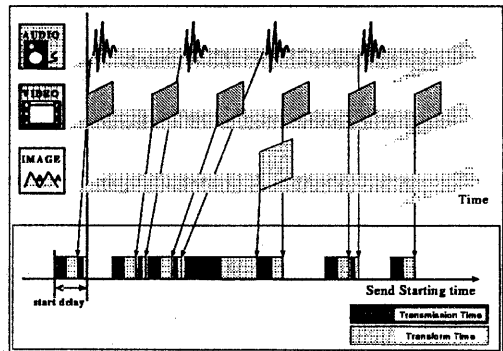


図 4: スケジューリングのアルゴリズム

5 メディア同期モデル

本プロトコルでは、サーバ側のスケジューリングによってメディアデータの転送が行われるために、シナリオに基づくメディア間の相対関係及び時間的關係が考慮されている。しかし、メディアデータがネットワークを越え、かつシステムの負荷影響を受ける事によって生じるメディアの時間的誤差を考慮した場合、そのメディアデータを受信したクライアント側で再同期を行う必要がある。そこで、ネットワークを越えてメディアを提供する際に必要とされるメディア同期方式を挙げ、ベトリネットを使用した Sync 層への導入について説明する。

5.1 メディア内同期

ビデオのような連続メディアを提供する際には、ユーザに対して一定のフレームレートで再生しなければならない。かつ、そのビデオの開始はシナリオに従って提供しなければならない。これらの問題に対応するために、以下のような同期処理が必要となる。

- レート一定制御 (Rate Regulation Control) : この制御はビデオ、オーディオなどの連続メディアの再生の際に、それらが録画や録音されたレートを基本として、QoS がシステム及びネットワークの負荷によって設定されたフレームレートを守る制御である。例えば、毎秒 10 フレームに設定された動画データの場合、0.1 秒間隔でフレームを提供する制御である。
- 時間的構成 (Temporal Composition) : 時間的構成とは、メディアを表示開始時刻に合わせて提供する事を意味し、メディアの提供開始時に必要とされる。

5.2 メディア間同期

マルチメディア情報ネットワークのサービスでは、複数のメディアがネットワークを介して提供されるために、メディア間の時間的關係が損なわれる可能性がある。この問題に対処するために、以下のような同期処理が必要となる。

- リップ同期 (Lip Synchronization) : リップ同期とは、例えば人物の口の動きに声を合わせるような、時間的に密接な時間關係があるメディア間で行われる連続的な同期である。この同期方式に関して筆者らは今まで、Strict Synchronization、Relaxed Synchronization、Silence Synchronization の 3 種類の同期方式の設計、実装を行ってきた [2, 3]。
- シーン同期 (Scene Synchronization) : シーン同期とは、映画の字幕と音声のように、メディアの開始と提供中の連続メディアの時間的關係を損なわないようにするための同期である。

5.3 ベトリネットによるメディア同期

クライアントの Sync 層では、先に述べた同期処理を必要とされるが、メディア属性にはリップ同期を行う全てのフレームとセグメントの記述や、シーン同期を行うイメージとオーディオの關係などといったメディア間の時間關係に関する情報は記されていない。そこでシナリオの中でメディア同期処理を明確にし、実装への展開を容易にするためベトリネットを使用した。

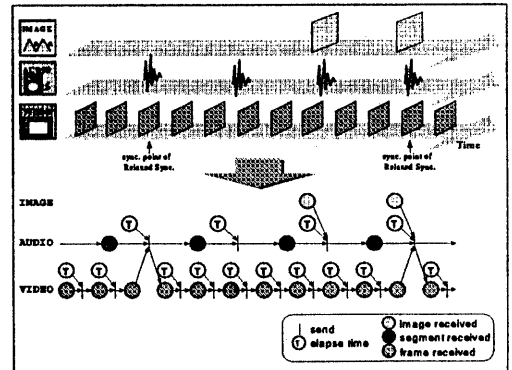


図 5: シナリオのベトリネット表現

図 5 は、Relaxed Sync. を行ないながら再生されているオーディオ・ビデオの途中に、2 枚の静止画が提供されるといったサンプルのシナリオを、ベトリネットで表現し同期箇所を明確にした図である。ビデオで行われるフレームレート一定制御を、図より“データの受信”と“レート時間の経過”の 2 つの状態 (プレイス) が完了すれば、データの送信への遷移 (トランジション) で表現する。また、リップやシーン同期に関しては、同期を行なうべきメディア全ての受信の完了後、次の状態への遷移で表現が可能となる。

図 6 は、このベトリネットで表現された各プレイスの状態及び遷移行程を詳細にしており、クライアント側の Sync 層では、この図に示される同期処理を行っている。メディア同期モデルと対応させると、メディア間同期はプレイス: p_3 と p_4 、メディア内同期はプレイス: p_5 と p_6 に対応する。まず初めは、Sync 層で SDU の受信状態となっている (p_1)。Trans 層から SDU を受信すると、 p_2 では次のような同期方式の選択を行う。受信された SDU のシーケンス番号を参照し、もしゼロであったならばシーン同期として t_2 を発火させ、 p_3 では同期をすべきメディア数 n 個のトークンを受けるまで待機することにより、シーン同期が可能となる。シーケンス番号がゼロでなく、そのシーケンス番号がリップ同期ポイントであれば t_3 を発火させ、 p_4 では p_3 と同様に、 n 個のトークンを受けるまで

待機することにより、リップ同期が可能となる。それ以外であった場合は、メディア間同期処理は行われず p_6 へ遷移する。 p_6 ではメディア内同期の時間的構成が行われ、 p_6 ではレート一定制御が行われる。このようにメディア同期モデルにペトリネットを用いることにより実装が簡単に行える。

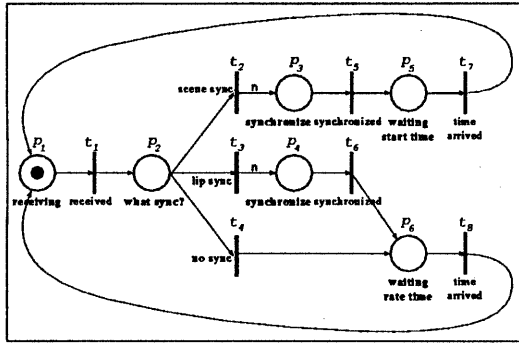


図 6: メディア同期モデルのペトリネット表現

6 転送方式

MFC層では、転送方式として以下の二種類の方式を考慮している(図7)。

6.1 マルチストリーム方式

シナリオには、特性転送の異なる複数のメディアが含まれている。マルチストリームは、これらの特性を考慮した転送を行う事を目的とし、メディア毎にストリームを割り当てる方式である。この方式ではメディア毎にストリームが異なるためにメディアの特性転送を考慮した転送が行えるが、ネットワーク転送中にメディア間の相対的關係が損なわれてしまう可能性があり、受信側でメディアの再同期が必要とされる。

6.2 シングルストリーム方式

この方式は、全てのメディアで一つのストリームを使用して転送を行う方式である。この場合、マルチストリームのようにメディア毎に特性を考慮した転送は行えないが、ネットワーク転送中でもメディア間の相対的關係が保持される。

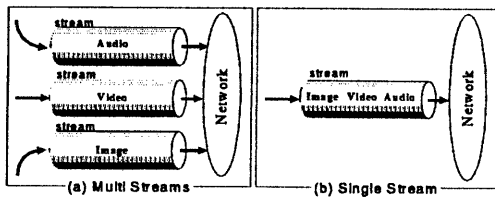


図 7: 転送方式

7 プロトタイプ及び性能評価

本研究で提案したシナリオに基づいたメディア転送方式の性能評価を行うために、オーディオ、ビデオ、イメージ、テキストといった4つのメディアを統合して提供するシステムのプロトタイプを図8の環境で構築した。ネットワークは転送速度 100[Mbps] の FDDI、ネットワークの通信プロトコルとして UDP を用いた。

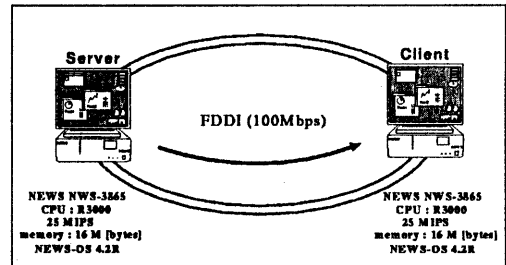


図 8: プロトタイプ

図9のような電子新聞を想定したスポーツ記事のシナリオを使用して、このプロトタイプの性能評価を次の3種類の転送方式で行った。

1. マルチストリームを使用して、スケジューリングを行わなかった場合
2. マルチストリームを使用して、スケジューリングを行った場合
3. シングルストリームを使用して、スケジューリングを行った場合

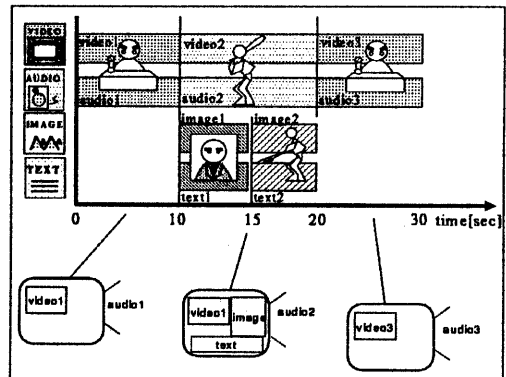


図 9: 評価に使用したシナリオ

7.1 提供時刻

図10に、スポーツ記事のシナリオをスケジューリングによって計算された送信時間、シナリオに示されている提供時間、プロトタイプで転送を行った場合の計測時間を、3本のバーで示した。バーの左側には実際の時間を記した。Service Phaseの開始となる start 要求をクラ

クライアントが発行した時刻を実時間の開始としているため、開始メッセージがサーバに伝わり、メディアデータが転送され、データが到着するまでの時間がシナリオ開始時の遅れとなっている。よってシナリオの開始時間が、スケジューリングによって予め計算された start delay よりも遅れてしまっている。しかし、シナリオ全体の転送時間から見ても、その開始時間だけが遅れているだけで、シナリオ通りにメディアデータが転送された事が確認できる。

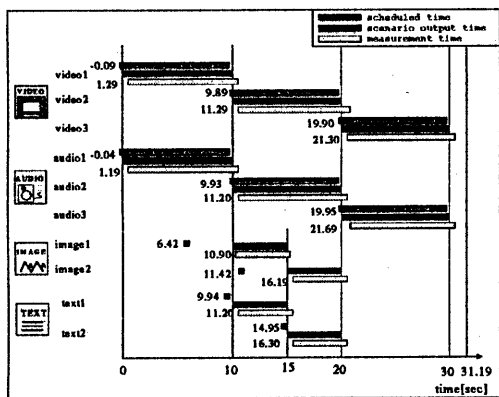


図 10: シナリオ提供時間 ((3) の場合)

7.2 提供時刻の遅れ

図 11 は、シナリオに記載されている論理時間と、実際に提供した際の実時間との誤差をビデオに着目した図である。スケジューリングを行わなかった場合の開始時刻の遅れが 3.8 秒となり、他の場合に比べ大きい。これは、Image1, Image2 をシナリオ開始時に送っているため、ビデオ転送に必要な End-to-End でのスルーポイントが確保できなかったためと考えられる。マルチストリーム方式を使用した場合でも約 3 秒位であった。しかしながら、シングルストリームの場合では、バーストデータによる転送の影響は殆ど見られなかった。

8 まとめ及び今後の課題

本稿ではまずシナリオに基づく転送方式を行うためのプロトコルストラクチャと、プロトコルで利用する SDU、並びにシナリオ情報としてメディア属性を定義した。またメディアの出力時間から転送開始時刻を計算するスケジューリングの方法、クライアントの Sync 層での同期処理としてメディア間としてはシーン及びリップ同期、メディア内としてはレート一定制御及び時間的構成を定義した。その同期モデルをベトリネットにより表現し、クライアントの Sync 層に導入した。FDDI ネットワーク上にプロトタイプを構築中し、スケジューリングを行った場合と行わなかった場合の比較を行った。連続メディアの転

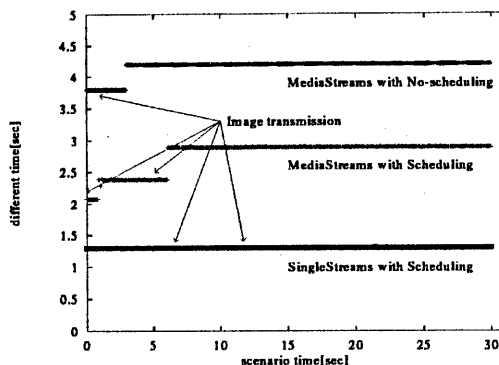


図 11: ビデオに着目したシナリオ時間と実時間との誤差

送中にバーストデータの転送を行った場合、スケジューリングによって、その連続メディアの転送への影響を押さえることが確認できた。

今後の課題としては、アプリケーションにおけるシナリオの記述方法、QoS 保証に対応したスケジューリング方法及び QoS 更新時における再スケジューリングの方法、クライアント-エージェント-データベースモデルへの対応などが挙げられる。

参考文献

- [1] 鈴木 隆之, “マルチメディア情報ネットワークのための Multimedia Remote Procedure Call の研究” 東洋大学電気工学科博士前期課程 平成 5 年度修士論文
- [2] 神原久夫, 河野太基, 柴田義孝: “パケットビデオシステムのための同期メカニズム, 情報処理学会第 46 回全国大会, IK-05, 1993
- [3] 清水省悟, 瀬田直也, 神原久夫, 柴田義孝: “マルチメディア情報ネットワークのためのパケットビデオシステムの設計と性能評価”, 情報処理学会第 47 回全国大会, 4V-05, 1993
- [4] 橋本浩二, 渡辺光輝, 柴田義孝, “パケット オーディオ・ビデオ システムの QoS 保証及び交渉機構について”, マルチメディア通信と分散処理研究会, 65-12, 1994
- [5] 渡辺光輝, 赤間孝司, 柴田義孝, “圧縮を考慮したパケット オーディオ・ビデオ システムのレート制御”, マルチメディア通信と分散処理研究会, 66-23, 1994
- [6] Thomas D.C. Little and A.Ghafoor, “Multimedia Synchronization Protocols for Broadband Integrated Services”, IEEE J. Select. Areas Commun, vol.9, no.9, pp. 1368-1382, Dec. 1991