

端末間時刻同期を用いた動画同期視聴における操作性の向上

森安 辰[†] 高野 祐太郎[‡] 大島 浩太^{††} 寺田 松昭^{††}

東京農工大学 工学部[†] 東京農工大学大学院 工学府[‡] 東京農工大学大学院 共生科学技術研究院^{††}

1. はじめに

インターネットを用いた動画視聴の普及に伴い、動画の多人数同期視聴サービスが多数登場している。多人数同期視聴とは、動画配信サービスにおいて、複数の視聴者が 1 つの動画をシーンまで同期した状態で鑑賞することを意味している。視聴者は動画の視聴時刻を合わせて、同じ動画を鑑賞することになる。既存サービスがいくつか存在し、文字によるコミュニケーションを提供しているものもある[1][2]。

しかし、これら既存サービスには以下の 3 つの問題点がある。(1)適切な動画同期技術を用いなければ視聴シーンがずれる。特に、同期精度は伝送遅延の影響を受ける。(2)サーバ・回線などのサービス提供コストが高く、利用ユーザ数や動画品質が制限される。(3)多くのシステムがライブストリーミングを対象としており、オンデマンド型動画配信には適していない。

(1)に関して、動画同期技術の先行研究は多数存在するが、研究ごとに適用環境が異なる[3][4]。本稿では、オンデマンド型動画配信を対象に、端末間時刻同期方式を用いて、高遅延クライアント存在下での操作性を向上する動画同期視聴システムの提案と実装について述べる。

2. 提案システム概要

図 1 に提案システムの概要を示す。動画同期視聴を行うユーザは、まず同期視聴グループを作成する。同期視聴グループを作成したユーザ端末がホスト端末となり、他のユーザの端末はクライアント端末としてホスト端末に接続する。

ユーザが動画ファイルを選択し再生すると、自動的に同期視聴グループ内の他ユーザ端末も同じ動画を再生する。再生中は、システムが動画同期制御を行い、グループ内の全ての端末が同じ動画の同じシーンを再生している状態になる。

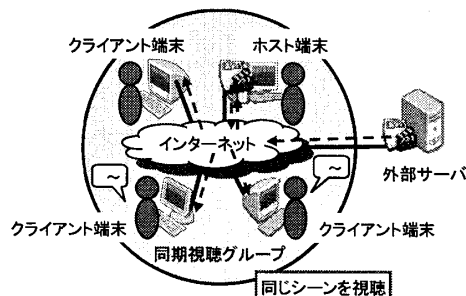


図 1 システムイメージ

ユーザは動画を鑑賞しながら、テキストチャットや音声通話でコミュニケーションを行う。

視聴する動画は、任意のユーザが所有している動画ファイルを参加者間でやり取りするか、外部サーバから取得する。

3. 課題

本システムの実現には次の課題がある。

(1) 動画同期制御

動画同期視聴を実現するには、全ての端末が同一時刻に同じシーンを再生する必要がある。これには適切な動画同期制御が必要となる。

(2) サービス提供コストの低減

動画同期制御と動画ファイルの転送を外部サーバで行うと、そのコストが利用ユーザ数や動画の品質に制限を与える要因となる。そのため、サービス提供コストを低減する必要がある。

4. 動画同期制御

本システムでは、同期した時刻を利用して動画同期制御を行う。また、動画の再生制御命令として再生、一時停止、シークに着目した。

4.1 端末間時刻同期

ホスト端末の時刻を基準としてクライアント端末の時刻同期を行う。時刻同期には、端末間でそれぞれの内部時計の時刻差を計測する必要がある。計測には SNTP と同様の方法を用いる。

クライアント端末のリクエスト発行時刻 T_1 、ホスト端末のリクエスト受信時刻 T_2 、ホスト端末の応答送信時刻 T_3 、クライアント端末による応答受信時刻 T_4 とすると、ホスト-クライアン

A Group Synchronized VoD System with High Response
Shin Moriyasu[†], Yutaro Takano[‡], Kohta Ohshimada^{††},
Matsuaki Terada^{††}

[†]Faculty of Engineering, Tokyo University of Agriculture and Technology

[‡]Graduate School of Engineering, Tokyo University of Agriculture and Technology

^{††}Institute of Symbiotic Science and Technology, Tokyo University of Agriculture and Technology

ト間の時刻ずれ D は

$$D = ((T2 - T1) + (T3 - T4)) / 2$$

となる。ホスト端末の時刻 T_s を求めるには、クライアント端末のローカル時刻 T_c として

$$T_s = T_c + D$$

とする。この T_s の値を同期した時刻として定義する。動画同期制御では端末のローカル時刻ではなく、 T_s の値を利用する。

4.2 再生命令の同期

図 2 に再生命令の同期方式を示す。まず時刻 t_1 においてクライアント A が再生要求をホストに送信する。ホストは、命令実行時刻 t_2 を指定して、全てのクライアントに再生命令を送信する。再生命令を受信した端末は時刻 t_2 になると再生を開始する。

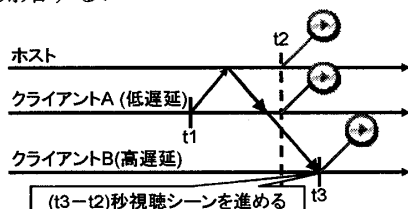


図 2 再生命令の同期方式

クライアント B のように、命令受信時刻 t_3 が t_2 より後の場合、視聴シーンずれが発生する。そこで、 (t_3-t_2) 秒視聴シーンを進めて視聴シーンずれを補正する。しかし、ずれの補正を行うと視聴シーンが不連続になり、ユーザの視聴品質が悪化する。

視聴シーンずれを回避するには、ホスト-クライアント間伝送遅延の最大値 D_{max} として

$$t_2 = t_1 + D_{max}$$

とすればよい。しかし、この方式では伝送遅延の大きいクライアントが存在する場合、 t_1 と t_2 の差が大きくなり全端末の操作性が低下する。

4.3 操作性向上のためのクライアント分類

前述の問題を解決するために、ホスト-クライアント間の伝送遅延値 100ms を基準として、低遅延クライアントと高遅延クライアントに分類する。基準値は、人間が違和感なく操作できる時間である[5]。

命令実行時刻 t_2 の設定を、低遅延クライアントの伝送遅延最大値 D_{max}' として

$$t_2 = t_1 + D_{max}'$$

とすることで、低遅延クライアントの操作性を向上する。高遅延クライアントでは視聴シーンずれが発生するので、ずれの補正を行う。

4.4 その他命令の同期

(1) 一時停止命令

一時停止命令は、ボタンを押した瞬間のシーンで停止させたいという要求がある。そこで、一時停止要求を視聴シーンと共に通知し、命令を受信した端末は即座に一時停止する。その後、通知された視聴シーンにシークすることで、視聴シーンずれを補正する。

(2) シーク命令

動画が再生状態の際にシークした場合、各端末の動画デコード処理時間の差により、視聴シーンずれが発生する。そこで、シーク命令を受信した端末は即座にシークし、一時停止する。その後、命令実行時刻 t_2 に再生を開始する。これにより、デコード時間の差を吸収する。

5. 実装

本稿で提案したシステムの構成を図 3 に示す。時刻同期や動画同期処理、動画ファイル転送などはユーザ端末間の通信で行う。これにより、サービス提供サーバにかかる負荷が低減される。

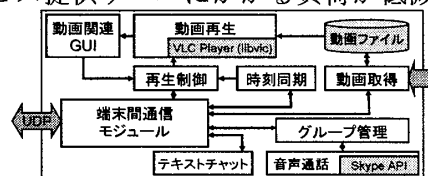


図 3 システム構成

6. おわりに

本稿では、オンデマンド型動画配信を対象に、端末間時刻同期方式を用いて、高遅延クライアント存在下での操作性を向上する動画同期視聴システムの提案と実装について述べた。

謝辞

本研究の一部は、共生情報工学推進経費の助成を受けている。

参考文献

- [1]Ustream: <http://www.ustream.tv/> (accessed 2010.1)
- [2]ニコニコ生放送: <http://live.nicovideo.jp/> (accessed 2010.1)
- [3]高野祐太郎, 田島孝治, 大島浩太, 寺田松昭: 投稿型動画視聴におけるユーザ間リアルタイムコミュニケーション支援システムの提案, インターネットと運用技術シンポジウム 2008 (2008).
- [4]Fernando Boronat, Jaime Lloret and Miguel Garcia: Multimedia group and inter-stream synchronization techniques: A comparative study, Information Systems, Vol.34, No.1, pp.108-131 (March 2009).
- [5]Jakob Nielsen: Usability Engineering, Morgan Kaufmann, San Francisco (1994).