

ストリーミングアプリケーションのトラフィック特性と 周期的切断が与える影響の解析

寺田雅徳^{†1,a)} 金子晋丈^{†2,b)} 寺岡文男^{†2,c)}

概要: 移動体が通信する場合、ハンドオーバーによって通信切断が発生する。列車通信システムではハンドオーバーが短い周期で定期的な発生し、動画や音声などのストリーム配信に大きな影響を与える。本稿ではYouTube, Skype, インターネットラジオという3つのストリーミングアプリケーションにおいて、ハンドオーバーが与える影響についての詳細な解析結果について示す。特に周期的切断が発生する環境で品質が低下した原因について、トラフィック特性から解析を行う。

キーワード: ストリーミングアプリケーション, 周期的通信切断, YouTube, Skype, インターネットラジオ

The Analysis of Traffic Characteristics of Streaming Applications and Impact of Periodic Communication Disruption

Abstract: When a node is communicating with another node while moving, communication disruptions occur due to handovers. In a mobile network on train, handovers occur in short period of time, resulting in bad effects on streaming applications. This paper investigates the behaviors of three kinds of streaming applications, i.e., YouTube, Internet Radio, and Skype when periodical handovers. We show effect of handover for the applications. Especially, we analyze the cause of media quality degradation from the point of view of traffic characteristics.

Keywords: Streaming applications, Media streaming, periodical disruption, YouTube, Skype, Internet radio

1. はじめに

当研究室では高速列車にブロードバンドインターネット環境を提供する通信システムについて研究を続けている。列車と地上とに赤外線通信装置 [1] を配置し、列車上の装置がハンドオーバーしながら地上の装置と通信することで、走行する列車に 1 Gbps 以上の帯域を列車内に提供する。当システムについて、2010 年 1 月～2 月に約 130 km/h で走行する列車を利用して性能を評価した。その結果、ネットワーク層のハンドオーバー所用時間は最良で 124 ミリ秒だった

たが、最悪では約 800 ミリ秒だった [2]。ハンドオーバーに際してはバースト的にパケットロスが生じ、通信品質に影響を与えると考えられる。Web 閲覧やメール送受信ではその影響は一時的であるが、動画や音声を扱うストリーミングアプリケーションは継続的に通信するため、動画や音声の品質低下や動作の停止など、大きな影響が生じることが予想される。

そこでストリーミングメディアアプリケーションについて、トラフィック特性と周期的ハンドオーバーの影響の解析を行っている。300 km/h で走行する新幹線に 300 m 間隔で地上に配置した赤外線通信装置を用いて 1Gbps のインターネット接続を提供することを想定し、3.6 秒ごとに 3.6 秒に 1 回の割合でハンドオーバーが発生する環境を構築して 2012 年 11 月に実験を行った。この実験では YouTube, インターネットラジオ, Skype という代表的な 3 つのストリーミングアプリケーションを解析対象とした。これらのアプリケーションは表 1 に示すように特性が異なる。擬似

^{†1} 現在、慶應義塾大学大学院理工学研究科
Presently with Graduate School of Science and Technology,
Keio University

^{†2} 現在、慶應義塾大学理工学部
Presently with Faculty of Science and Technology, Keio University

a) george@inl.ics.keio.ac.jp

b) kaneko@ics.keio.ac.jp

c) tera@ics.keio.ac.jp

列車ネットワークにおいてハンドオーバーごとに発生する通信切断時間を0ミリ秒から徐々に長くしていき、アプリケーションの動をビデオカメラにより動画として記録した。また同時にバケットもモニタリングし、トラフィック特性を解析した。記録した動画は以下のURLから参照可能である。

<http://www.inl.ics.keio.ac.jp/~tera/IA-201211/>
調査の結果、YouTubeでは切断時間が1,400ミリ秒を越えると、長時間の再生停止が発生することが分かった。Skypeについては、20ミリ秒程度の切断であっても、わずかな音声や動画が途切れてしまうことが分かった。さらに1,000ミリ秒切断すると、動画と音声の品質が低下することが分かった。ottava.jpについては、1,800ミリ秒の切断までなら音声は途切れずに再生できることがわかった。しかし上記の実験ではハンドオーバーの発生するタイミングが一定でないため、切断時間が同じであってもメディアの品質が悪化する場合とそうでない場合があった。アプリケーションが通信を行うタイミングによって、周期的切断の影響を受けるかどうかが変わるためであると考えられる。そのため、周期的切断と通信のタイミングについて、より詳細な解析が必要であると考えられる。

本稿では上記の3つのアプリケーションについて、周期的ハンドオーバーが与える影響をより詳細に調査した。そのため、アプリケーションの動作概要についてについて、利用プロトコルやメッセージシーケンスを明らかにした。また、周期的切断とメディア品質の関係を、アプリケーションのトラフィック特性から調査した。

2. 各アプリケーションの動作概要

本稿ではYouTube(蓄積動画の再生)、インターネットラジオ(高音質音楽放送)、Skype(テレビ電話)という特徴の異なるアプリケーションについて調査を行った。インターネットラジオ局としては、Ottava^{*1}を利用した。それぞれのアプリケーションの特徴を、表1にまとめた。以下本章では、調査対象としたストリーミングアプリケーションの動作概要とトラフィック特性の解析についての関連研究を示す。

2.1 YouTube

YouTubeはHTTP/TCPを使用してサーバから動画データをダウンロードしながら、再生する動画共有サービスである。動画を再生するとき、クライアントはまずポータルサイトから動画ページのHTMLデータをダウンロードする。HTMLデータには動画データのURLが含まれており、続いてクライアントは動画サーバから動画データをHTTPを利用してダウンロードする。

^{*1} CDを超える高音質でクラシック音楽を放送するインターネットラジオ局である。

YouTubeのトラフィック特性解析に関しては[3-6]など、多くの研究が存在する。文献[5]によるとYouTubeの動画データのダウンロードは2つのフェーズに分けられる。再生を開始するとまずある程度のデータ量を一気にダウンロードする。このフェーズをバッファリングフェーズと呼んでいる。続いてダウンロードを行わない期間(Off)と行う期間(On)が交互に現れる。このフェーズを定常状態フェーズと呼んでいる。On期間にダウンロードするデータ量をブロックサイズと呼び、OnとOffを繰り返すことによる平均転送レートが、動画の再生レートをわずかに上回るように調整している。

2.2 インターネットラジオ

インターネットラジオのストリーミング配信は、配信元やクライアントによって様々な配信形式が存在する。

本稿の調査環境ではOSとしてWindows7、WebブラウザとしてInternet Explore 9をクライアント環境に利用した。この環境でottava.jpを視聴すると、クライアントでは視聴時にWindows Media Playerが起動し、放送を聞くことができるようになる。通信にはトランスポート層プロトコルとしてTCPが使用されている。その上で、再生の開始や停止といったセッション制御にはRTSP(Real Time Streaming Protocol)[7]が使用され、音声データの送信にはRTP(Real-time Transport Protocol)[8]が使用されている。

我々が調べた限りでは、インターネットラジオのトラフィック特性を解析した研究は見当たらなかった。

2.3 Skype

Skypeはpeer-to-peer(P2P)技術によるテレビ電話のアプリケーションである。SkypeのクライアントはまずSkypeネットワークのスーパーノードと呼ばれる管理ノードと通信し、Skypeネットワークにログインする。2つのクライアントがSkypeで通話するときには、発呼ノードがスーパーノードとメッセージ交換し、着呼ノードのIPアドレスを発見する。その後両ノード間でUDPによって直接通話が行われる。さらに通話中にはTCPを利用してネットワークの状況が相手ノードに通知され、ネットワークの状況に応じてUDPパケットの送信レートを変化させる、通信経路をリレーノード経由に変更する、使用プロトコルをTCPに変更するなど様々な対応をとる。Skypeでは通信が暗号化されており、UDPやTCPの上位層で使用しているプロトコルは不明である。

Skypeのトラフィック特性解析に関しては[9-11]など、様々な研究が存在する。文献[11]によるとSkypeは以下のような特徴を持つ。Skypeはパケット損失率が10%を境にして動作が大幅に変化する。パケット損失率が10%以下の場合は送信レートはパケット損失率に依存せず、利用可

表 1 対象としたストリーミングアプリケーションの特徴

Table 1 Features of the streaming applications

アプリ	アプリの概要	プロトコル	インタラクティブ性	音質への要求	音声の途切れ	画質への要求	画像の中断
YouTube	蓄積された動画の再生	HTTP/TCP	不要	中程度	多少なら許容	中程度	多少なら許容
Skype	テレビ電話	独自/UDP, TCP	必要	会話に支障がない程度	許容されない	中程度	多少なら許容
Ottava	インターネットラジオ	RSTP/TCP RTP/TCP	不要	高音質を要求	許容されない	—	—

表 2 Skype の画像品質 (文献 [11] から引用)

Table 2 Video quality of Skype(from [11])

画像の解像度	640×480, 320×240, 160×120
フレームレート	5 – 30 fps (frame per sec)
画像のビットレート	5 – 950 kbps

能帯域が増加すると送信レートが増加する。一方、パケット損失率が 10%を超えた場合は大幅に送信レートを低下させ、利用可能帯域が増加しても送信レートは増加しない。Skype は様々な動画品質をサポートし、表 2 に示すように画質を変動させる。また SILK という音声コーデックを利用し、音声品質を変動させる [12]。

3. ストリーミングアプリケーションの解析

我々はまず調査対象のアプリケーションについて、通常時の動作特性を調査した。クライアントマシンを当研究室(横浜市港北区)に設置し、YouTube やインターネットラジオのサーバに接続し、さらにクライアントマシンの上流側に制御用 PC を設置した。この制御用 PC でパケットモニタリングを行い、アプリケーションのトラフィック特性を解析した。Skype に関しては、研究室内に設置した 2 台のクライアントマシンを利用した。これらクライアントマシンはスイッチを介して同一の IP サブネットに接続している。

続いて周期的ハンドオーバーがアプリケーションのトラフィックに与える影響を調査した。また本稿では、300km/h で走行する新幹線に 1 章で述べた赤外線通信装置で 1Gbps のインターネット接続を提供することを想定する。赤外線通信装置を線路に沿って 300m 間隔で配置すると仮定すると、3.6 秒に 1 回の割合でハンドオーバーが発生する。定期的な通信切断を発生させるため、制御用 PC で定期的に指定した時間パケットを廃棄するようにした。

3.1 YouTube

本稿の調査では、OS として Windows7, Web ブラウザとして Internet Explore 9 をクライアント環境に利用した。動画は再生時間 4 分 27 秒、再生ビットレート 551kbps のものを利用した。図 1 に YouTube の Web サーバに SYN メッセージを送信した時刻を 0 秒としたときの、動画パケットの総転送量の増加の様子を示す。

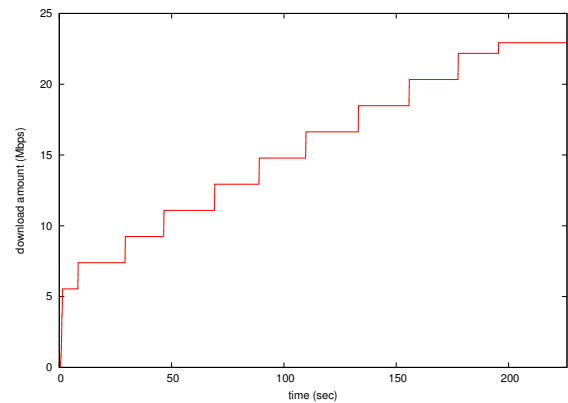


図 1 YouTube のデータ転送量の増加

Fig. 1 The Increase of the amount of transfer YouTube data

2.1 節で示したように、YouTube のダウンロードはバッファリングフェーズと定常状態フェーズに分かれることが分かる。クライアントは動画データのダウンロードのため、13 回 GET メソッドを送信していた。フェーズの違いはクライアントが GET メソッドの送信タイミングを調整することで実現していると考えられる。また 1-3 回目の HTTP トランザクション間隔は非常に短く、3-4 回目は約 7 秒、それ以降は約 20 秒間隔であった。HTTP トランザクションの時間は 170-600 ミリ秒間と短く、またダウンロードするデータ量は最後のトランザクション以外は 1.782 Mbyte と一定だった。1-3 回目のトランザクションがバッファリングフェーズ、それ以降が定常状態フェーズであると考えられる。3 回のトランザクションでダウンロードするデータ量と総データ量が 22.10 Mbyte であることから、初期バッファリング量は 64 秒程度であると推定される。また、クライアントは動画サーバとの間に最初、3 つの TCP コネクションを確立する。1-3 回目の HTTP トランザクションはそれぞれのコネクションで 1 回ずつ発生する。4 回目は 2 番目に確立したコネクションで、それ以降は最初に確立したコネクションでトランザクションが発生する。また、コネクションはクライアントから RST セグメントを送信することで終了していた。

次に、周期的ハンドオーバーとサーバからデータパケットが送信されるタイミングの関係を、途中で動画停止が発生せず再生できた場合と動画停止が発生した場合、それぞれについて図 2, 3 に示す。なお、切断時間は 1 周期 3.6 秒につき 2.4 秒である。図の色が濃い部分がハンドオーバーの発

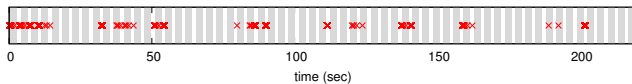


図 2 YouTube におけるハンドオーバーと動画データ送信 (停止なし)
Fig. 2 The timing of the video data transmission and handover (Good case)

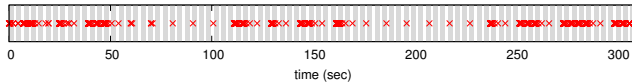


図 3 YouTube におけるハンドオーバーと動画データ送信 (停止あり)
Fig. 3 The timing of the video data transmission and handover (Bad case)

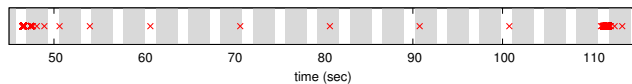


図 4 YouTube の動画停止時のトラフィック
Fig. 4 Traffics when YouTube movie stops

生時間を表す。

図を比較すると、停止なしの場合ではパケットの送信時間が集中している一方で、停止ありの場合ではパケットが送信された時間は広範囲である。特に 50–100 秒、160–230 秒では少数のパケットを長い時間かけて送信している。これは、再送に何度も失敗し、タイムアウト時間が非常に長くなってしまったためだと考えられる。さらに、動画データのパケットについて、パケットロスの状況を表 3 に示す。停止ありの場合の方が停止なしの場合よりもパケットロス率が高いが、著しく高いわけではない。また、連続してパケットロスが発生したときのパケットロス数については、停止なしの場合の方が大きい。動画が停止するほどタイムアウト時間が長くなるのは、同一パケットについて連続でパケットロスが発生することが原因でないかと考える。5 回再送タイムアウトに失敗すると、再送待ち時間が長くなり、データ送信のスループットが低下してバッファが枯渇すると考えられる。図 4 に停止時のトラフィックの様子を示す。パケット再送のタイミングが周期的切断発生時にぶつかり、再送を繰り返していることが分かる。今回の測定では解析していないが、クライアント側の TCP Ack のパケットロスも影響する。どちらにせよ、長時間サーバまでクライアントからの応答がない場合、応答があるまでサーバはパケットを送信しない。

3.2 Ottawa

Ottava の調査は 2.2 節に示したように OS として Windows7、Web ブラウザとして Internet Explore 9 を利用した。インターネットラジオを約 3 分間視聴し、測定した。

表 3 停止状況とパケットロス

Table 3 Video interruption and packet loss

	パケット ロス率 (%)	最大連続 パケットロス数	最大再送 回数	最長再送 時間 (秒)
停止無	1.9	113	4	5.12
停止有	5.6	93	5	41.2

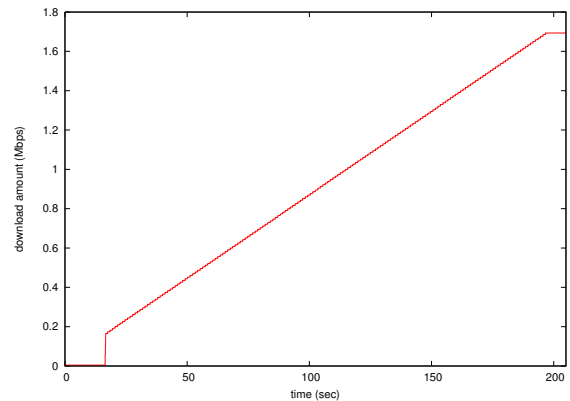


図 5 Ottawa のデータ転送量の増加

Fig. 5 The Increase of the amount of transfer Ottawa data

図 5 に動画サーバに対して RTSP メッセージを送信した時刻を 0 秒としたときの、音声パケットの総転送量の増加の様子を示す。YouTube 同様、バッファリングフェーズと定常状態フェーズが存在している。しかしライブ配信であるため、定常状態フェーズでは YouTube と比べるとデータ送信間隔は短く、約 1 秒間隔で 2,775 Byte の RTP メッセージを 3 回送信している。バッファリングフェーズで RTP メッセージを 54 回送信するため、最初のバッファリング量、は 18 秒程度と推定される。バッファリング量はライブ配信では遅延時間となるので、YouTube よりも少ない。また、RTSP セッション確立時にクライアントが送信する SETUP メソッドに対して、サーバが送信する Reply に `timeout = 60` の記述があった。RTSP セッション維持のため、クライアントは GET.PARAMETER メソッドを約 60 秒ごとに送信している。

次に、周期的ハンドオーバーとサーバからデータパケットが送信されるタイミングの関係を、途中で音声停止が発生せず再生できた場合と音声停止が発生した場合、それぞれについて図 6, 7 に示す。YouTube の場合とは異なり、停止ありの場合ではパケットの送信時間が集中し、停止なしの場合ではパケットが送信された時間は広範囲である。また停止した部分のトラフィックを調査すると、サーバからのパケット再送が 5 回失敗するとサーバが RST セグメントをクライアントに送信し、クライアントがこれを受信すると TCP コネクションと RTSP セッションを再確立し、再生を再開することが分かった。RST セグメントに対しては ACK が必要ないため、これをパケットロスするとクライアント側は 60 秒ごとに送信する GET.PARAMETER に

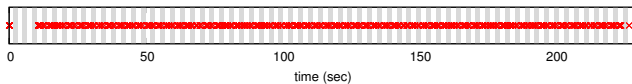


図 6 Ottawa におけるハンドオーバーと音声データ送信 (停止なし)
Fig. 6 The timing of the sound data transmission and hand-over (Good case)

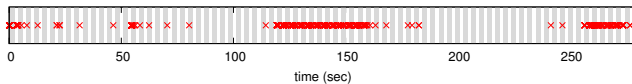


図 7 Ottawa におけるハンドオーバーと動画データ送信 (停止あり)
Fig. 7 The timing of the sound data transmission and hand-over (Bad case)

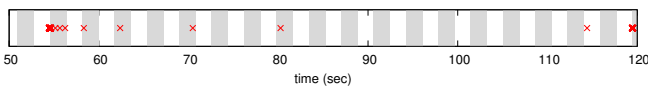


図 8 Ottawa の動画停止時のトラフィック
Fig. 8 Traffics when Internet Radio stops

対して RST が返信されるまでサーバがセッションを終了したことに気付かず、長い間再生が停止してしまう。図 8 に停止時のトラフィックの様子を示す。80 秒付近でサーバが RST セグメントを送信するもののパケットロスし、114 秒付近で RST を受信することで、通信再開できたことが分かる。

3.3 Skype

Skype の調査は OS として Windows7, Skype のバージョン 6.1.0.129 を利用した。測定しやすいように一方のクライアントを送信側とし、他方を受信側とした。送信側からは画面共有機能を利用して動画と音声、受信側からは音声を送信した。測定は送信側、受信側の両方で行った。

解析の結果、Skype で 2 つのノードが UDP を用いて直接通話をするときであっても、ノード間で TCP コネクションを確立することが分かった。ただ通話開始時と通話終了時以外はほとんど TCP で通信していないため、セッションの制御用と考えられる。SYN セグメントを送信した時刻を 0 秒としたときの、送信側から受信側へ送信した UDP パケットの総転送量の増加の様子を図 9 に示す。スループットは 1 秒間隔で測定した。図よりスループットは大きく変動することがわかる。Skype では音声データと動画データで別々にエンコーディングしている。そのため、音声と動画を別々で送信していると考えられる。音声はその途切れが通話に大きく影響すると考えられるため、パケットロスの影響を減らすために頻繁に送信する一方で、動画は多少停止しても通話への影響が小さく、動画データの送信間隔は広いのではないだろうか。動画データの送信間隔の大

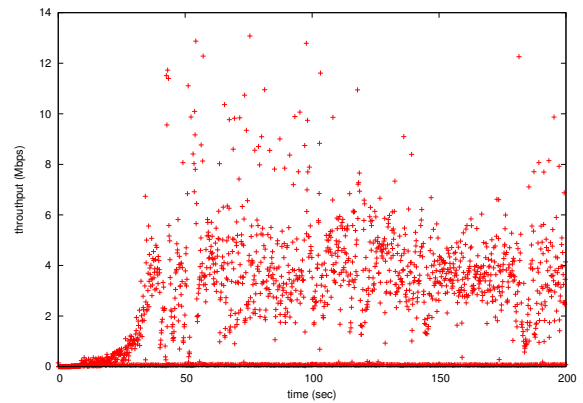


図 9 Skype のスループットの変動
Fig. 9 The variation of Skype throughput

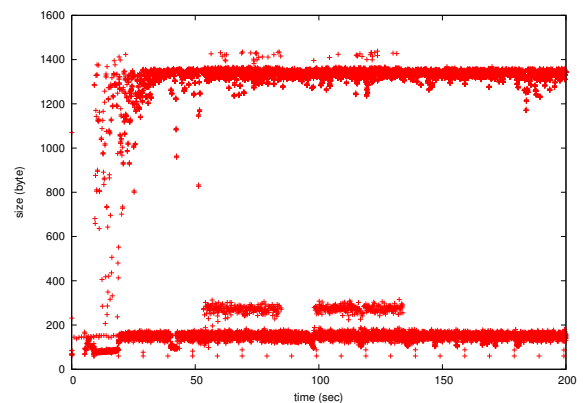


図 10 Skype のパケットサイズの変動
Fig. 10 Variation of packet size of Skype

きさから、スループットが大きく変動するのではないかと考えられる。また通話開始直後は画面共有機能を使っておらず、動画パケットがないために通信スループットが低い。また画面共有機能の利用開始直後もすぐスループットが高くなるわけではなく、徐々に上昇していることが分かる。

パケットサイズの変動の様子を図 10 に示す。パケットサイズが 150 byte 付近と 1,300 byte 付近で分かれている。これはそれぞれ音声データと動画データではないかと考えられる。また通信開始直後はパケットサイズが徐々に増加している。これによって、利用可能帯域以上の大量データ送信によるパケットロスを防いでいるのではないかと考えられる。通信開始直後は、TCP による通信も頻繁に行われており、通信状況の推定や通信相手への通知を行っているものと思われる。

次に、パケットサイズの変動を切断時間 0.6 秒、1.2 秒の場合、それぞれについて図 11, 11 に示す。メディアの品質についてみると、切断時間がいずれの場合も音声、動画の途切れが発生した。しかし、切断時間 0.6 秒の場合では画質、音質の低下は発生しなかったが、切断時間 1.2 秒の場合は画質の低下が発生した。図から切断なしの場合と比較すると、0.6 秒切断の場合のほうが、音声データも動画データも大きい。これはネットワーク状況の悪化によって

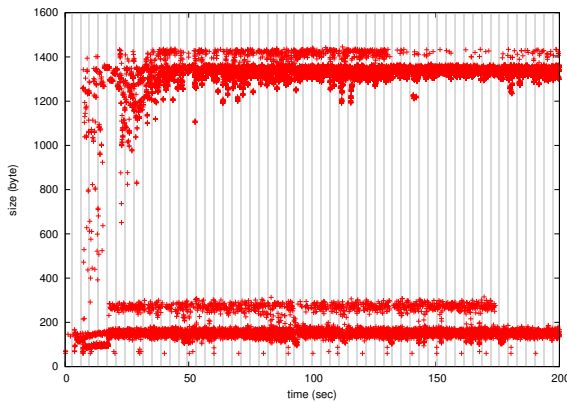


図 11 Skype のパケットサイズの変動 (切断時間 0.6 秒)

Fig. 11 Packet size of Skype (Disruption time: 0.6 sec)

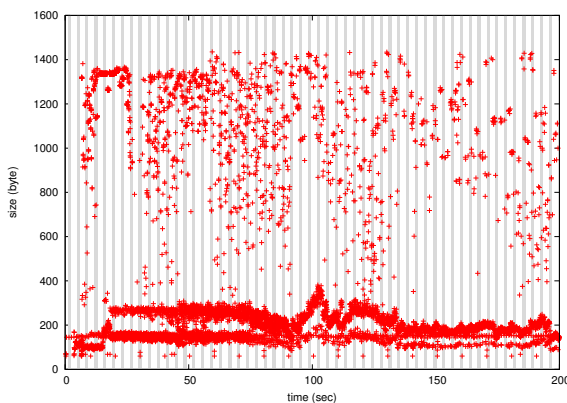


図 12 Skype のパケットサイズの変動 (切断時間 1.2 秒)

Fig. 12 Packet size of Skype (Disruption time: 1.2 sec)

FEC (Forward Error Correction) の誤り訂正率を向上させているからと思われる。一方、1.2 秒切断の場合、動画データのスループットは大幅に変動している。切断している間、クライアントはネットワーク品質が悪化していると判定して画質を悪いものに変更し、パケットサイズを減少させる。また切断が終わるとネットワーク状況が向上するので、動画品質とパケットサイズを元に戻す。これが繰り返して発生するため、パケットサイズが周期的に変動するのではないだろうか。また、音声データに関しては、あまり変動していない。これはパケットサイズが小さく、帯域を圧迫しないために切断が発生しても問題ない Skype が判定したためではないだろうか。このため、切断時間 1.2 秒の場合には画質のみ低下したと考えられる。また通話中の送信側から受信側への UDP パケットの送信スループットの平均を表 4 に示す。切断なしの場合と比べて、切断時間 0.6 秒の場合はパケットサイズの増加によってスループットも上昇しているが、切断時間 1.2 秒では著しく低下している。Skype はネットワーク状況に応じて戦略を変化させているといえるだろう。

4. まとめ

本稿では、YouTube、インターネットラジオ、Skype に

表 4 周期的切断と平均スループット

Table 4 Video interruption and average throughput

切断時間	0.0 秒	0.6 秒	1.2 秒
平均スループット (Mbps)	2.78	3.33	0.40

ついて、ハンドオーバーによる周期的切断が与える影響を調べた。YouTube とインターネットラジオでは連続してパケット再送に失敗すると、再送待ち時間の増加や RTSP セッションの終了により、動画や音声の長時間の停止が発生することが分かった。また Skype については、周期的切断の頻度により通信の方針が変化し、切断時間が長い場合にはパケットサイズが周期的に変動することがわかった。

今後は他のストリーミングアプリケーションについても、周期的切断によるメディア品質の変動やトラフィック特性を調べていきたい。

参考文献

- [1] Urabe, H., Haruyama, S., Shogenji, T., Ishikawa, S., Hiruta, M., Teraoka, F., Arita, T., Matsubara, H. and Nakagawa, S.: High Data Rate Ground-to-Train Free-Space Optical Communication System, *Optical Engineering*, Vol. 51, No. 3, pp. 031204-1 – 031204-9 (2012).
- [2] Arita, T. and Teraoka, F.: Providing a High-Speed Train with a Broadband NEMO Environment, *Proceedings of AINTEC 2010*, pp. 64–71 (2010).
- [3] Gill, P., Arlitt, M., Li, Z. and Mahanti, A.: YouTube Traffic Characterization: A View From the Edge, *Proceedings of IMC'07*, pp. 15–28 (2007).
- [4] Torres, R., Finamore, A., Kim, J. R., Mellia, M., Munafò, M. M. and Rao, S.: Dissecting Video Server Selection Strategies in the YouTube CDN, *Proceedings of ICDCS 2011*, pp. 248–257 (2011).
- [5] Rao, A., Lim, Y., Barakat, C., Legout, A., Towsley, D. and Dabbous, W.: Network Characteristics of Video Streaming Traffic, *Proceedings of CoNEXT 2011*, pp. 1–12 (2011).
- [6] Plissonneau, L. and Biersack, E.: A Longitudinal View of HTTP Video Streaming Performance, *Proceedings of MMsys'12*, pp. 203–214 (2012).
- [7] Schulzrinne, H., Rao, A. and Lanphier, R.: Real Time Streaming Protocol (RTSP) (1998). RFC 2326.
- [8] Schulzrinne, H., Casner, S., Frederick, R. and Jacobson, V.: RTP: A Transport Protocol for Real-Time Applications (2003). RFC 3550.
- [9] Baset, S. A. and Schulzrinne, H.: An Analysis of the Skype Peer-to-Peer Internet Telephony Protocol, *Proceedings of INFOCOM 2006*, pp. 1–11 (2006).
- [10] Bonfiglio, D., Mellia, M., Meo, M. and Rossi, D.: Detailed Analysis of Skype Traffic, *IEEE Transactions on Multimedia*, Vol. 11, No. 1, pp. 117–127 (2009).
- [11] Zhang, X., Xu, Y., Hu, H., Liu, Y., Guo, Z. and Wang, Y.: Profiling Skype Video Calls: Rate Control and Video Quality, *Proceedings of IEEE INFOCOM 2012*, pp. 621–629 (2012).
- [12] Spittka, J., Astrom, H. and Vos, K.: RTP Payload Format and File Storage Format for SILK Speech and Audio Codec, Internet-Draft draft-spittka-silk-payload-format-00, Internet Engineering Task Force (2009). Work in Progress.