

NAIST-IS-DD0361021

博士論文

ストリーミング配信システムのための
エンドユーザ視聴品質推定手法に関する研究

寺田 直美

2006年12月22日

奈良先端科学技術大学院大学
情報科学研究科 情報システム学専攻

本論文は奈良先端科学技術大学院大学情報科学研究科に
博士(工学) 授与の要件として提出した博士論文である。

寺田 直美

審査委員：

砂原 秀樹 教授

山口 英 教授

藤川 和利 助教授

中山 雅哉 助教授 (東京大学)

ストリーミング配信システムのための エンドユーザ視聴品質推定手法に関する研究*

寺田 直美

内容梗概

インターネットの広帯域化を背景に，高品質ストリーミングサービスへの要求が高まっている．中でも，増加し続けている商用利用においては，特に高いサービス品質が求められるため，課金時に備えて，クライアントにおけるサービス品質をサーバ側で正確に把握しておく必要がある．

ストリーミングサービスにおいて，クライアントのサービス品質を判断しようとした際に，定量的な品質判断基準はこれまで示されてこなかった．このため，ストリーミングでのサービス品質判断は，管理者の経験や勘に任される傾向があり，サービス品質判断が一定しないという問題があった．この背景には，ストリーミング通信やアプリケーションにおける詳細な内部解析が困難であることが挙げられる．ストリーミングサービスでは商用アプリケーションが主導的であり，実装の核になる部分の仕様については非公開となっている．通信手順や制御方式が部分的に非公開であることに加え，複雑な内部構造となっているため，出力される多数のログ項目と実際のサービス品質とが直結しにくい．また，実運用環境での利用を想定した場合，観測装置の設置，観測負荷による制約があるため，クライアント側での状態情報を直接収集することは困難である場合が多い．

本研究では，まず，クライアントにおけるサービス品質をアプリケーションの再生ステータスから定量的に判断するための客観評価基準を示した．クライアントでのサービス品質客観評価値と，クライアントネットワーク状態，サーバ側

*奈良先端科学技術大学院大学 情報科学研究科 情報システム学専攻 博士論文, NAIST-IS-DD0361021, 2006年12月22日.

で取得できる情報の3点での観測結果を分析することによって、サーバ側でアプリケーションから得られる限られた情報から、クライアントにおけるサービス品質およびネットワークの状態を細粒度に把握するための手法を提示し、実験により品質評価のための具体的な指標を明らかにした。これにより、個々のクライアントに対するサービス品質とネットワーク経路の状態をサーバ側で定量的に判断することが可能となった。管理者の主観に左右されず一定のサービス品質判断をサーバ側で行うことができ、複数の拠点に分散しているシステムにおいても統一的なサービス品質管理を実現できる。また、本研究で得られた結果から、これまで分析困難とされてきた仕様が非公開であるシステムや、詳細な観測が難しい実運用環境下でのネットワーク環境においても、限定的な外部情報を分析することによって新たな情報を抽出可能であることを示した。

キーワード

ストリーミング, 大規模配信, サービス品質, 運用管理, 管理支援

Study of Estimation of End-user QoS in Streaming Distribution Systems*

Naomi Terada

Abstract

Now that broadband Internet access has become widely available, there is ever-increasing demand for high-quality streaming services. Especially in commercial use, service managers are required to stringently manage service quality because server-side quality management is essential for accounting services. In streaming services, however, quantitative criteria of service quality still have not been revealed. Therefore, server managers need to estimate service quality experimentally or through intuition, which varies from one person to another.

A streaming system or protocol for in-depth analysis faces many problems because the streaming applications dominating the market nowadays are commercial softwares, for which the implementation specifications are not disclosed in detail. In addition, as multimedia applications generally behave in a complex manner, server log information does not directly describe real service quality. The service quality at each client is difficult to determine from the server-side because the number of network observation points is limited, and reducing server load or network load should be considered in live streaming systems.

In this study, first I provide an objective evaluation method for determining service quality and quality threshold at clients using an application's internal information that can be extracted from an additional module in the client application. Next, I propose a new method of extracting valuable information about

*Doctoral Dissertation, Department of Information Systems, Graduate School of Information Science, Nara Institute of Science and Technology, NAIST-IS-DD0361021, December 22, 2006.

service quality at clients and network quality between servers and clients only using limited information that can be obtained at the server side. This is achieved by analyzing the correlation of the following three indicators: “client service quality,” “network quality,” and “information at the server side.” I will then reveal which parameters indicate service or network quality and those parameters’ degree of quality in an experimental environment. The contribution is that server managers are able to estimate every each clientIndependent from heuristic judgment, constant estimation of service and network quality is accomplished, and service quality management of wide-area distributed system are also simplified with these common criteria. The results indicate that even if the target system is proprietary or a “black box” or includes limited observed data, valuable information can be extracted by analyzing the target in detail.

Keywords:

streaming, large-scale distribution, quality of service, management

目次

1. 序論	1
2. ストリーミング配信技術	5
2.1 ストリーミング配信形態と特徴	5
2.2 ストリーミング配信の運用管理上の難点	6
2.3 ストリーミングアプリケーション	7
2.4 ストリーミングファイルフォーマット	9
3. 大規模配信システムにおける運用情報管理とサービス品質管理関連技術	11
3.1 大規模配信システムの構成	11
3.2 広域負荷分散システムにおける運用管理情報	12
3.3 ストリーミングアプリケーションで得られる運用管理情報	13
3.4 ネットワーク観測による運用管理情報	15
3.5 クライアント側からの直接的な収集情報	17
3.6 サーバログの詳細な解析による運用管理情報	18
3.7 運用における要求事項と情報収集手法	19
3.8 視聴品質評価手法	21
3.9 研究目的と各手法の適応性	22
4. ストリーミングサービス品質管理における課題	25
4.1 ストリーミングサービス品質把握における問題点	25
4.2 エンドユーザ視聴品質に即したサービス品質評価	26
4.3 サーバ側でのクライアント配信品質把握	27
4.4 ストリーミングサービスに適した品質評価手法	29
4.5 サーバログ分析によるサービス品質管理	29
4.6 運用時のサーバログ有効性	31
5. ストリーミングサービス品質評価手法	33
5.1 サーバログからのサービス品質，ネットワーク状態推定	33

5.2	サーバログ	33
5.3	クライアントサービス状態	34
5.4	クライアント再生状態の分類	35
5.5	各状態滞留時間による再生品質評価	38
5.6	再生フレームレート下落の連続性による再生品質評価	39
5.7	実験手順	40
5.8	コンテンツ	42
5.9	実験結果	43
5.9.1	品質評価方法と目視によるコマ落ち観測結果の相関	49
5.9.2	ネットワーク状態とクライアント再生品質の相関	50
5.9.3	ネットワーク状態とサーバログの相関	50
5.9.4	サーバログとクライアント再生品質の相関	51
5.9.5	実験結果のまとめ	52
5.10	実験結果の考察と実験条件に関する議論	54
5.11	他のシステムへの適用性に関する議論	56
6.	得られた知見と課題	58
6.1	サーバログからのサービス品質情報抽出	58
6.2	サーバログからのネットワーク情報抽出	59
6.3	ネットワーク状態がサービスに与える影響	59
6.4	アプリケーション間，データ間の互換性	60
6.5	運用に対する知見	61
6.6	今後の課題	61
7.	結論	63
	謝辞	65
	参考文献	67
	著者研究業績	71

図目次

1	クライアント品質管理ポイント	19
2	セッション継続時間の累積頻度分布	31
3	サーバログ情報からのサービス受信状態の推定	34
4	配信中のフレームレート変動の例	36
5	クライアント再生状態遷移	37
6	フレームレート下落の連続性による評価	39
7	実験ネットワーク環境	41
8	目視によるコマ落ちの有無とネットワーク状態の相関	44
9	再生フレームレート平均値とネットワーク状態の相関	44
10	良好再生状態時間による品質評価	45
11	下落再生状態時間による品質評価	45
12	下落連続性 V_{down} による品質評価 ($m = 2$)	46
13	下落連続性 V_{down} による品質評価 ($m = 3$)	46
14	下落連続性 V_{down} による品質評価 ($m = 4$)	47
15	クライアント受信パケット数	47
16	クライアント喪失パケット数	48
17	クライアント回復パケット数	48
18	良好再生状態時間による品質評価	51
19	下落再生状態時間による品質評価	52
20	下落連続性 V_{down} による品質評価	53
21	リバッファ発生タイミングと継続時間	54

表目次

1	管理情報収集手法の精度と記録単位，運用への導入難度	20
2	クライアント品質評価手法	29
3	再生状態判断アルゴリズム	39
4	ネットワーク設定パラメータの値	42

5	エンコード設定	42
6	3回の実験結果の標準偏差と平均値に対する割合	54

1. 序論

インターネットを通じた動画配信は、コンテンツの充実に伴って利用者数が大幅に増加している。最近の映像コンテンツに関する報告では、44.9%のユーザが既に映像コンテンツを利用しており、23.4%のユーザが今後の利用を希望すると回答している [1]。コンテンツが増加し続ける動画配信では、有料の高品質サービスと無料の低品質サービスの二極化が起きている。無料で視聴できるもののひとつに挙げられる動画共有サイト youtube では、日本からの利用者数は700万人を突破したとの報告 [2] が出ている。現在著作権違反コンテンツの温床となっており大きな問題となっているが、この数字は逆に、コンテンツの種類によって動画コンテンツ利用者が潜在的に相当数あることを示しているともいえる。一方、有料サービスでは高品質化とコンテンツの充実がはかられている。「動画配信サービスのここ1年の変化」として、大画面で視聴可能な高画質で配信されるようになった点を挙げる人が22.7%となっている。また、視聴要望の高いコンテンツには新作映画、ドラマの順になっており、インターネットで配信された動画を大画面で視聴するというスタイルが広まりつつあることを示している。別の調査では光ブロードバンド利用者の動画配信契約率も昨年度比7.4%となっており [3]、高速回線利用者が契約時に有料契約を結ぶケースが増加していることが分かる。これは従来からのWeb、メールといったインターネット利用に加え、レンタルビデオや有料放送を視聴する代わりに動画コンテンツが視聴されていることを示している。また、携帯電話ではワンセグ対応端末が出てきており、携帯端末での動画視聴も一般化することが予想される。今後はこのような低ビットレートでの有料配信も増加すると考えられる。

こういった背景から、商用利用においては特に、安定的に高品質なサービスを提供することが必須となってきている。特に近年増加している商用利用においては、課金の際にサービス品質が問題となることがあり、サーバ側で個々のクライアントに対する配信品質を正確に把握しておく必要がある。本研究では、サーバ側における個々のクライアントに対するサービス品質情報把握を目標とする。

ストリーミングサービスにおけるサーバ側でのサービス品質把握には、大きく2つの問題があった。1つは、サービス品質そのものの評価基準が定まっていな

いことであった。一般に動画像の客観品質評価では、画像のノイズ比率など、画質そのものを対象としているが、ストリーミングサービスでは、ネットワークを経由するため、クライアントでの再生状態と再生ビットレートがパケット転送状態の影響で変動する。ストリーミングアプリケーションはネットワークの状態によって最適なレートを動的に選択する機構を備えているため、サービス提供者は複数のビットレートを1つのストリームに含めてエンコードを行うことが通例となっている。サービス品質は高ビットレートであるほど高いが、パケットロスによるフレーム落ちやデータリバッファのための再生中断などの再生状態も視聴感に影響する。また、ビットレート切替時にはリバッファによる再生中断が発生するため、視聴感が大きく劣化する。サービス品質判断では、これらの視聴品質に影響する現象を考慮する必要がある。

2つめの問題は、サーバ側で得られる情報からサービス品質やネットワーク状態を推測するための材料となる情報が不足している点であった。ストリーミング品質向上技術は、動画像圧縮技術、レート調整やエラーコントロールなどのアプリケーションレイヤ QoS、キャッシュやミラーリングなどの配信システム改善、サーバの性能向上、音声と画像のメディア同期、プロトコル改善の6つに大分される [4]。このうち、現行の商用システムにおけるサービス品質把握では、アプリケーションレイヤ QoS に分類される受信状態情報、プロトコル改善に分類される制御プロトコルによるフィードバックを用いることができる。これらはクライアントアプリケーション、サーバアプリケーション双方に標準的に組み込まれている機能であり、配信サーバではこれらの情報を全てのクライアントから収集しており、サーバログには個々のクライアント毎に受信状態が記録される。

サーバで記録されるクライアントでの受信状態は、視聴者が実際に体験する視聴品質とは異なる。実際の視聴品質とサーバ側で得られる受信状態情報との関連、またネットワーク状態がサービスに与える影響が不明確であるために、サーバ側で得られる情報の示す意味が読み手により様々に解釈されてきた。このため、サーバに集約された情報であるサーバログの、ログの値が示す品質の読み取りは管理者の経験的推量に大きく依存しているのが現状であった。

これら2つの問題を解決するために、本研究では、まず、クライアントアプリ

ケーション内部での再生ステータスを収集し、ステータス情報を用いた再生品質評価式を設け、実際の視聴品質との比較による検証を行った。その上で、サーバで得られる受信状態情報とクライアントで測定した視聴品質、ネットワーク環境との3点の関係を実験により明らかにした。これにより、配信時に直接収集が困難な個々のクライアントでのサービス品質である視聴品質と、サーバクライアント間のネットワーク環境を、サーバ側で得られる限られた情報から、高い精度で定量的に推量することを実現する。ログは個々のクライアント毎に記録されることから、サーバ側でサービス品質の正確な記録をクライアント単位で保持することができる。また、配信異常は早期に発見することによって、配信時のすばやい対処が可能となる。

サーバ側でのサービス品質把握は、サービス提供者にとって重要な課題ではあるものの、これまで行われた研究は非常に少ない。研究が進みにくい原因には、ストリーミングでは商用アプリケーションの市場占有率が非常に高く、詳細な仕様が公開されていない点が挙げられる。リバースエンジニアリングにより、詳細な分析を行った研究 [5] も存在するが、仕様が変更される度に再分析を行う必要があり、時間コスト、人的コストが高いという問題がある。しかしながら、ストリーミングアプリケーションに限らず、仕様が非公開である商用製品は多く、また、仕様は公開されていても、データ量が膨大である、暗号化されているなどの理由により内部構造からの詳細な分析が困難な場合も多々存在する。こういったシステムは増加しており、有益な運用情報を分析、抽出する手法は今後いっそう重要なものとなる。

本研究では、実運用環境下においてもシステム外部から得られる情報を収集し、運用時に必要な情報となるクライアントでのサービス品質やネットワークの状態情報との関連性を分析することにより、内部構造が不明なシステムにおいても有益な運用情報を抽出することができた。特に、現実的には困難であった、ネットワークを介した先のクライアントにおけるサービス状態、サーバクライアント間のネットワーク状態の高精度の推測を可能とした。これらにより、システムに負荷を与えることなく、サービス状態、ネットワーク状態をサーバ側で把握可能であり、管理側でのサービス品質の保証に貢献することができた。本方式では、シ

システムをブラックボックスとして取り扱うため、他の内部分析困難なシステムにおいても分析の可能性を示すことができた。

本論文の構成

1章では、ストリーミングにおけるサービス品質把握の重要性、ストリーミングアプリケーションにおける内部分析の困難性、本研究の位置づけについて述べた。2章では大規模ストリーミング配信システムの構成、特徴について述べる。3章では、ストリーミングアプリケーションの歴史的変遷、フォーマット、プロトコルと既存の品質把握手法について関連研究を示す。4章で本研究の目的として、クライアントにおけるサービス品質評価とサーバサイドでのログ解析について述べる。5章で実験における環境、詳細な手法、結果と考察を述べ、6章で得られた知見と課題として、本研究での仮説と検証した結果について、また、結果から得られた運用上の知見を示す。最後に7章で結論として研究の総括を述べるとともに、今後のストリーミングサービスおよび実運用環境下での情報収集、分析における展望を述べる。

2. ストリーミング配信技術

ストリーミング配信形態には，録画済みのコンテンツをクライアントからのリクエストに応じて提供するオンデマンド配信と，カメラ，マイクからの映像，音声を実時間エンコーディングするライブ配信の2つがある．ライブ配信では，過去にさかのぼって再度視聴ができないため，ハイライトシーンでの配信失敗は大きな問題となる．また，配信時間が限られているためにアクセスが集中し，帯域逼迫や負荷の上昇によりサービス品質の劣化を招きやすい．そこで本研究では，ライブ配信におけるサービス品質管理の向上を目指す．

大規模配信システムの運用情報管理では，従来から WWW サービスに関して多くの研究が行われてきた．ストリーミングのシステム構成や管理手法は WWW サービスと共通する点もあるが，データ特性の違いなどから，既存の手法をそのまま適用することが難しい．また，apache[6] が大きな市場占有率を占める WWW サービスと異なり，ストリーミングアプリケーションでは複数の商用ベースアプリケーションが普及しており，研究対象とする際に互換性が問題となる．そこで現在普及している複数のアプリケーションについて，通信方式，画像圧縮方式の側面から共通点，相違点を探る．

2.1 ストリーミング配信形態と特徴

ストリーミング配信形態は，大きく2つの方法がある．1つはオンデマンド配信で，ディスク上に保管した動画ファイルを用いる．サーバは，クライアントのリクエストにより適切なファイルを呼び出し提供する．動画データはあらかじめエンコードされている必要があるため，オンデマンド配信は非リアルタイムコンテンツに適している．

もうひとつはライブ配信で，生中継に用いる．動画はリアルタイムにエンコードされ，エンコードされたデータはサーバに転送される．ライブ配信では，オンデマンド配信と比較した場合，視聴者の動作によって引き起こされる運用上の難点が存在する．ライブ配信の場合，中継時間やハイライトシーンが限られているために短時間にアクセスが集中し，このときサーバはクライアントから膨大なサー

ビスリクエストを受け取ることになる。ライブ配信運用上の難点は、放送開始時やハイライトシーンにおけるバースト的なアクセスと、サーバやネットワークの高負荷によって引き起こされる予想外の障害にある。サーバへの過剰なアクセスはサーバ高負荷やネットワーク帯域逼迫を招き、これらは双方ともサービス品質低下の主な原因となる。ネットワーク負荷が高い場合、データパケットロスや遅延が発生し、クライアントアプリケーションでの品質に悪影響を与える。ネットワーク利用率が非常に高い場合は、パケットロスが発生し新規セッションが確立できなくなり既存セッションの多くがタイムアウトによる切断を起こす。これらは、メルトダウン現象とも呼ばれる。

ライブ配信ではアクセス集中による障害発生リスクが高いにもかかわらず、オンデマンド配信のように過去に配信されたコンテンツをさかのぼって視聴することはできない。このため、ハイライトシーンなどにおけるサービス品質管理への要求はより厳しいものとなる。本研究では、大規模ライブ配信を対象としてサーバ側でのクライアントサービス品質把握を目指す。

2.2 ストリーミング配信の運用管理上の難点

ストリーミングサービスは WWW などの他サービスと比較した場合、異なる側面を持つ。ストリーミングはリアルタイムアプリケーションであるため、データは再生タイミングまでに到着する必要がある。ネットワークにおけるデータ到着遅延が再生品質に影響しないよう、今日のほとんどのクライアントアプリケーションはデータバッファに先取りしたデータを蓄積している。いくつかのパケットが喪失し、クライアントに不着となった場合は、クライアントアプリケーションは喪失パケットの再送要求を行う。このメカニズムは特に低品質なネットワークにおいてサービス品質を向上させることに貢献したが、同時にサーバサイドでのサービス品質把握を困難にした。例えば、ネットワークパケットロスは必ずしもサービス品質劣化を意味しない。これは、データ再送が再生タイミングまでに成功すれば、クライアントは再生を滞りなく行えるからだ。

データフォーマット形式の複雑さもサービス状態把握を困難にしている。主なストリーミングアプリケーションフォーマットは MPEG-4 を用いている。MPEG

はフレーム間圧縮であり，データストリームは複数の GOP(Group of Pictures) に分けられ，その中にはキーピクチャと他の圧縮ピクチャが含まれる．これは，パケット喪失時にどのピクチャが失われるかによって画質への影響が変動することを示している．

全てのパケットをクライアントに近い部分で回収し，リアルタイムに再構築することができれば，サービス提供者はサービス品質を相当高い精度で把握することが可能である．しかしながら，この手法は実際の配信現場ではほぼ不可能である．なぜなら，クライアント数は非常に多く，配信ネットワーク上には膨大なデータパケットが存在するからである．何よりも，ほとんどのクライアントはインターネットを介した先にあり，コントロール不可能である．

2.3 ストリーミングアプリケーション

ネットワーク動画像配信の歴史は古く，1990 年にはストリーム利用を前提としたプロトコルである ST2(Internet Stream Protocol Version 2)[7] が提唱されている．しかし当時は帯域が狭かったこともあり，ストリーミングが広く利用されるようになるのは 1990 年代半ば以降になる．Xing Technology 社¹の StreamWorks，RealNetworks 社の Real シリーズ [8] などの商用アプリケーションが発表された後くらいから，イベントの大規模配信が行われるようになった．以降現在まで，ストリーミングアプリケーションは商用アプリケーションが多くのシェアを占める状態が続いている．1998 年にコロンビア大学，Netscape 社，RealNetworks 社により RTSP(Real Time Streaming Protocol)[9] が共同策定され，アプリケーションが対応するようになり，サーバクライアント間でのきめ細かな制御を可能にした．

ストリーミングアプリケーションは通例，RTP(Real-Time Transport Protocol)[10] でデータ転送を行い，トランスポート層では UDP を用いる．制御セッションは RTSP(Real Time Streaming Protocol) で，トランスポート層は TCP を用いる．データ転送に UDP を用いることで，多くのデータを少ないオーバーヘッドで転送することができ，リアルタイム性を確保しやすくする．また，制御用に TCP を用いることで，再生，停止手続き，再送要求，レポートなどを高い信頼性で行

¹1999 年に RealNetworks 社に吸収合併

うことができる。RTSPではテキストベースの情報を交換できるため、サーバアプリケーションとクライアントアプリケーション間で様々な状態情報を通知することが可能である。ファイアウォールの制限などによりRTP、RTSPによる配信が不可能な場合は、HTTPによるデータ転送および制御をほとんどのアプリケーションがサポートしている。ファストスタートにより全データの到着を待たずに再生できるが、ストリーミングのような制御は行えないため、疑似ストリーミングとも呼ばれる。

現在、ストリーミングアプリケーションは企業ベースのソフトウェアが中心となっており、Microsoft社のWindows Media Technology[11]、RealNetworks社のRealシリーズ、同社のオープンソースプロジェクトによるHelix Universalシリーズ[12]、Apple社のQuickTimeシリーズ[13]、同社のオープンソースプロジェクトによるDarwin Streaming Server[14]などがある。Windows Mediaはストリーミングプロトコルに独自フォーマットであるMMS(Microsoft Media Server)を用いていたが、Windows Media Technology 9からはRTP、RTSPに対応した。Windows OSに再生アプリケーション、Windows 2003などのサーバOSに配信アプリケーションが標準搭載されていることなどから、近年急激に普及した。Realシリーズではデータ転送でRTPの代わりにRDT(RealNetworks Data Transport)と呼ばれる独自フォーマットを用いてきたが、RTPにも対応しており、制御にはRTSPを用いる。ストリーミングの初期からアプリケーションをリリースしており、特に低ビットレートでの配信でのシェアが高い。QuickTimeシリーズでは、QuickTime形式が動画像ファイルフォーマットの国際標準としてISO(International Organization for Standardization)に採用されるなど、ファイルフォーマット、転送形式とも3社の中で最も国際基準に沿ったものとなっている。データ転送にはRTP、制御にはRTSPを用いる。近年リリースされるバージョンでは、他社アプリケーションのフォーマットもサポートするなど、相互運用性が徐々にではあるが向上する傾向にある。

2.4 ストリーミングファイルフォーマット

ストリーミングファイルフォーマットは、動画像ファイルフォーマット(動画像コーデック)、音声ファイルフォーマット(音声コーデック)と、それらを包含するマルチメディアコンテナフォーマット(コンテナフォーマット)で定義できる。

動画像コーデックには、ISO、IEC(International Electrotechnical Commission)で規定された MPEG-1、MPEG-2、MPEG-4 や ITU-T(International Telecommunication Union Telecommunication Standardization Sector) 規定の H.261、H.262、H.263、H.264²などがある。初期のストリーミングアプリケーションは MPEG-1、MPEG-2 や H.261、H.263 に準拠したフォーマットが主流であったが、MPEG-4 では高い圧縮率と、低ビットレートから高ビットレートまで広くサポートしていることなどから、ストリーミングに用いる動画像コーデックは MPEG-4 ベースになりつつある。MPEG-4 は 2 つのフォーマットが流通しており、H.263 ベースで 1999 年に規定された MPEG-4 Part 2³と、2003 年に ITU-T と共同で規定された H.264(MPEG-4 Part10⁴) がある。特に H.264 では MPEG-2 と同等の画質で 2 倍以上の圧縮を行うことができるため、Blu-ray ROM や HD DVD、1 セグメント放送などに採用されており、高画質配信での利用が期待される。動画像コーデックでは、MPEG や H.261 ~ H.264 のようなフレーム間での差分を利用したフレーム間圧縮方式の他、フレーム内圧縮を用いた Motion JPEG などの形式もある。

音声コーデックは ISO/IEC の MPEG-1 Layer3(MP3) や AAC(Advanced Audio Coding)、iLBC(Internet Low Audio Codec) などがある。iLBC は Google Talk や Skype などの音声通話に採用され、ストリーミングでは AAC や MP3 ベース、企業独自のプロプライエタリなフォーマットなどが使用されている。

コンテナフォーマットには、ISO/IEC の MP4、Apple 社の QuickTime、Real Networks 社の RealMedia、Windows 標準搭載の AVI(Audio Video Interleave) や Microsoft 社の ASF(Advanced Systems Format) などがある。このうち、AVI はストリーミングに適さないため、Windows Media Technology では ASF を採用している。

²ISO/IEC と共同策定のため MPEG-4 AVC と同義

³あるいは MPEG-4 Visual

⁴あるいは MPEG4-AVC

いずれのストリーミングアプリケーションも複数のコーデックやコンテナフォーマットを選択可能であるが、QuickTime は動画コーデックに H.264、音声コーデックに AAC、コンテナフォーマットに MP4 と ISO/IEC 準拠フォーマットが標準的である。Real シリーズでの RealVideo は、バージョン 8 までは H.263 ベースの動画コーデックであり、バージョン 9 以降はこれをベースにしたプロプライエタリなフォーマットとなっている。H.263 は MPEG-4 Part2 のベースであり、MPEG-4 との共通部分は多い。Windows Media Technology では動画像コーデックに VC-1⁵を用いる。これは Windows Media Video 9 の規格を拡張したものを SMPTE(Society of Motion Picture and Television Engineers) が規格化したもので、MPEG-4 ベースである。Microsoft 社の MPEG は ISO/IEC の MPEG-4 Part2 をベースに独自開発されたため、MS-MPEG4 と呼ばれており、現在は ISO/IEC 完全準拠の MPEG-4 動画コーデックも開発されている。VC-1 は H.264 と同程度の圧縮率、同ビットレートで同等の画質性能であるとされる。

ストリーミングアプリケーション毎に採用している動画コーデック、音声コーデック、コンテナフォーマットは異なるが、動画コーデックは MPEG-4 とその亜種といえる。これらはいずれもフレーム間圧縮方式であり、通信時にフレームが失われた場合は前後のフレームに影響する。

⁵2005 年に規格化作業完了、SMPTE 421M として発表

3. 大規模配信システムにおける運用情報管理とサービス品質管理関連技術

アクセスが集中する大規模ライブ配信では、データセンタ内などの狭域、プロバイダ間などの広域の両方で負荷分散を行う場合が多い。運用上のポリシーや構成の異なる複数地点からの情報を管理する点、配信負荷の高いシステムからの情報を抽出する点から、現実的に収集可能な情報は限られる。広域負荷分散システムにおいては、システム状態の管理技術が開発されており、運用上必須である情報を管理者に提供するが、本研究の目的であるクライアントサービス品質管理に必要な情報を得るためのものとは異なる。そこで、既存の運用情報収集手法を挙げ、実運用における要求とストリーミングのデータ特性という視点から有効性について検証する。また、サービス品質管理を行うために、クライアントにおける視聴感を定量的に測定する手法についても考慮する必要がある。そこで、既存の視聴品質評価手法とストリーミングに適用する際の問題点について述べ、各手法と本研究の目的との適応性について考察する。

3.1 大規模配信システムの構成

サーバ負荷の逓減は、ロードバランサ等を用い複数台のサーバにアクセスを分散させるローカル負荷分散によって可能である。この場合、サーバ群はネットワーク上のごく限られたエリアに設置されるため、帯域負荷の逓減は期待できない。また、サーバに近い経路上での障害や、ラック、データセンタ規模で起きる障害に対して耐性が低い。これらは、配信が大規模かつサービスが必須のものであるほど問題となる。クライアント数が多いほど帯域負荷は高くなり、サービス要求が高ければより高い耐障害性が求められる。

これらの問題に対する解決策として、広域負荷分散システムがある。複数のサーバ群をネットワーク上の離れたエリアに分散配置することでクライアントからのアクセスを分散させ、アクセスに対するサーバの割り当ては、DNSなどを用いてネットワーク距離が近くなるように構成される。このため、ネットワーク上の一点にアクセスが集中せずネットワーク負荷の逓減が実現できる。また、サー

バククライアント間のネットワーク距離が短縮されるため、遅延縮小などの効果も得ることが期待できる。広域負荷分散では異なるサイトにまたがってサーバ群を配置することが多く、運用上それぞれの管理ポリシーを逸脱しないことが求められる。このため、配信システム全体に変更を加えることは難しい。また、運用支援という目的であっても、業務上の守秘事項を含むデータの外部持ち出しや分析は困難であり、例えば通信内容そのものから運用状況を解析する手法は適用しにくい。

以上のことから、広域負荷分散システムの運用管理では、なるべく配信システムに変更を加えず、かつ通信内容を含まない最小限の情報から解析を行う必要がある。

3.2 広域負荷分散システムにおける運用管理情報

広域負荷分散システムである CDN(Contents Delivery Network) では、サーバへクライアントを割り当てる際に、負荷情報やキャパシティに関する情報が必要となる。また、安定運用のために、多数のサーバ群の動作情報を収集し、サーバに対して監視サイドから遠隔で操作する必要がある。このため、地理的に離れた場所に分散する多数のサーバ群から状態情報を収集、コントロールするための統一的なシステムが求められる。

CDN の最大手、akamai が開発したシステムでは、インターネット上の各所に配置されたエッジサーバ全てでモニタプロセスを起動する。モニタプロセスはサーバ上のアプリケーションを監視し、CPU、ディスク、メモリなどリソースの使用状況、定期的チェックによるアプリケーションの動作確認、アプリケーションプロセスからの定期的なレポートイベントの検出を行う。モニタプロセス情報は、ロードバランスシステムと akamai reporting system に送られ、負荷分散やシステム全体のチェックに用いられる。reporting system は全てのエッジサーバから収集した情報をまとめ、管理者に提供する。CDN での運用管理情報は、CDN システム自体の動作をスムーズに行うためのものであり、その上のアプリケーションの動作状況は各アプリケーションで考慮する必要がある。

広域負荷分散システムでは、配信拠点であるサイト内での管理と配信システム

全体の管理の両方が必要となる。サイト内では、スイッチによるサーバ負荷分散を行う場合、負荷の監視やアクセス状況の監視を行う。

ネットワーク監視項目は以下のものが中心となる。RTT (Round Trip Time) は TCP パケットを送信して応答を受け取るまでの時間を指す。TCP パケットを受信したホストでの処理時間、応答メッセージを送信する時間も RTT に含まれる。そのため、ホスト間で通信を行うときの実質的な応答時間の目安となる。

パケット損失は通信中にパケットが喪失することを示す。パケット損失が発生すると TCP の場合は再送処理が行われる。送出パケット数に対するパケット喪失数をパケット損失率とし、ネットワークの信頼性を表す指標となる。

スループットはネットワークが単位時間あたりに転送できるデータ量を表す。ストリーミング配信の場合はスループットにあわせて配信画質を選択するため、配信時に重要な要素となる。

3.3 ストリーミングアプリケーションで得られる運用管理情報

ストリーミングアプリケーションでは、サーバクライアント間での制御セッションに用いる RTSP からアプリケーションの状態情報を得ることができる。RTSP はアプリケーション、特にクライアント再生アプリケーションでの様々なイベントを通知する。配信開始時の SETUP、再生開始時の PLAY、再生を一時停止したときの PAUSE、再生終了時の TEARDOWN が基本的な制御コマンドになる。RTSP を用いたストリーミングでは、コンテンツの特定の場所から再生することが可能なため、VoD の場合は早送り (途中をスキップして再生)、巻き戻し (過去にさかのぼって再生) ができる。ライブ配信の場合はリアルタイムにエンコードされたデータを配信しているため、早送り、巻き戻しは不可能である。サーバ側でこれらのイベントを記録することにより、エンドユーザのアクションを検出したり、ネットワーク環境不良による再接続回数や要した時間などを知ることができる。

ストリーミングアプリケーションは通信状態に応じて、エンコードされた何通りかのビットレートの中から、最適なビットレートを選択する機構を持つ。最適レートのネゴシエーションは配信開始時に行われる他、定期的に通信状態を監視

することにより、通信状態の変化に応じて途中で配信ビットレートを変更する場合がある。配信ビットレートは配信品質を知る上で重要な情報であり、サーバ側ではクライアント単位で配信ビットレートに関する情報を記録することができる。

UDP によるデータ配信ではフィードバックが返らないため、途中でデータパケットが喪失した場合の再送要求を送受信することが困難である。クライアント再生アプリケーションはバッファを保持しており、一定時間内に再送を成功させることができれば、再生画質への影響を避けることができる。このため、リアルタイムアプリケーションとはいえ、再送制御は非常に重要である。クライアント再生アプリケーションでは、RTSP を用いて受信したパケット情報や喪失パケット情報をサーバに通知し、再送制御を行っている。サーバアプリケーションでは送出パケットに関する情報を保持しているため、クライアントから RTSP を介して得られる受信パケット情報と比較することにより、途中ネットワークの状態、クライアントでの再生状態を大まかに推測することが可能である。

通信状態を示すこれらの情報に加え、再生時のデータデコードに必要なスペックを考える際、クライアント性能やソフトウェア情報が必要となる。ストリーミングアプリケーションは、クライアントの OS やソフトウェアバージョン、およびその CPU 性能などの情報をサーバ側に通知する。サーバではクライアント単位でこれらの情報を保持する。

サーバアプリケーションは、RTSP セッションから得た多くの情報を、配信単位毎にログに記録する。ログにはクライアント IP アドレス、ホスト名、イベントや時間など種々の通信記録、ビットレート、パケット送信数、パケット受信数、クライアント情報などが記録される。ログ情報は配信品質を知るために非常に有用な情報であるが、50 項目以上にも上るこれらの項目の値が、配信時のネットワーク状態や実際の配信品質をどう表すかについては、統一的な目安は存在していない。その結果、運用現場では経験豊富な管理者の勘や経験的推量、過去のデータによる推量に頼っているのが現状である。

3.4 ネットワーク観測による運用管理情報

WWW サービスにおいては、サーバ付近でのネットワーク観測によってセッションの詳細な状態情報を収集することが可能である。WWW サービスで利用する HTTP は、下位プロトコルに TCP を用いているため、クライアントにおけるパケット転送状態が逐一サーバに通知される。このため、実運用環境下で設置可能なサーバ付近の観測点において、収集したパケットを分析することにより、通信に関する非常に詳細な情報を得ることができる。運用時には、サーバ内でパケットを収集した場合にはサーバ負荷を上昇させるという欠点があることから、サーバ直近のネットワーク機器でパケットの複製を行い、パケット収集サーバでデータを蓄積、分析して観測するパケットモニタリングが行われる。一方、ストリーミングサービスにおいては、リアルタイム性を優先するためデータ転送にはトランスポート層に UDP を用いることが一般的である。このため、通信の大半を占めるデータパケットに関しては、TCP の ACK に相当するクライアントからの応答が得られない。運用環境で設置可能であるサーバ付近の観測点で収集したパケットからは、データパケットの転送状態を知ることができないため、ストリーミングにおけるパケットモニタリングは有効性が低い。また、収集したパケットは、クライアントのプライバシーに関わる非常に多くの情報を含むため、情報の取り扱いには十分に注意を払う必要があり、商用利用の場合は特に外部に渡すことが困難である。このため、複数の配信拠点を持つ広域負荷分散システムを採用している場合には、情報のフィルタを行わず相互にこれらの情報を交換することは難しい。

アプリケーションレイヤまでパケットを展開することにより、データ到着状態の解析を行う手法もある [15]。この手法では、ネットワーク上に観測地点を設置し、通過するパケットのデータフレーム ID を観測する。ID は連続した番号が記録されているため、どのパケットが喪失したかを非常に正確に知ることができる。この手法では、通過する全てのデータパケットを上位レイヤまで展開しているため、観測機器の負荷が高くなる。また、アプリケーション実装や動画像、音声コーデックが異なる場合は新たに分析のためのフィルタを用意する必要がある。

これら UDP データパケット観測に対して、TCP セッションによって通信され

る制御ストリームである RTSP の観測を行う方法がある。これはクライアントからのフィードバック情報を含むため、サービス品質を知る上での有用性が高い。クライアントからのフィードバック情報は、クライアントにおけるパケットの受信情報である。本研究では、実験の段階で RTSP プロトコル、また、アプリケーション (Windows Media Player) が以前に採用していた mms プロトコルに対し、パケットモニタリングによって状態情報を収集できることを確認している。しかし、これらはアプリケーション実装に依存する点があることと、観測対象である接続クライアント数が増加した場合、観測するセッション数も増加するため観測が容易でなくなる点が難点である。RTSP セッションから得られる情報はセッション切断時にログに記録されるため、リアルタイム性は低下するが、ログから同じ情報を得ることができる。

パケットモニタリングは、ネットワーク観測手法としては一地点でのパッシブ観測に分類される。パッシブ観測によるより正確なネットワーク性能の計測には、ネットワークの入口部分と出口部分に観測地点を設置し、2 点間でのデータ比較が必要となる。既存研究では、キャンパスネットワークの出入口での観測 [16] といったように、ネットワークの集約点で観測するケースが多いが、本研究の目的である各クライアントにおけるサービス品質観測およびネットワーク状態把握を行うためには、全てのクライアントに対して観測地点を設置する必要がある。

ネットワーク上のデータを観測するのみのパッシブ観測に対し、データを送出してネットワークの状態を自ら計測する手法がアクティブ観測である。アクティブ観測では、プローブパケットと呼ばれるテストパケットを送出し、応答状況によりネットワーク性能を計測する。アクティブ計測に関する研究は古くから進められており [17]、必要な情報を直接収集できるため有効性が高い。アクティブ観測には、定期的にパケットを送信するものと、大量のトラフィックを送信するものの 2 つがあるが、後者は主にネットワーク性能測定に用いられ、運用中に利用可能であるのは前者になる。ストリーミング制御に用いられる RTSP は、定期的にクライアントに対しテストパケットを送出しており、レスポンス状態を観測していることから、広義でのアクティブ観測の一種と考えることもできるが、一般的にはアクティブ観測は運用時には負荷を上昇させる可能性があるため難しいと

される。

3.5 クライアント側からの直接的な収集情報

クライアントにおけるサービス品質を収集する手段として、最も直接的なのはクライアントから状態情報をフィードバックさせる手法である。具体的には、フィードバック機構を組み込んだクライアントアプリケーションを配布する、あるいはフィードバック部分をプラグインとして組み込むという方法がある。サービス品質情報はサーバ、あるいは情報収集サーバに集約され、管理者が得ることが可能である。また、視聴者からのフィードバックという方法もある。

実環境下で、クライアントに再生品質をサーバサイドにフィードバックするような実装を行い、ネットワーク状態との連動を長期間調べた研究 [18] がある。ここでの実装はネットワークの状態との相関を得るために利用されている。アプリケーションは実験当時非常に大きなシェアを占めていた Real Player をターゲットとしており、オンデマンド配信を対象としたものである。

クライアントで品質を分析、サーバ側へフィードバックを行うことで品質管理を行うためのフレームワークを提示したものがある [19]。この方式ではリアルタイムにクライアントでの再生品質を収集するため、最も正確なサービス品質をサーバ側で得ることができる。

これらの手法では、アプリケーションに変更を加える必要があるため、配布、インストールコストが高いという問題がある。特に、不特定多数を対象とする大規模配信で全てのクライアントに変更を加えることは現実的ではない。また、クライアントの情報を収集するにあたり、プライバシーに関する問題も発生する。

エンドユーザの視聴感とアプリケーションでの再生状態による客観評価により、サーバへのフィードバックを行う研究もある [20]。この方式では設置クライアントから直接サーバ側にフィードバックするため、配信品質の管理には有効な方法である。しかし、本研究では全てのクライアントを対象にすることを指すため、目的が異なる。

3.6 サーバログの詳細な解析による運用管理情報

サービス提供状態把握のための最も一般的な手法に、サーバログ解析がある。しかし、ストリーミングサービスの場合、ログ解析によるサービス状態の把握には次に述べる2つの問題点がある。

1つは、ログはサーバプログラムやクライアントプログラムの挙動を示すものであって、エンドユーザの視聴感を直接表現するものではないことである。サーバログには、サーバにおけるサービス状態とクライアントから返されたいくつかのアプリケーション状態情報が記録される。しかし、これらの指標値とエンドユーザの視聴感の関係が明確ではない。そのため、多くのサーバ管理者はクライアント側での受信品質に関する定量的な評価基準を持っていないのが現状であり、配信現場において、サービス品質は管理者の経験と勘により判断される場合が多い。

もう1つの問題点は、サーバログに記録される項目のうち、サービス品質を示すといわれている項目が示す値と、実際のサービス品質との間に隔たりがあることである。例えば、Windows Media Server のログには (c-quality) という項目が存在する [21]。この値はサーバが送信したデータのうち、クライアントが受信できた確率を示しており、値が十分高ければクライアントは良い品質のサービスを受信していたと考えるのが一般的である。しかし、我々の行った予備実験では、サーバログでは c-quality を 100% と記録しているにもかかわらず、クライアントでは目視によるフレーム落ちが観測された場合があった。このような例では、実際のサービス品質は必ずしも良好ではないにもかかわらず、サーバログからは受信品質が非常に良好であると判断しがちである。

サーバアプリケーションログおよびサーバへのアクセスログから、アクセスパターンの分析を行った研究は数多い。実環境下で収集したデータを用いてストリーミングトラフィックの特徴を分析したもの [22] や、ファイルの更新によるアクセスパターンの変化に着目したもの [23]、接続元分析を行ったもの [24] やコンテンツタイプに着目したアクセスパターンの違い [25]、オンデマンドコンテンツとライブコンテンツとの比較 [26] など様々である。あるいは、実際にダイヤルアップによりインターネット上の複数箇所から接続し、アクセスパターンやログの解析を行った研究がある [27]。実環境下で、広域での実験結果で有効性は高い

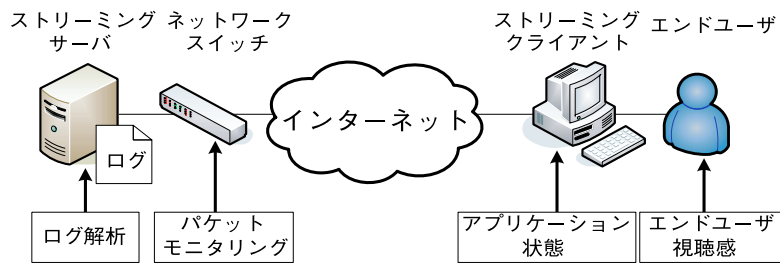


図 1 クライアント品質管理ポイント

が、実験年が古いため 64kbps での接続実験であることと、オンデマンドコンテンツによるものであるため、今日の高速回線を利用した高ビットレートでのストリーミングにそのまま適用することは難しい。また、本研究はライブ配信を前提としており、ターゲットが異なる。アクセスパターンを知ることは、サーバ負荷予想やキャパシティプランニングには有効であるが、クライアントサイドでの品質を知ることは難しい。

クライアントにおける受信品質とネットワーク環境の相関を調べた研究 [28] は本論文でのアプローチと基本的なアイデアは類似している。リバッファ回数とリバッファ時間をクライアントにおける品質の指標としており、ログ中のクライアント喪失パケット数を品質判断の指標としている。しかし、研究に先立って予備的に行った実験からは、リバッファによる再生中断が起きる時点までに、パケット不着によるフレーム落ちが頻発していることが分かった。また、品質の分析粒度が荒いために、ネットワーク環境の変化が配信状態へ与える影響も明らかにはなっていない。本研究では、リバッファの観測だけではクライアントのサービス品質を判断するには不十分であり、ネットワークの影響を導くにも不十分であると考え、クライアントにおけるフレーム変動を定量化した。これにより、視聴感に即したサービス品質とネットワーク状態をより詳細にとらえることを行った。

3.7 運用における要求事項と情報収集手法

ネットワーク状態の計測手法には、アクティブ計測とパッシブ計測の二つがある。アクティブ計測とは、測定のためのパケットを送出し、応答状況からネット

表 1 管理情報収集手法の精度と記録単位，運用への導入難度

		精度	導入	クライアント単位の記録
ネットワーク観測	Passive			
	Active		×	×
情報収集モジュール				×
サーバログ				

ワーク情報を得る手法である。一方，パッシブ計測とは，測定時に特別な動作を行わず，既存パケット状態からネットワーク情報を得る手法である。アクティブ計測では直接的に知りたい情報を収集することが可能であるが，測定パケットの送出によりネットワーク状態に影響を与える可能性があり，負荷の上昇を引き起こすなどの欠点がある。

これと同じように，実環境下での運用管理情報を得る方法は，アクティブ手法とパッシブ手法の二通りに分類することができる。すなわち，アクティブ手法とは，運用管理情報を得るために，配信システムやネットワークに何らかの変更や影響を与える手法と定義する。パッシブ手法とは，配信システムやネットワークに変更や影響を与えることなく，管理情報を収集する手法と定義する。計測手法の場合と同じく，アクティブ手法では必要とする情報をより直接的に収集することが可能であり，精度の高い情報を収集することができる。具体的には，ストリーミングクライアントアプリケーションにフィードバック機構を実装する手法，クライアントサイドのルータにパケット再構築のための機構を導入する手法などがある。こういった手法では個々のクライアントにおける再生状態を高精度に，サーバサイドで把握することが可能である。

しかしながら，実環境下では，アクティブ手法は運用上の要求に沿わないため，多くの場合導入することが困難である。運用上の要求とは，まず，運用管理情報収集が運用自体に影響を与えないことが大前提となる。そのためには，システム構成を変更しないこと，情報収集による負荷を与えないこと，アプリケーションやネットワークの挙動が不安定となる要素を持ち込まないことが挙げられる。パッシブ手法では，アプリケーションに変更を加えるため動作が不安定になる可能性

や、途中ルータの負荷を上昇させる可能性がある。

また、実運用環境、特にライブ型の大規模配信では非常に多数のアクセスが見込まれる。限られたクライアントにフィードバック機構を導入したり、特定箇所ルータからのみ情報を収集することにより、配信状態の概要は把握できる可能性はある。しかしながら、本研究で解決すべき問題は、個々のクライアントに対するサービス品質の把握とサーバクライアント間ネットワークの状態である。もしアクティブ手法で必要な配信情報を収集するとすれば、クライアント単位で情報収集機構を導入する必要がある。

3つ目に、システム変更を加える場合には、配布、導入コストが高くない必要がある。クライアントアプリケーションに変更を加える場合、配信先を限定しない配信形態では一般の利用者に特殊なアプリケーションをダウンロード、インストールしてもらう必要があり、難点が多いといえる。特にフィードバック機構のみの変更であれば、そのためだけにアプリケーション導入を試みる視聴者の割合は未知数である。ルータへの変更についても、ネットワーク上に存在する様々な機器に対応したシステムを逐次開発、導入する手数は非常に大きいといえる。

加えて、インターネットの特性上、サーバクライアント経路上に存在する機器への変更は非常に難しい。配信ネットワークが単一の管理主体の支配下にあるモデルの場合はシステムへの統一的な変更は可能であるだろうが、広域分散システムでは複数の管理主体が混在しており、完全な囲い込みは困難である。

以上から、アクティブ手法は有効な情報収集手段ではあるものの、実環境下での導入は現実的でないといえる。各手法における観測点を図に、表1に各手法の情報精度と運用システムへの導入難度、クライアント単位での情報取得の可否をまとめた。本研究では、パッシブ手法を用いてより高精度で運用管理に有用な情報収集を目指す。

3.8 視聴品質評価手法

画質の評価手法として一番直接的な方法が、観測者による直接観測である。画質評価の主観評価については、ITU (International Telecommunication Union) によって評価方法が厳密に定義されている。評価には、専門家でない観測者が合

計 15 名以上必要とされ，原画像と対象画像の比較を定められた手順，評価方法にのっとり行う [29]．代表的なものに，DSIS（二重刺激妨害尺度法），DSCQS（二重刺激連続品質尺度法），SSCQE（単一刺激連続品質評価法），SDSCE（同時二重刺激連続品質評価法）がある．

主観評価は，複数の人間が視聴した品質を直接評価するので，エンドユーザ視聴感と直結している．また，定められた基準にのっとりることにより，個人差や評価方法のばらつきも比較的強く抑えることに成功している．しかし，主観評価を行うためには人手と時間が必要であるため，長時間の動画像評価や長期観測は非常に困難になる．

観測者によらず画質を評価する手法として，画像の客観評価がある．最も基本的な客観手法として，PSNR（Peak Signal to Noise Ratio）があげられる．PSNR は原画像と対象画像を比較し，混入したノイズ量の比率を示すことによって画像の劣化指標とする測定方法であり，主観評価値との高い相関が認められている．この手法では原画像が必要になるため，あらかじめ原画像が不要な評価手法も開発されており，ITU-J.144 勧告 [30] をはじめ多数の技術が存在する．

一方で，これらの客観評価をネットワークの影響を受けるストリーミングに適用しても，ユーザ視聴感との高い相関を得られるとは限らない．なぜなら現在のインターネットストリーミングでは，レートの自動切替え，自動リバッファ機能を備えており，ネットワークの状態に合わせて動的に配信品質の変更を行うため，品質に影響を与える再生中断やレート変更が発生する．また，ネットワーク上でデータが失われることによって再生時にフレーム欠落や再生中断が発生する．これらインターネットストリーミング特有の現象は，映像に含まれるノイズよりはるかに大きな影響をユーザ視聴感に与えるため，リバッファやフレーム欠落を定量的に評価する必要がある．

3.9 研究目的と各手法の適応性

本研究では，運用中に個々のクライアントにおけるサービス品質を取得することを目指している．クライアントサービス品質を収集するには，クライアントにおける再生品質情報を直接サーバにフィードバックするのが最も正確である．しか

し、現在のストリーミング再生アプリケーションでは再生状態情報の記録、フィードバックは行わない。このため、再生状態情報を送信するモジュールを追加することによって情報の直接収集が可能である。しかし、不特定多数を対象とした配信で情報収集のためだけのモジュールを配布、インストールすることは実質的に不可能である。再生アプリケーションの変更という方法も考えられるが、対象が商用アプリケーションとなる上、OSに標準搭載されているアプリケーションを使用するユーザも多くこれも配信対象全ての情報収集手段としては適さない。視聴者による品質フィードバックもあり、これは視聴感そのものが値として得られる。しかし、これも全てのクライアントから収集することは困難である。

ネットワークの観測によるサービス品質情報は、サーバ側と配信ネットワーク上に設置するものがある。配信ネットワーク上に設置する方式はネットワーク管理上有効なものであるが、クライアントへの配信状態を直接示すものではない。また、主要な経路上に設置するため、ネットワークの末端に存在するクライアント付近で発生する異常とサービス品質への影響を知ることができない。サーバ側での観測では、パケットモニタによる方法がある。このとき、データパケットはUDPが用いられ、サーバへのフィードバックが返らないためクライアントへの配信状況を示す情報が得られない。これに対し、RTSPなどの制御に用いるセッションはTCPを用いており、クライアントアプリケーションにおけるパケット情報を含むため、受信状態を知るには有効であるが、パケット展開の時点で実装に依存する点を考えると、同じ情報が記録されるサーバログを利用する方が容易である。

サーバログはクライアントからの受信状態フィードバック情報や品質情報を含む。このうち品質情報は予備実験の結果や一般的知見から、クライアントにおける再生品質と異なった値を示す場合が多く見られるため、指標としては利用しにくい。パケットの受信状態情報は再生品質を知るために重要な情報ではあるが、再生品質を直接示すものではなく、品質劣化、正常配信の指標も不明である。

本研究では、直接的で正確であるが配信時に収集不可能な情報であるクライアント再生状態情報と、配信中に収集可能であるが情報が不十分なログ情報、配信中の計測が難しいネットワーク環境の情報を統合することで、有益な情報を取り

出せると考えた．配信中収集困難な情報と配信中に収集可能な情報との関係を明らかにすることにより，限られた情報から他の状態を高い精度で推測することが可能である．

4. ストリーミングサービス品質管理における課題

高品質なサービスを安定的に供給するためには、クライアントにおけるサービス品質およびネットワーク状況をサーバ側で正確に把握しておく必要がある。ストリーミングの場合、動画像の再生品質が、クライアントにおけるサービス品質と直結しており、動画像の再生品質評価値はエンドユーザの視聴品質に即している必要がある。そこで、ストリーミングに適した、再生品質評価について考察する。実運用環境下でクライアントでのサービス品質を収集する場合、運用に影響を与えない方式である必要がある。この点において、配信システムから容易に収集できるサーバログはサービス品質を知る上で重要な情報である。サーバログにはクライアントアプリケーションの通信状態が詳細に記録されており、運用中に収集が困難なネットワーク状態情報や再生品質を知るための手がかりとして有効である。

4.1 ストリーミングサービス品質把握における問題点

ストリーミングにおけるサービス品質管理で最も難しい点は、クライアントでの再生状態をネットワークパケットの状態から直接推測するのが困難なことである。推測が難しい理由は大きく2つ存在する。

1つは、アプリケーションデータバッファによるデータロス緩衝によるものである。クライアントアプリケーションは、データロスによる再生中断を避けるために、独自のデータバッファを保持している。このため、ネットワーク輻輳によるパケット喪失が発生しても、短期間であればクライアントアプリケーションのバッファ内に蓄積されたデータがあるため、再生は継続される。

もう1つは、コンテンツの違いに起因する再生品質変動によるものである。1つのデータパケットロスが再生品質に与える影響は、動画像圧縮方式ごとに異なる可能性がある。また、コンテンツの動きが小さければ、データパケットロスによりフレームが欠落しても視聴者に認識されない可能性もある。このため、パケットレベルでのデータパケットロスと再生品質の劣化程度は必ずしも直結しない。

加えて、現在のストリーミング配信では実装が非公開であるシステムが主流と

なっており，内部的な構造からシステムの挙動を分析することは困難となっている．このため，ブラックボックスであるシステムにおいては，外部から得られる限定的な情報から，システム挙動に関する運用上有益な知見を導出する必要がある．

4.2 エンドユーザ視聴品質に即したサービス品質評価

RTSP の登場によって，これまでの UDP による一方通行のストリーミングや，HTTP による疑似ストリーミングとは異なり，再送を含むきめ細かな配信制御と，クライアント情報のフィードバックが実現した．RTSP を通じて，クライアントアプリケーションにおけるデータパケットの到着状態は逐次サーバに通知される．データパケット自体は UDP の上で転送されるため，トランスポート層レベルでの到着応答は送られないが，RTSP のストリーム制御セッションにより，UDP のリアルタイム性に信頼性を付加することに成功している．ただし，ログに記録されるデータパケットの到着状態が，クライアントの再生品質をどのように示しているかは不明であるため，ログ情報だけではクライアント再生品質を正確に知ることはできない．そこで，クライアント再生品質をまず正しく評価し，ログ情報との関連を明らかにする必要がある．

クライアントでの配信品質は，ストリーミングアプリケーションでは再生品質が直結する．ネットワーク環境が悪化した場合，クライアント再生アプリケーションでは再生，早送り，巻き戻しなどのアクションに対するレスポンスが著しく低下，再生映像がコマ落ちによりスムーズに再生されない，データ不着によりバッファ内データが枯渇しリバッファにより再生停止するなどの現象が見られ，視聴品質は大きく低下する．

画質劣化によるユーザ視聴感の変化については，多くの既存研究が存在する．しかし，これらはローカル環境下での動画像圧縮によるブロックノイズ等を対象としている．対して，ストリーミングで画質劣化が起こる要因は，エンコード時の圧縮によるもののほか，途中ネットワークでのパケット遅延や喪失によるものがある．パケット不着が発生した場合，再生がスムーズに行えないなど，エンコード時圧縮よりも視聴感への影響は大きいとされる．

サーバサイドでのクライアント配信品質把握で問題となるのは，これら視聴品

質を低下させる現象を評価する方法が存在しないことである。特に，再生時のコマ落ちによる品質劣化は主観評価によるものが中心となっている。主観評価には多くの人的リソースと時間が必要であり，現実的には実現困難である。また，配信の都度主観評価を行うことは不可能である。動画像の品質客観評価については多くの既存研究が存在するが，そのほとんどは動画像圧縮による品質劣化を対象にしており，ブロックノイズの検出や画質劣化に基づくものである。一方，ストリーミングのようにネットワークを介した場合の packets 不着による品質劣化ではコマ落ちが主な品質劣化の原因であり，動画像圧縮による劣化とは性質が異なるため，従来研究を適用することはできない。

クライアント再生アプリケーションから再生状態を抽出することにより，評価基準となる値を得ることが可能である。ただし，ここで得られる値は非常に微細な変動も含むため，視聴者には視認できない程度の状態変動でも劣化とみなす可能性が高い。エンドユーザ視聴感に評価値を近づけるために，まず視認できる閾値を実験により定め，視覚的に影響が大きい再生状態変動に対しては大きく評価値を下げる評価方法が必要である。

本研究では，アプリケーションの観測から得られる情報から，エンドユーザ視聴感に基づいた動画像品質客観評価手法を確立する。これにより，実験環境，実環境を問わず，統一的な基準に基づいて，視聴感に影響する品質劣化を検出，劣化程度を評価できる。動画像配信劣化を定量化することにより，各クライアントでのサービス品質を評価することができる。

4.3 サーバ側でのクライアント配信品質把握

大規模ストリーミング配信システムで得られる運用管理情報は，サーバプロセスの監視や負荷などサーバから直接得られる情報，ネットワークのタッピングや応答時間などネットワーク環境に関する情報，そして RTSP やクライアントアプリケーション内モジュールによるフィードバック情報などクライアントでのアプリケーション状態情報の 3 つに分けることができる。

CDN におけるエッジサーバ監視システムでは，サーバ情報とネットワーク情報の収集，分析に焦点がおかれており，ネットワーク的，地理的に分散したエッ

ジサーバ群から各拠点の状態情報をリアルタイムに収集し、分析した情報に基づいて負荷分散を行う。これは大規模な配信システムを安定して効率よく運用するために非常に重要ではあるが、個々の詳細な配信状態や品質についてはターゲットとしていない。

ネットワーク観測方法は様々な手法があり、サーバサイドでのネットワークタッピングによるパケット情報収集、途中ルータでのパケットそのものの収集あるいはパケットに含まれる情報の抽出、観測パケット送出による応答時間や喪失率の計測などがある。このうち、途中ルータへの変更、観測パケット観測は、運用システムの安定性や、負荷に大きな影響を与える可能性があり、大規模配信では現実的に困難である場合が多い。サーバサイドでのネットワークタッピングは、システムへの影響がないという点で導入がしやすい。データ配信に TCP を用いている場合、プロトコルによっては終了状態などフィードバックされる情報から配信状態を推測できる場合もある。しかし、ストリーミング配信では通信の大半を占めるデータ配信に UDP を用いており、クライアントサイドからのフィードバックを得ることができないため、サーバサイドでの観測は有効性が低い。

以上から、大規模ストリーミング配信において、クライアントの状態情報を収集するには、クライアントからのフィードバック情報が重要となる。クライアントからフィードバックを得るには、クライアント側で特殊なモジュールを用いることによりサーバ側へフィードバックさせる方法と、RTSP などにより収集された情報の分析がある。モジュールを用いる方法は、配布、インストールコストの高さとアプリケーションの安定性に影響を与える可能性があり、不特定多数を対象とする大規模配信では適応が困難である。RTSP はアプリケーションにおける状態情報や受信状況をフィードバックするため、多くの情報をシステムの変更なしに収集することが可能である。しかし、RTSP を通じたフィードバック情報に含まれるパケット受信情報やイベント情報は、クライアントで発生するこれらの現象を知る手がかりにはなるが、実際にこれらの現象が起きたときどのような値の変化を示すかは明らかではない。

ログ情報の解析に関する研究はこれまで数多く存在するが、それらはアクセス状況の分析であったり、キャパシティプランニングに有用なアクセス数予測に関

表 2 クライアント品質評価手法

評価手法	利点	欠点
主観評価	エンドユーザ視聴感を反映	人的リソースが必要
客観評価	エンドユーザ視聴感を反映	ストリーミングに適した評価が必要
パケットモニタリング	正確なパケット情報	ストリーミングで低い有効性
サーバログ分析	手軽に利用可能	エンドユーザ視聴感と直結していない

する研究，あるいは VoD とライブ配信のアクセスパターンの違いを解析したもののなど，本研究の目指す，クライアントサイドでの配信品質分析につながるものではない．

4.4 ストリーミングサービスに適した品質評価手法

各評価手法の利点と欠点について表 2 にまとめた．本研究では，実運用でのストリーミング品質管理に有効性が高い評価項目として，再生品質の客観品質評価とサーバログを取り上げる．アプリケーションでの再生品質とサーバログ項目の相関を明らかにすることにより，サーバログからより詳細なサービス状態品質を引き出すことを目標とする．

4.5 サーバログ分析によるサービス品質管理

現在の商用ストリーミングアプリケーションでは，RTSP によりクライアントから収集された通信記録に基づき，アプリケーション層でのパケットの受信状態などを，ログに非常に詳細な情報が配信先毎に保存することができる．ストリーミング配信におけるログ解析の難しさは，これらの数値から配信状況をどう読みとるかにある．WWW サーバのログ解析とは異なり，ストリーミングの場合はこれらの情報が品質にどう影響しているかを直感的に判断しにくい．ストリーミングでは，クライアント側でもバッファ，データデコードなどの負荷が比較的高い

処理を行っており，単純にパケットが転送成功したから品質が維持されるとは限らない．また，サーバログには品質分析項目があり，パーセンテージで配信品質を表すものもある．これは非常に直感的に管理者に配信品質を知らせる項目ではあるが，実験の過程で，実際にはクライアント側で品質劣化が起きているにもかかわらず，配信品質良好な 100% という値を示す場合もあり，必ずしも正確に配信品質を表しているとはいえない．

サーバログには，クライアントアプリケーションにおけるパケット受信状態が記録される．具体的には，1 回の送信で受信したパケット数，再送が成功したパケット数，再生タイミングまでに到着しなかったパケット数をサーバに通知する．クライアントにおけるサービス品質すなわち再生品質は，パケットの到着状態によって大きく左右される．パケットが不着の場合はフレーム落ち，再生中断あるいはリバッファによる再生停止が発生し，視聴品質を低下させる．このため，ログに記録されるパケット到着状態は，サービス品質を知るための重要な手がかりになる．

パケットの到着状態はまた，サーバクライアント間ネットワークの状態によって変動する．例えば，パケットロス率の上昇はパケット不着の増加につながり，遅延や遅延ジッタが増加している場合は再送パケット数が増加することが予想される．特定のネットワーク環境においては，パケットの到着状態が一定のパターンを示すと考えられ，ネットワーク環境とパケット到着状態パターンを明らかにすることができれば，サーバログからネットワーク環境を推測することができる．サーバログは個々のクライアントに対するパケット到着状態を管理しているため，配信クライアント全てに対して，サーバクライアント間のネットワーク状態を把握できることになる．

このように，サーバログはシステムに変更を加えたり負荷を与えることなく利用できる上，クライアントアプリケーションの通信状態を詳細に記録しており，運用環境下でのサービス品質把握には非常に適した情報源である．しかし，通信状態は実際のクライアントサービス品質を直接示すものではなく，どのようなログの状態がサービス品質の低下を示すのかは明らかではない．また，パケット転送状態パターンとネットワーク環境との関連も不明である．サーバログを用いて，

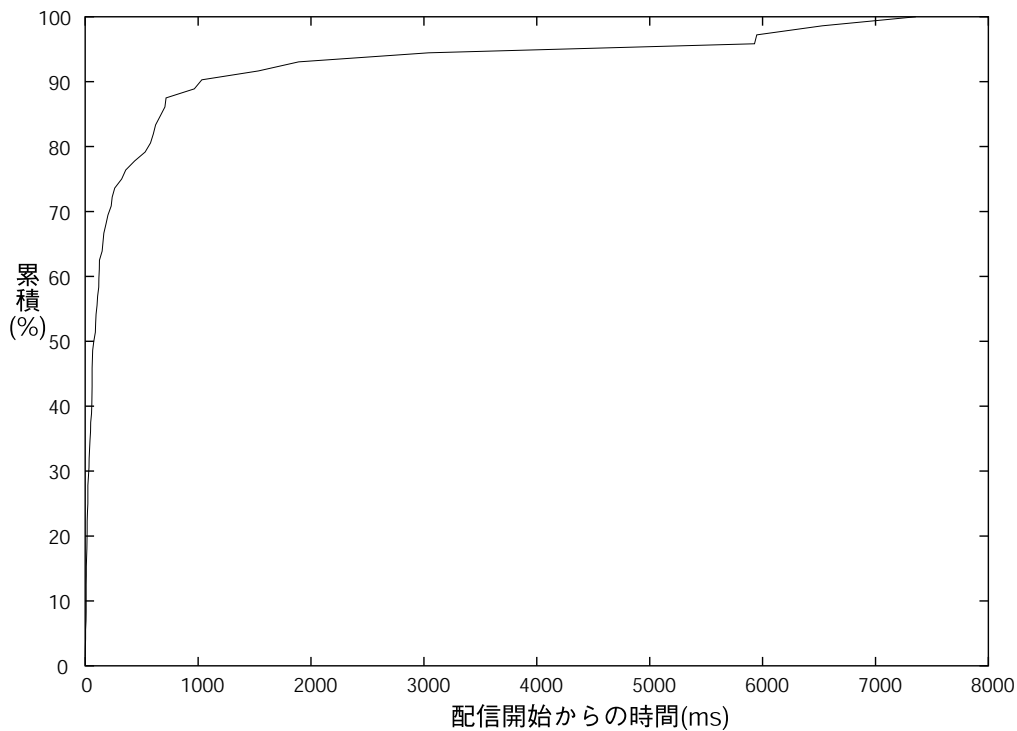


図 2 セッション継続時間の累積頻度分布

サーバ側でサービス品質管理を行うためには、これらを明らかにする必要がある。

4.6 運用時のサーバログ有効性

サーバログからは、RTSP フィードバックによるクライアントアプリケーションの受信状態情報がクライアント単位で記録されるため、サービス品質把握に有効である。しかし、これらの記録はセッション継続中はサーバアプリケーションが内部的に保持しており、切断時にはじめて記録されるため、配信中に外部から得ることはできない。本研究に先立ち、2003年5月のLIVE!UNIVERSEによる金環日食中継で実験を行った際のログから、セッションの継続時間を累積度数で示したものが図2である。ここでのセッション切断はクライアントでのユーザアクションによるものであるが、75%のクライアントは6分以内にセッションを切断している。これより、ライブ配信中であっても短時間でセッションが切断され、

ログに記録されていることが分かる。

また、本研究ではクライアントに対するサービス品質把握を目指す。正常配信に関しては配信中にリアルタイムに知る必要性は低いと考える。サービスの保証と記録という面で情報としては必要であるが、リアルタイムに得る必要なのは配信異常が重要となる。配信異常が発生した場合は、クライアントアプリケーションまたはサーバアプリケーションによりセッションが切断、あるいはユーザアクションによる切断や再接続が発生し、早い段階でログにあがってくると考えられる。このため、サーバ側で障害発生時などの初期に発見、対処が可能であり、異常検出という目的に対しては有用であると考えられる。

5. ストリーミングサービス品質評価手法

インターネットストリーミングにおいて、クライアントサービス品質をサーバサイドで把握することはサービス提供者やシステム管理者に必須となってきた。しかし、クライアント再生品質をサーバログから定量的に推定するための指標はこれまであまり知られていない。

クライアントサービス品質評価方法として、クライアントでの再生フレームレートに着目して分類を行い、実験結果から、クライアントサービス品質、サーバログとネットワーク状態の相関を明らかにした。サーバログに出力される情報の中で、「クライアント回復パケット数」がクライアントサービス品質劣化の重要な指標となることが分かった。

5.1 サーバログからのサービス品質、ネットワーク状態推定

提案手法では、図3に示すように、サービス品質判断の基準となるクライアント、サービス品質の把握が必要なサーバ、そしてサービス品質に影響を与えるネットワーク環境のそれぞれの関係に注目した。まずクライアントサービス品質指標として、エンドユーザの視聴感を反映させた定量的な画質評価方法を提案する。その上で、様々なネットワーク状態を人工的に再現した環境下において、クライアントサービス品質、サーバログ出力、ネットワーク状態との相関を明らかにする。

なお、ストリーミング再生品質評価では動画像と音声の両方について評価項目が考えられるが、実験では動画像の再生品質評価のみを対象としており、音声に関する再生品質評価は行っていない。

実験、分析により、エンドユーザ視聴感に即したより正確なクライアントサービス品質をサーバログから取り出すことを目指す。

5.2 サーバログ

ストリーミングサービスでは、コンテンツデータ配信のためのセッションと、クライアント通信制御情報交換のためのセッションを分離している。クライアン

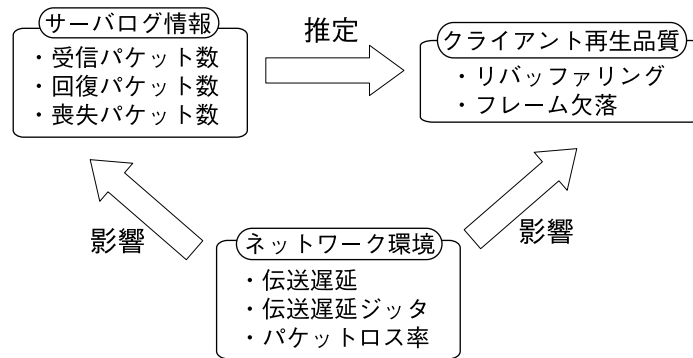


図 3 サーバログ情報からのサービス受信状態の推定

トアプリケーションにおけるパケット受信状態や再送リクエストは、後者のコントロールセッションを通じてサーバに送られる。このクライアントから送られる状態情報は、サーバログに記録されるため、このログを用いることでクライアントアプリケーションの挙動を比較的詳細に知ることができる。

Windows Media Server では、1 配信毎にクライアントアプリケーションから得た情報を記録している。なかでも、クライアントにおけるサービス品質と関係が深いのは、クライアントアプリケーションの受信パケット数、回復パケット数、喪失パケット数、リバッファ時間である [21]。受信パケット数はクライアントが再送なしで受信できたパケット数を、回復パケット数はデータ喪失により再送要求を送り、再生タイミングまでに受信することができたパケット数を、喪失パケット数は再生タイミングまでに受信することができなかったパケット数を表している。リバッファ時間は、1 回の配信で発生したりバッファの合計時間を表している。

5.3 クライアントサービス状態

ストリーミングクライアントにおけるサービス品質は、クライアントアプリケーションにおける再生品質である。動画像の再生品質に影響を与える要素として、再生途中に起こるリバッファによる再生停止、動画像がなめらかに再生されないフレーム落ち、再生画質の劣化があげられる。再生画質の劣化は、1 つのフレームに含まれるスライス、あるいはより小さな単位であるマクロブロックやブ

ロックの欠損によるものである。これらの原因による再生画質劣化は、フレーム落ちと比較した場合、視聴品質への影響はより小さいとされている [31]。

ストリーミングサービスでは、指定された時間にフレームを再生するため、数秒先の再生データまであらかじめアプリケーションバッファに保持しておく仕組みになっている。このバッファに、連続的な再生をするのに十分なデータがないと判断される場合は、一定量のデータを蓄えるまでは再生が開始されず、この動作をバッファリングと呼んでいる。バッファリングは配信開始時のほか、ネットワーク上でデータロスなどが発生し、配信中にバッファ内のデータが枯渇した場合にも起こる。バッファリングの期間は再生が中断するため、視聴感への影響は非常に大きいといえる。

動画のフレーム落ちも再生品質に与える影響が大きい。配信動画像は指定されたフレームレート値でエンコードされ、クライアントでデコードし再生される。クライアントアプリケーションでは、デコード時のフレームレートを取得することができる。これを再生フレームレートとし、エンコード時の指定フレームレートをエンコードフレームレートとする。通常はエンコードフレームレートと再生フレームレートは一致するが、何らかの原因でフレームの一部が再構成できなかった場合、再生フレームレートの値が減少する。

Windows Media Player では、再生フレームレート値は配信中にのみ表示させることが可能であるが、記録することができないため、今回 Windows Media のライブラリキットである SDK を用いて 1 秒ごとのステータス変化を取得した。

5.4 クライアント再生状態の分類

配信中のフレームレート変動は、図 4 に示すような挙動を示す。予備実験より、バッファ直後に再生フレームレートの急激な上昇が発生し、その後減衰しながらエンコードフレームレートに収束する現象が確認されている（図 4, I の区間）。

配信中のクライアントアプリケーションは、停止、再生、バッファリングの 3 状態に分けることができ、再生状態は品質に基づきさらにいくつかに分類することができる。それぞれの状態の遷移を示したのが図 5 である。再生品質の指標と

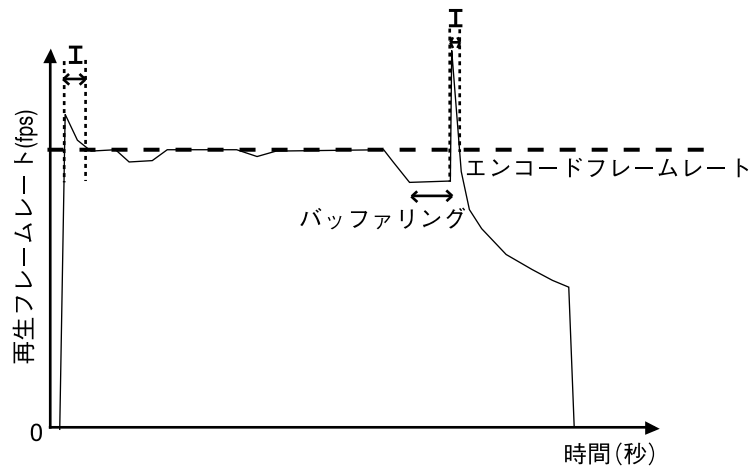


図 4 配信中のフレームレート変動の例

なる再生時のフレーム落ちは，再生フレームレートの変動を観測することで検出が可能である⁶。

停止状態

アプリケーションが起動され，再生を開始していない状態は停止状態である．他のステータスにあるときに，アプリケーションで再生停止命令を検出した場合や配信が終了した場合には停止状態に遷移する．

バッファリング

アプリケーションで再生イベントを検出すると，サーバとのコネクション確立後，データ転送リクエストが送られ，間もなくバッファリングに入る．

⁶クライアントでの再生フレームレートは1秒間隔で小数点以下2桁まで記録されているが，1秒間に1フレーム未満という微細な変動であれば，品質への影響は目視観測で確認できなかった．このため，本論文の分析では秒間1フレーム以上の再生フレーム変動のみを再生状態変動検出に用いた．

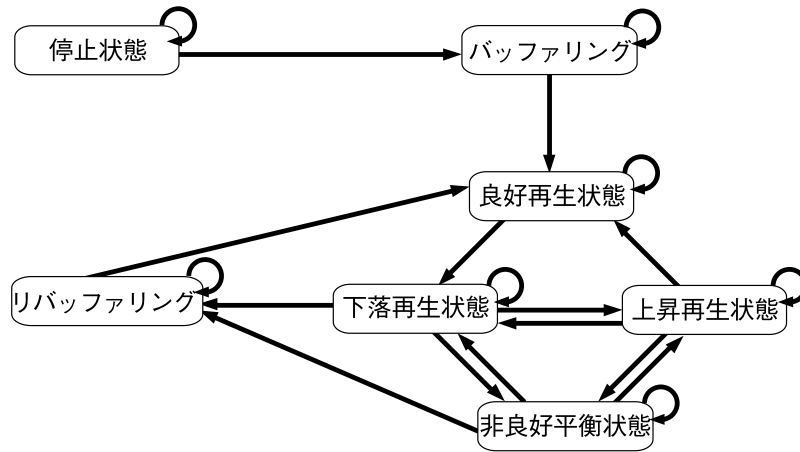


図 5 クライアント再生状態遷移

リバッファリング

配信中にデータの転送が滞りバッファが枯渇した場合には、リバッファリングを行い、データ再蓄積を行う。

良好再生状態

再生フレームレートがエンコードフレームレート以上の値を記録しているときは、フレームレートの下落が発生していても再生に影響が見られなかったことから、これを良好再生状態と定義する。データが損失し、再生タイミングまでにフレームとして再構成できなかった場合にはフレーム欠落が発生、下落再生状態に遷移する。

下落再生状態

再生フレームレートがエンコードフレームレート未満の値で、かつ再生フレームレートが下落を続けているとき、視認可能なフレーム落ちが多発する。フレームレートが下落したまま変動しなければ非良好平衡再生状態に、上昇すれば上昇再生状態に遷移する。

上昇再生状態

再生フレームレートがエンコードフレームレート未満の値で、かつ再生フレームレートが上昇を続けているとき、視認可能なフレーム落ちは観測されない。予備実験では、再生フレームレート上昇時の品質劣化は目視で確認できなかったが、再生フレームレート上昇時の視聴感への厳密な影響は明らかではないため、商業利用においては影響が出る可能性がある。

非良好平衡再生状態

再生フレームレートがエンコードフレームレート未満の値で、かつ再生フレームレートが変化していないとき、視認可能なフレーム落ちがところどころ観測された。

5.5 各状態滞留時間による再生品質評価

配信中に、各状態に滞留している時間の長さはクライアント品質の基準となる。サーバからのデータ受信が問題なく行われ、安定した再生ができている場合は、エンコードフレームレートどおりに再生が行われるため、良好再生状態が長いほどクライアントにおける品質は高いといえる。下落再生状態が発生している場合は、再生時にフレーム落ちが発生していることを示しており、この時間の長さは、再生品質劣化を細粒度で観測する際に重要な指標になると考えられる。リバッファ状態では再生が停止するため、エンドユーザ視聴感に大きなマイナスになる。リバッファ状態に遷移する時点で既にクライアントでの再生状態は相当悪化しており、品質劣化を早期に検出するためのパラメータとしては不適である。

上昇再生状態と非良好平衡再生状態については、他のパラメータほど直接的にクライアント品質を示さないと考えられる。上昇再生状態が存在するということは、前段階でデータフレームレート以下への下落が発生していることを示しているため、一概に高品質とはいえない。非良好平衡再生状態は、再生フレームレートが連続的に下落しているときほどのコマ落ちは観測されないが、コマ落ちが発

表 3 再生状態判断アルゴリズム

良好再生状態	$F_n \geq E$
下落再生状態	$F_n < E$ かつ $F_n < F_{n-1}$
上昇再生状態	$F_n < E$ かつ $F_n > F_{n-1}$
非良好平衡再生状態	$F_n < E$ かつ $F_n = F_{n-1}$

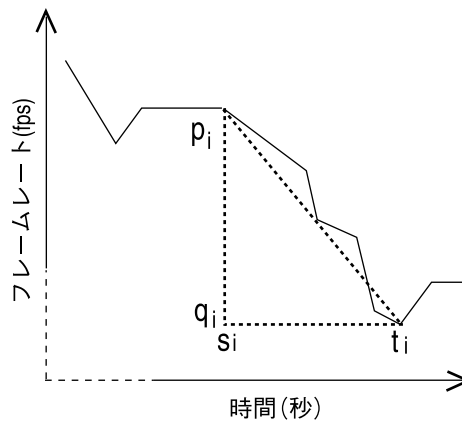


図 6 フレームレート下落の連続性による評価

生していないことを表しているのでもないため、品質評価指標としての優先度は下がる。

各再生状態の判断アルゴリズムを表 3 に示す。なお、エンコードフレームレートを E (fps)，配信開始から n 秒後のフレームレートを F_n とする。

5.4 節で示した各配信状態について、配信中の合計時間をそれぞれ「良好再生状態時間」「下落再生状態時間」「上昇再生状態時間」「非良好平衡再生状態時間」と定義する。

5.6 再生フレームレート下落の連続性による再生品質評価

下落再生状態が続いた場合、フレーム落ちが連続して起こるために再生が中断したように観測され、視聴感を大きく下げる。下落再生状態が持続している場合

と、ごく短時間に突発的に発生している下落再生状態を区別するために、再生フレームレート下落の連続性を考慮した評価方法を考える。

クライアントでの再生フレームレートを毎秒記録し、5.4節で定義した各状態が連続している箇所を検出する。この結果、1回の配信は、いくつかの状態連続区間の集合に分けられる。これらのうち、下落再生状態の連続区間(図6)において、 i 番目のものの開始地点の時刻を s_i 、フレームレートを p_i 、連続状態が終了する地点の時刻を t_i 、フレームレートを q_i とする。

下落再生状態が継続した場合は連続的なフレーム落ちが発生するため、視聴者には再生が停止したように観測される。再生停止はリバッファと同様に再生品質を大きく低下させるため、短時間のバースト的なフレーム落ちとは区別する必要がある。そこで、下落再生状態が継続した時間が長いほど品質評価値が低くなるよう、下落再生時間 $t_i - s_i$ に重みをつける。1 配信中で検出した下落再生状態の区間数を n とし、下落再生時間の重み付けを m としてクライアント再生品質評価式を次のように定義する。

$$V_{down} = \sum_{i=1}^n (q_i - p_i) \cdot (t_i - s_i)^m$$

なお、エンコードフレームレートが高い場合は同じフレームレート下落幅であっても再生品質への影響は小さくなる。逆にエンコードフレームレートが低い場合はフレームレート下落の影響は大きくなる。現在は多くの配信で 30fps によるエンコードが標準的に用いられているため、エンコードフレームレートに対する下落割合は評価式では考慮していないが、異なるエンコードフレームレートで評価する際には下落割合について検討する必要がある。

5.7 実験手順

ストリーミングの配信形態には、サーバにあらかじめエンコードしたファイルを配置しておき、クライアントからのリクエストに応じて提供するオンデマンド方式と、エンコーダで動画像をリアルタイムにエンコードしながら順次サーバに送出し、サーバから配信を行うライブ方式がある。オンデマンド配信では、まとめてクライアントにデータを送信することが可能なため、QoS 管理要求は一般に

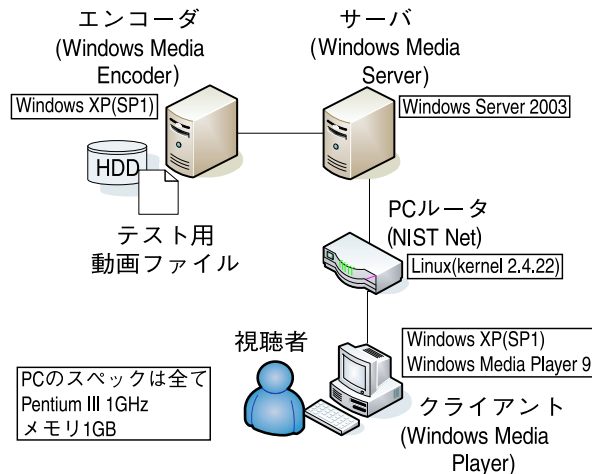


図 7 実験ネットワーク環境

ライブ配信ほど困難ではない．本論文では，より高い精度で QoS 管理が求められるリアルタイム配信を適用対象とし，実験はライブ方式で行った．ライブ方式では，テレビカメラなどの映像をデータソースとすることが一般的であるが，実験では再現性の観点から動画ファイルデータをデータソースとし，エンコーダハードディスク上に配置した．

図 7 に実験環境の概要を示す．実験機には Pentium III 1.2GHz，メモリ 1GB のサーバを用いた．各機器は 100Mbps のイーサネットによって接続されている．インターネット上で発生する様々なネットワーク環境を再現するため，サーバとクライアントの間に PC ルータを設置し，NIST Net [32] によるネットワークエミュレートを行った．実験ではネットワーク伝送遅延，ネットワーク伝送遅延ジッタ（ゆらぎ），パケットロス率について表 4 に示す値を設定した．

本研究で提案する方式は，サーバ・クライアント形態で配信するストリーミング配信システムには基本的に適用可能である．本実験では，サービスソフトウェアに Windows Media シリーズ [11] を用いた．Windows Media シリーズは現在最も普及しているストリーミング配信コンポーネントである．

クライアント再生品質取得には，Windows Media SDK を用いて 1 秒ごとの再生状態とイベントをログとして記録した．再生状態として記録した項目は，再生

表 4 ネットワーク設定パラメータの値

遅延 (ms)	0, 50, 100, 200, 300
遅延ジッタ (%)	0, 50, 100
パケットロス率 (%)	0, 3, 5, 7, 10

表 5 エンコード設定

動画 Codec	Windows Media Video 9
音声 Codec	Windows Media Audio 9
エンコードビットレート	340kbps, 148kbps, 58kbps
エンコードフレームレート	30fps

開始からの経過時間と毎秒の再生フレーム状態である。イベントとして記録した項目は、再生開始、再生停止、バッファリング開始、バッファリング終了がある。このログに記録された毎秒の再生フレーム状態と、5.5節で示した再生状態判断アルゴリズムを用いて、各再生状態時間と下落の連続性による再生品質評価値を算出した。

5.8 コンテンツ

エンコーダで利用可能な動画像のエンコード方式は大きく VBR (Variable Bit Rate) と CBR (Constant Bit Rate) に分けることができる。VBR が動画像の内容によって時間あたりのデータ量が増減するのに対して、CBR は動画像の動き量などに関係なく一定のレートでデータをエンコードする方式である。ネットワーク消費帯域がコンテンツに依存しにくいいため、ストリーミングのエンコード方式では、一般に CBR が用いられている。実験では、CBR を用いてエンコードを行った。エンコード設定は表 5 のとおりである。

映像の長さは 3 分間で、講義の様子を撮影したものを使用した。実験機では動きの大きい動画像のエンコードが困難であったため、動きの比較的小さい映像を用いている。動きの大きい動画像を用いた場合は、フレーム落ちがより視聴者に視認されやすい可能性がある。本論文では再生フレームレートからフレーム落ち

を検出するため，目視による観測を併用して閾値の設定を行った．その結果，比較的動きの大きい部分においても1秒間1フレーム未満の再生フレームレート下落ではフレーム落ちとして視認できなかった．このため，1秒間1フレーム以上の下落を目視可能なフレーム落ちとして統一的に取り扱った．

5.9 実験結果

本節では，まず提案した品質評価関数がエンドユーザ視聴感を表しているかを確認するために，目視によるフレーム落ち観測結果と提案方式による再生品質評価を比較する．次に，ネットワーク環境がサービス品質にどのような影響を与えているかを明らかにするために，クライアント側で現れる影響とサーバ側で現れる影響の両方について検討する．具体的には，ネットワーク環境とクライアント再生品質評価値との相関，ネットワーク環境とサーバログ項目との相関を示す．これらをふまえ，サーバログパラメータとクライアント再生品質評価値との相関を示し，サーバサイドでのクライアント品質把握のために，有効なログ項目に関して議論する．

なお，以降の節で示す図中で，プロットが打たれていない部分は，測定不能であったためのデータ欠落による．

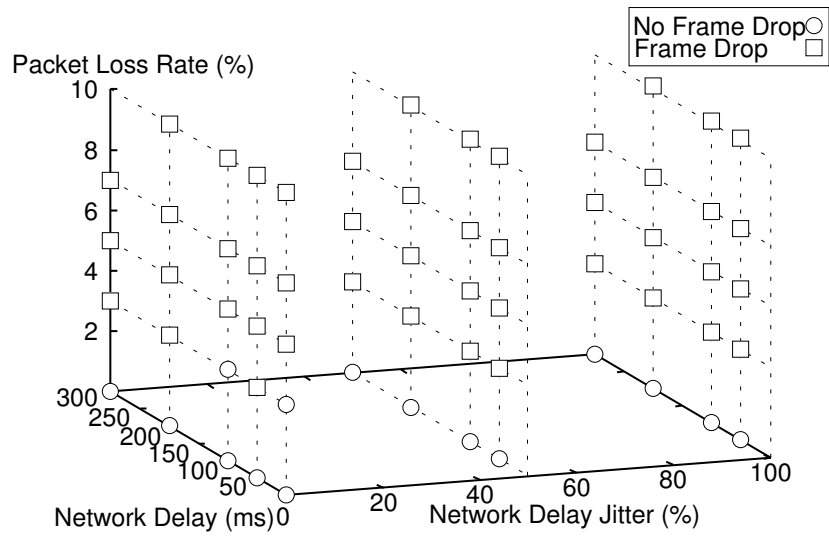


図 8 目視によるコマ落ちの有無とネットワーク状態の相関

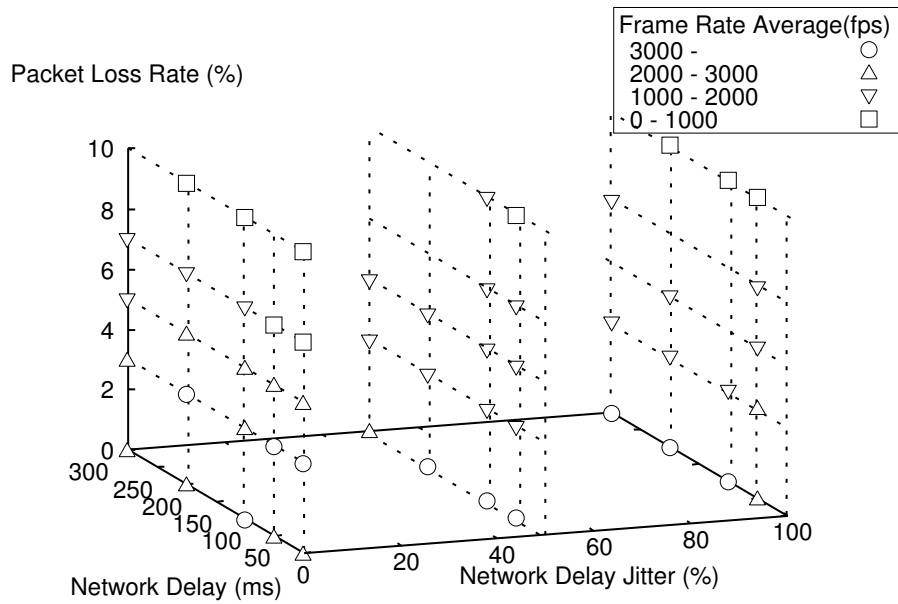


図 9 再生フレームレート平均値とネットワーク状態の相関

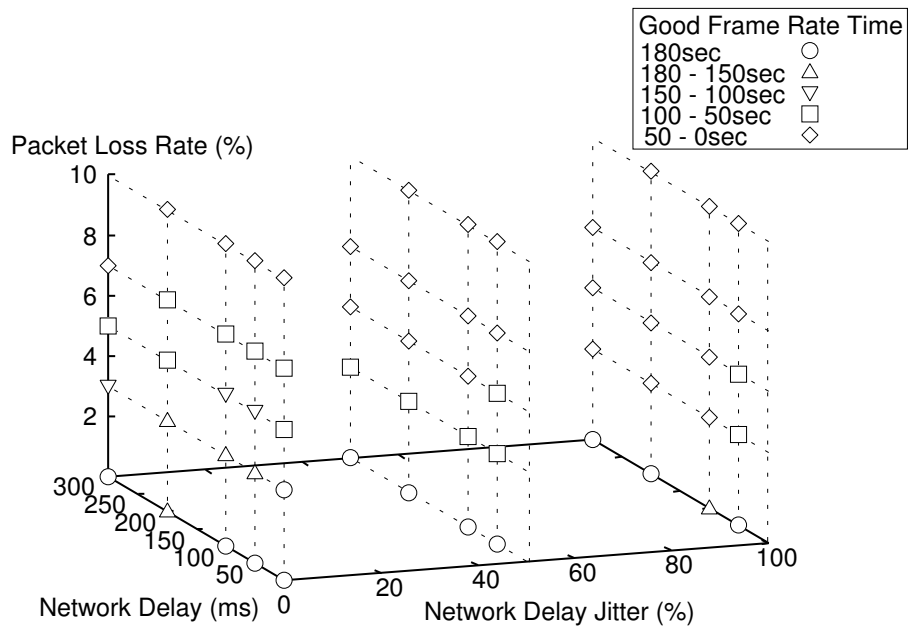


図 10 良好再生状態時間による品質評価

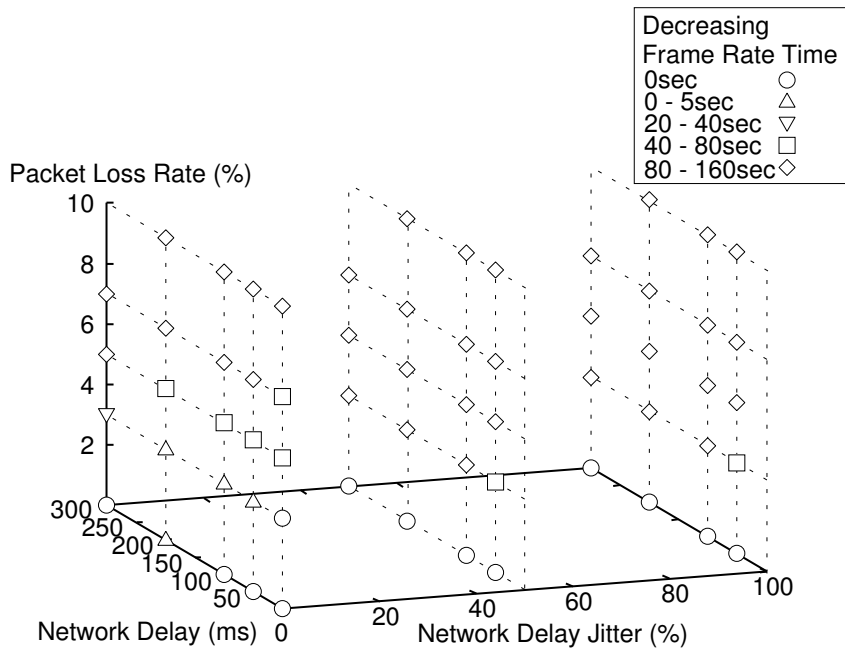


図 11 下落再生状態時間による品質評価

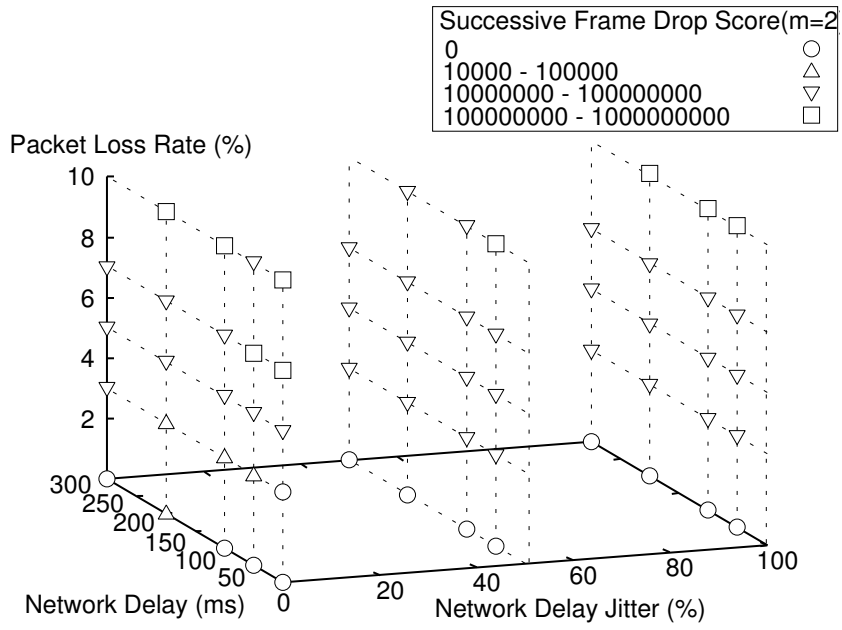


図 12 下落連続性 V_{down} による品質評価 ($m = 2$)

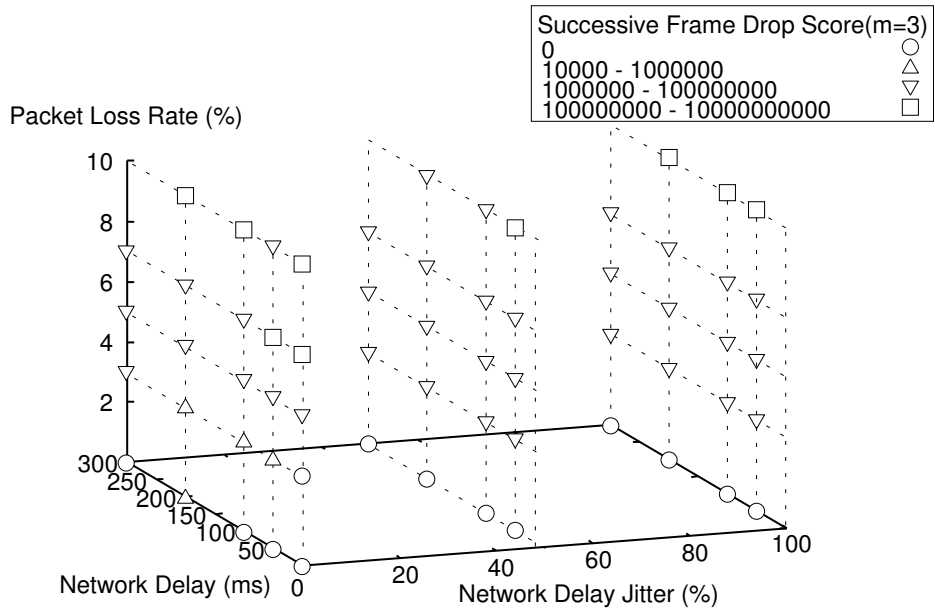


図 13 下落連続性 V_{down} による品質評価 ($m = 3$)

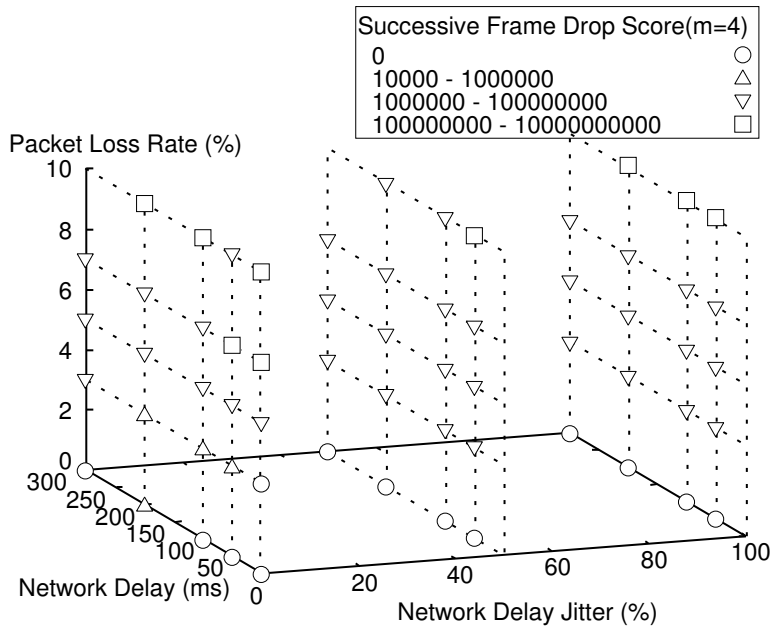


図 14 下落連続性 V_{down} による品質評価 ($m = 4$)

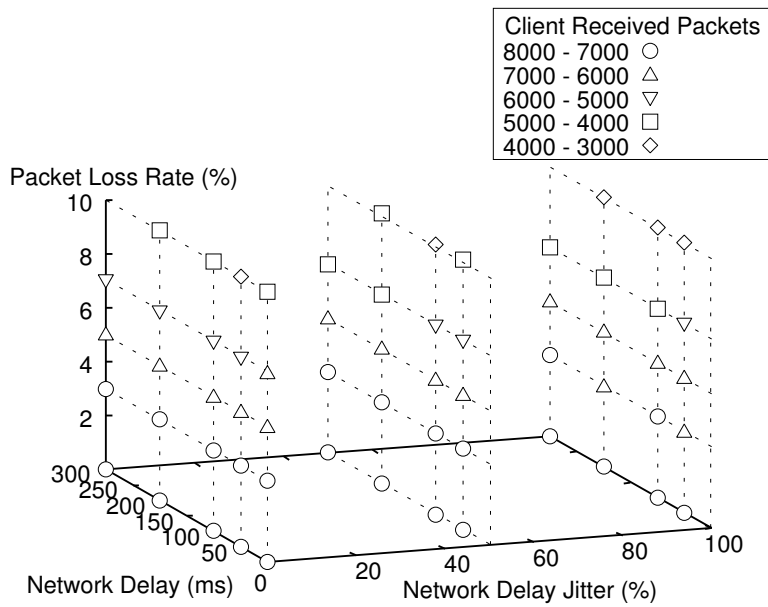


図 15 クライアント受信パケット数

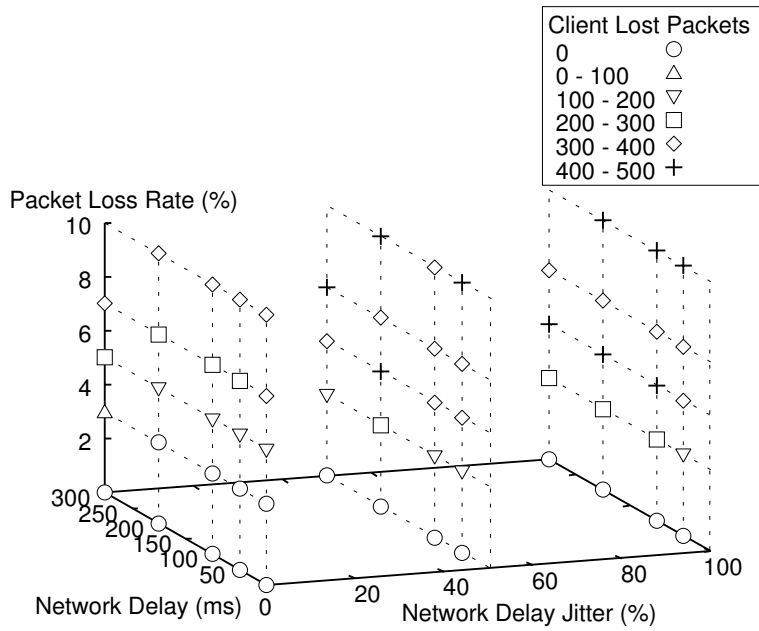


図 16 クライアント喪失パケット数

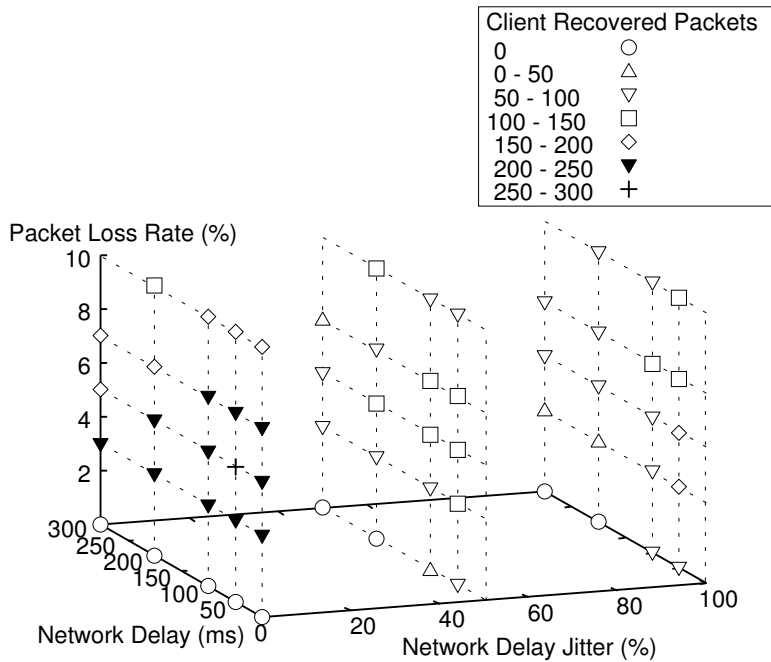


図 17 クライアント回復パケット数

5.9.1 品質評価方法と目視によるコマ落ち観測結果の相関

図8は、ネットワーク環境と目視によるフレーム落ちの有無を示したものである。結果に若干のばらつきは見られるが、パケットロスに遅延ジッタ、遅延をともなう場合はほとんどの場合でフレーム落ちが観測された。

図9は、再生フレームレートの平均値による評価を示したものである。目視でフレーム落ちが観測されていない場合に低い品質を示すものや、逆に目視でフレーム落ちが観測されている場合でも高い品質を表すなど実際の視聴品質との不一致が各所に見られる。これより、単純なフレームレートの平均値は品質評価値としては不適であることが分かる。

図10～図14は、ネットワーク環境とクライアントでの再生品質の評価値との相関を示したものである。

クライアント再生品質として、5.5節で定義した良好再生状態時間については図10で、下落再生状態時間については図11で示す。これらは、目視によるフレーム落ち結果と一致している。なお、5.5節で定義した再生状態時間のうち、上昇再生状態時間と非良好平衡再生状態時間についてはネットワーク状態との強い相関は見られなかった。

図12、図13、図14は5.6節で定義した下落連続性 V_{down} による品質評価値において、時間的に連続するフレームレート下落が品質指標値を下げるよう付けた重みの累乗部分 m が2, 3, 4の場合についてそれぞれプロットしたものである。これより、 m の値が2乗, 3乗, 4乗の場合でも得られる再生品質の傾向は変わらず、いずれも目視によるフレーム落ちを検出できていることから、ここでは数値的に小さく扱いやすい2乗を採用した。

3つの評価方法による品質評価結果と目視結果と比較すると、目視でコマ落ちを確認できた場合についてはいずれのケースでも劣化を検出しており、加えて目視で確認できなかったケースでも品質劣化を検出できていることから、目視結果よりやや細かい精度で品質劣化を検出できているといえる。

5.9.2 ネットワーク状態とクライアント再生品質の相関

図 10～図 14 より，パケットロスがクライアント再生品質に直接的に影響していることが分かる．パケットロスが発生している場合は，遅延時間が長くなると品質は劣化するが，遅延ジッタをともなう場合は特に大幅な品質劣化が起きている．この結果より，パケットロスが発生している環境下では，遅延そのものの大きさよりも遅延ジッタの大きさがクライアント再生品質に大きな影響を与えていることが分かる．

5.9.3 ネットワーク状態とサーバログの相関

ネットワーク環境によるサービス品質への影響がサーバログにどのように現れるかを知るために，ネットワーク状態とサーバログとの相関を示す．図 15～図 17 は，サーバログ項目とネットワーク状態の相関を示したものである．図 15 は，サーバログ中のクライアント受信パケット数について，図 16 は，クライアント喪失パケット数について，図 17 は，クライアント回復パケット数についてプロットしている．

図 15，図 16 より，パケットロス率が増加すると，サーバログでは受信パケット数の減少，喪失パケット数の増大が観測される．また，パケットロスが発生している環境下では，遅延ジッタの増加や遅延の大幅な増加が，受信パケット数減少，喪失パケット数増大を招く．回復パケット数は，受信，喪失パケット数と異なる変動を見せる．図 17 より，ネットワーク環境の悪化にともなって回復パケット数も増大するが，悪化が著しい場合には逆に回復パケット数は微減している．回復パケット数とは再送が成功した数を示しており，パケットの到達性が非常に悪くなると再送成功率が下がるため，回復パケット数も減少していると考えられる．

図 10～図 14 の結果と合わせると，再生品質の劣化度合いが大きいほど，受信パケット数は小さく喪失パケット数が大きいことから，サーバログにおける受信パケット数，喪失パケット数の割合はクライアント再生品質の程度を直接的に表していると予想できる．回復パケット数は，再生品質劣化が見られない場合（パケットロス率 0%，遅延ジッタ 50% 以上の場合など⁷）でも若干見られ，再生品質劣化

⁷パケットロス率 0%，遅延 200ms 以上では，短い遅延の場合と異なり，遅延ジッタの影響が

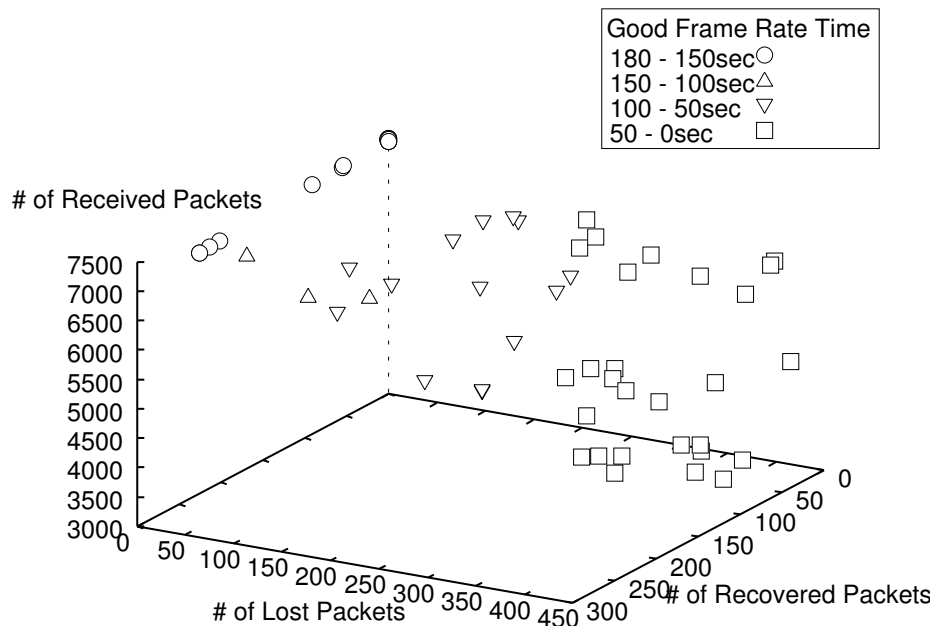


図 18 良好再生状態時間による品質評価

が起きている場合は必ず大きな値を示している．このことから，サーバログ中の回復パケット数は，ネットワーク環境，クライアント再生品質の低下に敏感に反応しているといえる．

5.9.4 サーバログとクライアント再生品質の相関

図 18～図 20 は，サーバログ項目とクライアント再生品質の相関を示したものである．図 18 では，良好再生状態時間によるクライアント品質評価を，図 19 では，下落再生状態時間によるクライアント品質評価を，図 20 では，下落連続性によるクライアント品質評価を行った．

品質の劣化が見られない場合は受信パケット数が 100%の地点を起点とし，回復パケット数の増加にともない再生品質の劣化が見られる．喪失パケットが発生ほとんど見られない現象が見られた．アプリケーションサイドの要因が考えられるが，詳細は不明である．ただし，片道遅延 200ms 以上の遅延は実配信ではそれほど頻繁には見られない．大規模配信では広域負荷分散技術によりネットワーク的に近いサーバを選択させることが通例であるからである．

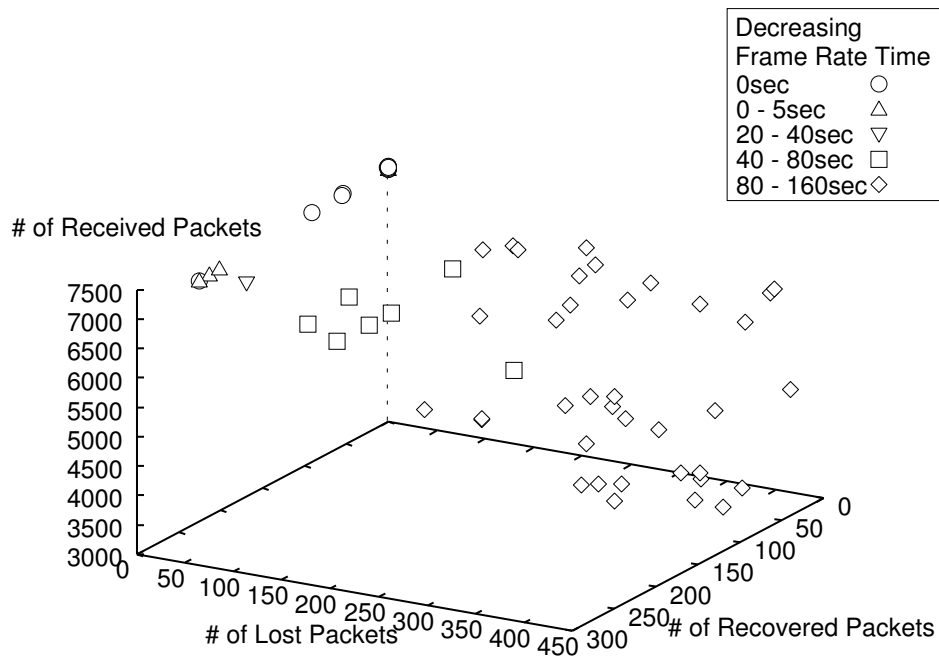


図 19 下落再生状態時間による品質評価

し始める時点では再生品質はかなり劣化している．さらに再生品質が劣化するほど受信パケット数と回復パケット数が減少，喪失パケット数が増加する．この傾向はいずれの再生品質評価でもほぼ同じである．

ここで注目したいのは，再生品質がやや劣化を始める地点では，喪失パケットは観測されず，受信パケット数もほとんど減少しないことである．このため，受信パケット数，喪失パケット数だけに着目した場合，品質の劣化が始まっている事実を見逃す可能性がある．一方，回復パケット数は，品質劣化が始まる少し前の状態から大幅な増加を見せるため，サーバログにおける回復パケット数は品質劣化の重要な指標になることが分かる．

5.9.5 実験結果のまとめ

クライアント品質評価，ネットワーク状態情報，サーバログの相関を見たとき，ネットワークでパケットロスが発生した場合は遅延ジッタ，遅延の順で品質に影響

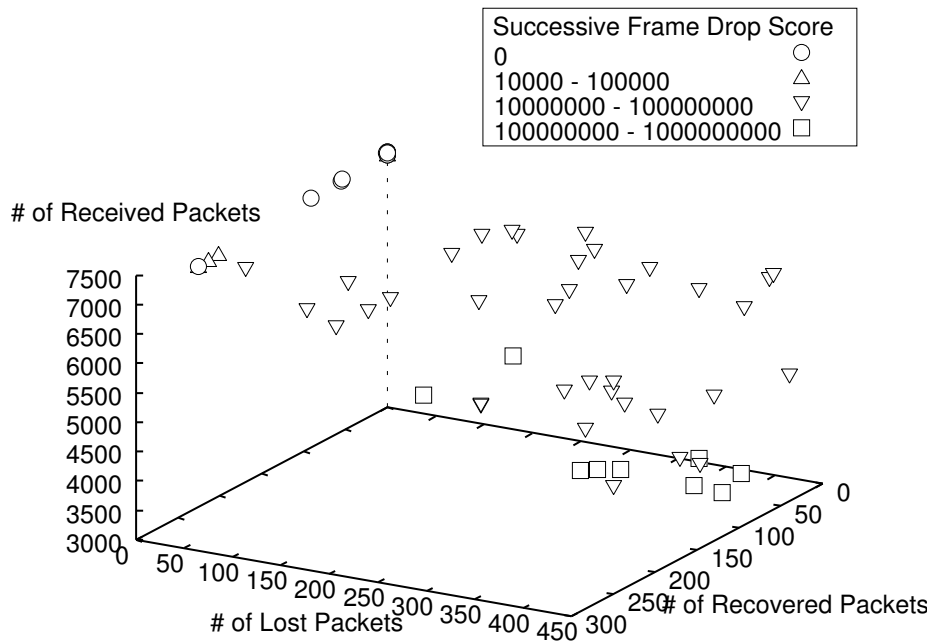


図 20 下落連続性 V_{down} による品質評価

響が出るのが分かった。サーバログ項目のうち、受信パケット数、喪失パケット数はクライアント再生品質劣化の発生と、サービスに影響を与えるだけのネットワーク環境劣化検出には有効である。ただし、微細な品質の劣化は検出できない場合がある。一方、回復パケット数はクライアントでの品質劣化が発生した時点では必ず増加が見られ、また品質劣化が発生する前段階のネットワーク環境劣化が起きた場合にも増加が見られた。このため、サーバログで回復パケット数を観測することは、クライアント再生品質把握とネットワーク環境把握に有効であることが明らかになった。

表 6 3 回の実験結果の標準偏差と平均値に対する割合

	標準偏差 σ	平均値に対する割合
受信パケット数	27.90	0.44
回復パケット数	8.042	0.79
喪失パケット数	12.02	3.40

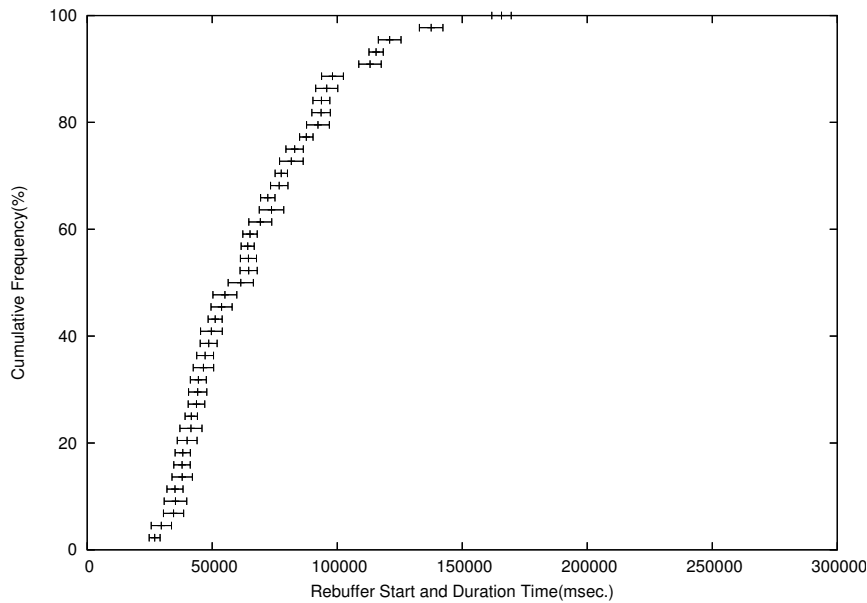


図 21 リバッファ発生タイミングと継続時間

5.10 実験結果の考察と実験条件に関する議論

回復パケット数の再生品質劣化への影響に関しては、以下の可能性が考えられる。再送発生でデータパケット到着タイミングが遅れた場合、アプリケーション内のデコード処理、表示といったクライアント内処理が加わることにより、再生タイミングに遅れることがある。これは品質劣化として検出される。この場合、クライアント負荷が高いほど回復パケットの影響は顕著になると考えられる。

実験では、各ネットワークパラメータ設定につき3回の実験を行い、平均値を採用した。3回の実験結果の分散を示すため、パケットロス率 $\alpha\%$ 、遅延 βms 、遅延ジッタ $\gamma\%$ の設定で行う3回の実験で、得られる受信パケット数をそれぞれ1回目 $x_{\alpha\beta\gamma 1}$ 、2回目 $x_{\alpha\beta\gamma 2}$ 、3回目 $x_{\alpha\beta\gamma 3}$ 、同様に回復パケット数を $y_{\alpha\beta\gamma 1}$ 、 $y_{\alpha\beta\gamma 2}$ 、 $y_{\alpha\beta\gamma 3}$ 、

喪失パケット数を $z_{\alpha\beta\gamma 1}$, $z_{\alpha\beta\gamma 2}$, $z_{\alpha\beta\gamma 3}$ とする．このときの標準偏差をそれぞれ受信パケットは $\sigma_{x\alpha\beta\gamma}$, 回復パケットは $\sigma_{y\alpha\beta\gamma}$, 喪失パケットは $\sigma_{z\alpha\beta\gamma}$ とする．また, それぞれの標準偏差の平均値に対する割合 $\frac{\sigma_{x\alpha\beta\gamma}}{\bar{x}}$, $\frac{\sigma_{y\alpha\beta\gamma}}{\bar{y}}$, $\frac{\sigma_{z\alpha\beta\gamma}}{\bar{z}}$ を算出する．ネットワーク設定ごとに得られるこれらの標準偏差値の平均と, 平均値に対する割合の平均を表 6 に示す．なお, 実験コンテンツにおける総パケット数は 7330 パケットである．ここで, 喪失パケット数の項目のみ, 平均値に対する割合が高くなっているのは, 喪失パケット数が最大時でも 300 パケット程度と, 総パケット数の 5%程度であるためであると考えられる．総パケット数と標準偏差の値を比較した場合, 3 回の実験でのばらつきは 1%以下に収まっており, 3 回の実験でも十分な結果を得られていると考える．

実験では 3 分間の配信を行ったが, 予備実験では 5 分間の実験を行っている．本研究ではリバッファの有無を品質評価基準のひとつとしたことから, リバッファが発生するネットワーク環境では少なくとも 1 回のリバッファが観測される必要があった．実験時間が短過ぎる場合, 起こるべきリバッファが観測できない可能性があるため, 最低限必要な実験時間を見積もる必要がある．ここで, 5 分間の実験において, リバッファの発生タイミングとリバッファ時間の長さを, リバッファが早く発生した順に累積的に並べたものを図 21 に示す．図より, 最も早くリバッファが発生したのは配信開始から 30 秒後, 最も遅い場合でも 160 秒でリバッファを終了している．以降, 300 秒間の実験でリバッファは観測されなかった．これより, リバッファが発生する場合には 3 分間の実験で観測が可能であるといえる．

実験で用いたネットワーク環境は, NIST Net によるネットワークシミュレートで合成しており, 実験は 1 台のクライアントで行った．配信時, サーバには数千のクライアントからのアクセスが予想される．本研究に先立って, Microsoft 社から提供されていたストリーミングクライアントエミュレーター Microsoft Load Simulator を用いて数千クライアントからのアクセスを擬似的に発生させた実験を行った．ただし, これは古いバージョンである Windows Media Server 8 を対象としており, 通信プロトコルは MMS となっている．エミュレートを行った結果, 接続数の増加に伴い, ネットワークではサーバ付近での輻輳が発生し, クラ

クライアントではパケットロス，ルータでの遅延，遅延ジッタ増大として観測された．また，サーバ負荷の上昇によって，サーバでの処理遅延が若干増加した．これはクライアントでは伝送遅延，ルータ遅延，サーバ処理遅延を足しあわせた遅延，遅延ジッタとして観測された．1クライアント接続時と1800クライアント接続時の遅延の差は1ms以内であった．なお，クライアント数を増加させた場合，セッションそのものが確立できなかったため，データは収集できていない．これらの現象は，実験環境で設定したネットワークパラメータに含まれる範囲であるため，本研究で得られた結果は，多数のクライアントが接続した場合でも適用可能と考える．

5.11 他のシステムへの適用性に関する議論

Windows Mediaのほか，代表的なストリーミングサーバシステムとしてはReal社のHelix Server，Apple社のDarwin Streaming Serverがある．これらも含めた現在のストリーミング配信システムでは，クライアントでのデータパケットの受信状態をRTSPプロトコルを用いてサーバに通知する仕組みとなっている．サーバログではRTSP制御コネクションから得た情報を記録するため，送信成功パケット数(受信パケット数)，再送信パケット数(回復パケット数)や再送失敗パケット数(喪失パケット数)と各システムとも項目は共通している．

また，現在これらのシステムで標準的に用いられているのはMPEG-4に準拠した圧縮方式となっている．今回用いた動画コーデックであるWindows Media Video 9もMPEG-4方式である．特に高画質なものにはMPEG-2が用いられることもあるが，これらはいずれもフレーム間圧縮方式である．再生品質劣化が起き始める閾値が異なる可能性はあるが，データロスによる映像への影響の傾向は基本的には共通していると予想される．

実装の違いが大きい部分として，配信状態が変化した際にサーバからの配信フレームレートを切り替えるタイミングがある．フレームレート切替え時にはクライアントではリバッファリングとして観測されるが，本論文ではフレームレート切替えが起こる以前の，フレーム落ちによる再生品質劣化を対象としているので，結果に大きな影響は与えないと考えている．

このようにストリームデータ転送，制御の枠組みは同様であることから，サーバログで得られる再送パケット数 (Windows Media における回復パケット数) が品質劣化を示す指標となる点，ネットワーク環境における遅延ジッタが再生品質により大きな影響を与える点に関しては同様の結果が得られると考えられる．

また，実験で用いた MPEG-4 のようなフレーム間圧縮方式では，厳密にはパケットロスによってどのフレームが喪失するかにより再生品質に影響が現れる可能性があった．このため，複数回数実験を行い，得られた結果に差が見られないことを確認した．

本論文では音声に関する品質評価は行っていないが，音声の途切れは，画像の乱れ，フレーム落ちよりも一般に敏感に知覚されるとされており，ユーザ視聴感に大きく影響すると考えられる．このため，動画像では1秒間1フレーム未満の下落は視認できなかったため検出しなかったが，音声の場合はより細かい時間単位での検出が求められると考えられる．クライアントでの音声品質評価を行い，本論文での評価手法を適用することにより，音声に関してもネットワーク環境，音声再生品質，サーバログの相関を導出することは可能である．

6. 得られた知見と課題

本研究では、仕様が公開されていないアプリケーションにおいて、運用時に管理者が容易に入手できるサーバログに着目し、クライアントでのサービス品質情報とサーバクライアント間のネットワーク状態情報の抽出をすすめてきた。サーバログが示す値から、クライアントで発生する再生状態の劣化を検出可能となった。また、サーバログに現れるパケット受信情報の分布から、サーバクライアント間のネットワーク状態を推定することができた。他に、ネットワーク状態がストリーミングサービスにどのような影響を与えるかが明らかになった。得られた結果に対するアプリケーション間、データ間の互換性、特に今後サービスが高品質化した際に考慮すべき点について考察する。また、得られた知見から運用時に着目すべき点を述べる。課題として、今後も増加するであろうブラックボックスであるシステムにおいて、評価に必要な情報の取得、評価の仕方を考える必要がある。

6.1 サーバログからのサービス品質情報抽出

サーバログに記録される受信パケット数、再送成功を示す回復パケット数、喪失パケット数はこれまでサービス品質との関連があいまいであった。特に、アプリケーションにおける再送成功パケット数を示す回復パケットは、再送が成功しているのであるから品質にはほぼ影響しないと考えられてきた。しかし、本研究での実験を通じ、再送失敗を示す喪失パケットが存在せず、かつ全てのパケットが1回あるいは再送により受信成功している場合でも、クライアントでの若干の品質劣化が起こることが判明した。これは回復パケット数がサービス品質の劣化を示すということだけでなく、劣化の前兆として注意すべきパラメータであることを示している。

アプリケーション内部のデコード状態観測は本研究の研究対象としては含まないが、回復パケットによる品質劣化は、デコード時のタイムラグが影響していると考えられる。パケット受信状態は、アプリケーションでのパケット受信タイミングと、そのパケットが含むデータの再生タイミングによって再送成功か失敗か

が判断される。このため、再生タイミングに非常に近いタイミングで受信したパケットはデコード中に再生タイミングを迎える場合がある。これはパケットレベルでは再送成功しているが、実際には再生時に欠落したデータであり、場合によってはフレーム落ちとして品質の劣化に影響する。

喪失パケット数が発生する段階では、再生時に連続したフレーム落ちが観測されており、品質は相当劣化しているといえる。この場合、回復パケット数は若干減少する。

回復パケット、喪失パケットの有無により、クライアントにおけるサービス品質をフレーム落ちレベルで検出可能となった。

6.2 サーバログからのネットワーク情報抽出

再送成功である回復パケット、喪失パケットが見られない場合、パケットロス率は0%に近いと判断できる。また、遅延ジッタが小さいことも示している。

回復パケットが発生し、喪失パケットが見られない場合は、回復パケットの値が少なければパケットロスの発生は0%に近く、回復パケットが増加していればパケットロスが発生していると判断できる。実験環境では回復パケット数が全パケット数の2.5%が閾値であった。閾値以下の場合はパケットロスはなく、遅延ジッタが増大している。なお、遅延そのものの大きさは配信品質には影響しない。

喪失パケットの検出時には、パケットロス率が相当高いか、あるいはパケットロスに加え遅延ジッタが大きいと判断できる。

サーバログの観測により、喪失パケットの有無、回復パケットの割合をみることでネットワーク状態の推測ができる。ただし、ネットワーク状態が著しく悪化した場合は逆に回復パケットの減少が見られ、大半を喪失パケットが占めるようになるので細かな判断は難しくなる。

6.3 ネットワーク状態がサービスに与える影響

ストリーミング配信では、回線品質において遅延、遅延ジッタの影響が非常に大きいと考えられてきており、実験においてもこれらの影響が大きいことを想定

し、パケットロスに加え様々な遅延時間，遅延ジッタを設定した．

本研究を通じて，遅延そのものの大きさは実際のインターネットで観測される程度であればクライアントでの品質劣化には直接影響しにくいことが明らかになった．ただし，ジッタ値が同じであっても，遅延の値が大きい場合は振幅も大きくなるため，クライアントバッファ内でのデータ到着順序入れ替わりなどが発生しやすくなり，再生品質に影響する可能性は高くなる．実験結果からも，遅延が大きい場合は遅延ジッタへの耐性が大きく落ちる現象が観測されている．

また，パケットロスに関しては，アプリケーションバッファの存在と，再送制御の存在により品質への影響は緩和されると考えられるが，実際にはパケットロスの影響は非常に大きいことが判明した．遅延が非常に小さい環境においても，パケットロスが発生した場合，品質は大幅に劣化する．さらに遅延，遅延ジッタとパケットロスが同時に発生した場合は，わずかなパケットロス率であっても再生不能に陥る現象も観測された．

以上より，ストリーミング配信においては，パケットロスが品質に直接影響する．特に，パケットロスが発生している場合，遅延そのものの大きさよりも遅延ジッタの影響が大きい．パケットロスが発生していない場合は，遅延，遅延ジッタとも品質への影響は非常に小さく，通常インターネット上で観測される程度の遅延，遅延ジッタであればクライアントにおける品質劣化はほとんど起きないと推測される．

6.4 アプリケーション間，データ間の互換性

アプリケーション互換性に関して，大規模ストリーミング配信では他に多く利用されているソフトウェアとして，Apple 社の Darwin Streaming シリーズ，RealNetworks 社の Real あるいは Helix シリーズなどがある．これらは仕様が公開になっているものもあれば，非公開のものもある．公開されている情報から，コーデックの具体的な実装や詳細な通信方式は不明な部分もあるが，MPEG4 をベースにしたコーデック，RTSP による制御などの通信方式は共通していることが分かった．ログの出力に関しても，クライアントアプリケーションでのパケッ

ト受信状態について，アプリケーション毎に名称は異なるが，受信，回復，喪失パケットと同様の項目が記録されることも確認した．

上記アプリケーションにおいて，同様の実験を補足的に行った結果，フレームレート変動パターンはほぼ同様に観測され，閾値に差はあるもののネットワーク状態との相関，ログ出力との相関に関して同じ傾向を得ることができた．

6.5 運用に対する知見

運用時には，サーバログ項目の中でも特に，再送成功パケットである回復パケットに注目すべきである．回復パケットの発生により，クライアントでは再生時にフレーム落ちが発生している可能性が高い．喪失パケットが検出された場合は，クライアントでの再生状態は相当悪化していると判断してよい．喪失パケットが増加，かつ回復パケットが減少している場合はネットワーク状態が通信困難なレベルに達しているといえる．

ログにはクライアントの IP アドレスが記録されるため，ドメイン毎に集約して観測を行うことにより，通信，サービスの状態をまとめて観測することも可能であろう．特定のドメインに属するクライアント群で回復パケットの増加が見られれば，ネットワーク経路上に何らかの障害が起きていると推測できる．

6.6 今後の課題

本研究では，従来のように複雑な挙動の観測が行えないシステムとして，大規模ストリーミング配信システムを対象に運用情報抽出を行ってきた．対象システムでは，アプリケーションがオープンシステムでないことだけでなく，運用上の制約からネットワーク上でのパケットに関する詳細な情報も得ることが困難であるという前提で，限られた情報をソースに分析を行ってきた．いわば，インターネットとアプリケーションというミクロな挙動解析が困難な2つの部分に関して，入出力のみを観測して状態を予測する試みである．

インターネット上でのネットワーク構成はVPN，トンネリングが普及してきており，データは暗号化され，商用製品による特殊なプロトコルによるデータが

氾濫している．データ量も膨大なものとなっており，従来のようにデータをくまなく詳細に解析することは年々難しくなっている．アプリケーションもオープンソースでアーキテクチャが公開されたものが増加する一方で，クローズドなシステムも同様に数多く開発されている．これらに対してはリバースエンジニアリングによる詳細な解析が行われることも多いが，開発速度が加速するとコミュニティベースで追いつかない場合もしばしばある．なにより，バージョンアップによる仕様の大幅な変更が起きる度最初から解析をやり直すコストは非常に大きい．こういった背景を踏まえて，外部で得られる情報のみを収集し，詳細な分析を行うことで新たな情報を生み出す手法は今後不可欠なものになると予想される．

本研究では，サービス品質評価値としてクライアント再生状態を定義し，アプリケーションからフレームレートを取得，評価を行った．フレームレートの取得は，アプリケーションから SDK として提供されていたため利用可能であったが，本来はシステム内部情報であり，収集が難しい可能性もあった．この場合，外部アプリケーションによる画質の客観評価，あるいは人による主観評価などの他の手段を用いる必要がある．本研究におけるサーバログのように，最終的に運用時に利用する外部出力情報はシステムへの変更を必要とせず得られる情報であることが望ましい．しかし，必要な情報を得るためのステップとして，今回のサービス品質分析のようにシステム内部からの情報が必要となる場合がある．完全なブラックボックスシステムの分析を行う場合，どのようにして必要な情報を得て，評価を行うのかは映像，音声評価技術も含めて，今後検討していく必要がある．

7. 結論

ストリーミングの高品質化に伴い、サービス品質管理の重要性がますます重要になってきている。サービス品質管理には、サーバ側でのサービス品質情報把握が必須である。一方で、実運用システムへの適用を考えた場合、情報収集方法に大きな制約がある。加えて、ストリーミングでは仕様が公開されていない商用アプリケーションが主導的であり、内部構造からの分析は困難であった。このため、運用時に必要な情報が不足しており、サービス品質判断基準も明確でないために管理者の経験と勘によって判断されることが多いという実情があった。

内部解析が困難なシステムに対しては、外部から得られる限定的な情報の相互関係から新たな情報を抽出する必要がある。本研究では、運用時に入手可能なサーバログを用いて、クライアントにおけるサービス品質である再生状態と、サーバクライアント間ネットワーク状態の判断するための手法と基準を示した。その段階でまず、ストリーミングに適した再生状態の定量的判断基準を定義する必要があった。これは、既存の動画像評価手法にストリーミングで起こる様々な品質劣化現象を考慮したものが存在しないためである。再生状態では、ストリーミング特有の現象である、パケット不着によるフレーム落ち、再生中断、レート変更時に発生するリバッファについて考慮する必要がある。クライアントでフレームレートの変動を記録し、フレームレートの変動から視認可能なフレーム落ちを定量化した。これにより、エンドユーザの視聴品質に近い品質評価値を得ることができる。

実験環境下で様々なネットワーク状態をシミュレートし、サービス品質情報、ログ情報を収集、分析を行った。その結果、サーバログからのサービス品質判断、ネットワーク状態の推測が可能になった。また、ストリーミングサービスに影響を与えるネットワーク状態についても知ることができた。運用時、クライアント毎のサービス品質やサーバクライアント間のネットワークを推定するために、着目すべきログ内パラメータとパラメータの状態について述べた。また、一般に品質劣化に無関係とされてきたパラメータに関しても、フレーム落ちなどの微細な品質劣化を示すことが明らかになった。これにより、経験則を多く持たない管理者でも、サービス品質を把握することが可能である。

ストリーミングのような内部構造の解析が困難なブラックボックスシステムは、今後も増加すると予想される。本研究で、このようなシステムに対し外部情報の分析が有効であることを示したことにより、これまで情報が少なく詳細な解析が不可能とされる対象からの情報抽出に一つの方向性を示した。

謝辞

本研究を行う機会を与えていただき、研究方針はもちろんのこと、研究活動全般にわたって熱心な御指導をしていただきました、情報科学研究科 砂原 秀樹 教授に心から感謝の意を表します。

副指導教官であり、研究に関する多くの御助言や叱咤激励をいただきました、情報科学研究科 山口 英 教授に深く感謝致します。

研究方針に関する御指導をはじめ、日々の研究活動において多くの御助言をいただきました、情報科学研究科 藤川 和利 助教授に深く感謝致します。

情報科学研究科 河合 栄治 助教授には、常に研究の方針や研究内容に関して御指導いただきました。加えて、研究姿勢をはじめ研究の基礎となる様々なことから関しても多くの御指導をいただきましたことを重ねて感謝致します。

研究に関する御助言をいただき、研究の基礎となるネットワーク運用に接する様々な機会を与えていただきました、附属図書館研究開発室 森島 直人 助手に心より感謝致します。

本研究に先立ち、ストリーミング運用現場での実験機会を与えてくださいましたライブ!ユニバースの皆様、実験で様々なご配慮をいただきました、東京大学 中山 雅哉 助教授、ストリーミング運用に関する貴重なアドバイスや、実験への手引きをいただきました IJ の山本 文治 氏に心より感謝致します。

本研究をはじめのあたり、サーバ運用支援研究に関する様々な御助言をいただきました、九州工業大学の中村 豊 助教授、大阪大学大学院の中山 貴夫 助手に心より感謝致します。

Windows Media に関してプログラミング上の多大な情報をいただきました WIDE プロジェクトの小川 浩司 氏、DVTS をはじめ動画配信に関する情報をいただきました、NTT ネットワークサービスシステム研究所の入野 仁志 氏に心より感謝致します。

研究生生活において様々なアドバイスをいただき、研究以外の面でも大変お世話になりました、情報科学研究科 和泉 順子 助手、情報科学研究科 垣内 正年 助手、情報科学研究科 島田 秀輝 助手に心より感謝致します。

さまざまな面から研究活動を支えて下さった、情報科学センターの職員の皆様、

事務補佐員の呂 悠妃 女史，東京大学 田坂 佳苗 女史に心より感謝致します。また，インターネット・アーキテクチャ講座の学生，OB，OGの皆様には入学当初から様々な面で研究活動を支えていただきました。心より感謝致します。中でも特に戸辺 論氏，千葉 周一郎氏には様々な場面で研究活動を支えていただきました。心より感謝致します。

最後に，研究活動を陰で支えてくれた家族に感謝します。

参考文献

- [1] 平成 18 年度版 情報通信白書, 総務省. 2006.
- [2] 2006 年 9 月度インターネット利用動向情報サービス調査, ネットレイティングス株式会社.
- [3] インターネット白書 2006, Internet Association Japan.
- [4] Dapeng Wu, Yiwei Thomas Hou, Wenwu Zhu, Ya-Qin Zhang, and Jon M. Peha. Streaming video over the internet: Approaches and directions. *IEEE Transactions on Circuits and Systems for Video Technology*, Vol. 11, No. 3, pp. 282–300, Mar 2001.
- [5] The SDP Multimedia website for ASF download and MMS protocol specification, SDP Multimedia Team. <http://sdp.ppona.com/>.
- [6] Apache. <http://www.apache.org/>.
- [7] RFC 1819: Internet Stream Protocol Version 2 (ST2) Protocol Specification - Version ST2+, Luca Delgrossi and Louis Berger. Status: EXPERIMENTAL. <http://www.ietf.org/rfc/rfc1819.txt>, August 1995.
- [8] Real products, RealNetworks Inc. <http://www.realnetworks.com/products/>.
- [9] RFC 2326: Real Time Streaming Protocol (RTSP), Henning Schulzrinne, Anup Rao, and Robert Lanphier. Status: PROPOSED STANDARD. <http://www.ietf.org/rfc/rfc2326.txt>, April 1998.
- [10] RFC 3551: RTP Profile for Audio and Video Conferences with Minimal Control, Henning Schulzrinne and Stephen L. Casner. Status: STANDARDS TRACK. <http://www.ietf.org/rfc/rfc3551.txt>, July 2003.
- [11] Windows media technology, Microsoft Inc. <http://www.microsoft.com/windows/windowsmedia/>.

- [12] Helix. <https://helixcommunity.org/>.
- [13] Quicktime streaming server, Apple Computer Inc. <http://www.apple.com/quicktime/streamingserver/>.
- [14] Darwin streaming server. <http://developer.apple.com/opensource/server/streaming/>.
- [15] Amy Reibman, Subhabrata Sen, and Jacobus Van der Merwe. Video quality estimation for internet streaming. In *in Proceedings of the 14th International Conference on World Wide Web*, pp. 1168–1169. ACM, 2005.
- [16] Prasad Calyam, Dima Krymskiy, Mukundan Sridharan, and Paul Schopis. Active and Passive Measurements on Campus, Regional and National Network Backbone Paths. In *Proceedings of International Conference on Computer Communications and Networks (ICCCN 2005)*, pp. 537–542, October 2005.
- [17] Warren Matthews and les Cottrell. Internet end-to-end performance monitoring for the high energy nuclear and particle physics community. In *Proceedings of Passive and Active Measurement Workshop 2000 (PAM 2000)*, April 2000.
- [18] Yubing Wang, Mark Claypool, and Zheng Zuo. An Empirical Study of Realvideo Performance Across the Iinternet. In *Proceedings of the 1st ACM SIGCOMM Workshop on Internet Measurement*, pp. 295–309, November 2001.
- [19] Amy Csizmar Dalal and Ed Perry. A New Architecture for Measuring and Assessing Streaming Media Quality. In *Proceedings of the Third Workshop on Passive and Active Measurement Workshop (PAM 2003)*, pp. 223–231, La Jolla, CA, April 2003.

- [20] 大浴孝治, 山本文治, 中山雅哉, 門林雄基. 再生統計情報に着目したストリーミング品質評価方式とその効果. 情報処理学会研究報告 高品質インターネット, 第2巻, pp. 105–110, January 2005.
- [21] H. Koyun. *Loggin Model for Windows Media Services 9 Series*, Mar. 2003. <http://www.microsoft.com/windows/windowsmedia/howto/articles/LoggingMod%el.aspx>.
- [22] Maureen Chesire, Alec Wolman, Geoffrey M. Voelker, and Henry M. Levy. Measurement and Analysis of a StreamingMedia Workload. In *Proceedings of 3rd USENIX Symposium on Internet Technologies and Systems*, March 2001.
- [23] Ludmila Cherkasova and Minaxi Gupta. Analysis of Enterprise Media Server Workloads: Access Patterns, Locality, Content Evolution, and Rates of Change. In *IEEE/ACM Transactions on Networking (TON)*, Vol. 12, pp. 781–794, October 2004.
- [24] Kunwadee Sripanidkulchai, Bruce Maggs, and Hu Zhang. An Analysis of Live Streaming Workloads on the Internet. In *Proceedings of the 4th ACM SIGCOMM Conference on Internet Measurement*, pp. 41–54, October 2004.
- [25] Lei Guo, Songging Chen, Zhen Xiao, and Xiaodong Zhang. Analysis of Multimedia Workloads with Implications for Internet Streaming. In *Proceedings of the 14th International Conference on World Wide Web*, pp. 519–528, May 2005.
- [26] Eveline Veloso, Virglio Almeida, Wagner Meira, Azer Bestavros, and Shudong Jin. A Hierarchical Characterization of a Live Streaming Media Workload. In *Proceedings of the ACM SIGCOMM Internet Measurement Workshop*, pp. 117–130, November 2002.
- [27] Dmitri Loguinov and Hayder Radha. Measurement Study of Low-bitrate

Internet Video Streaming. In *Proceedings of the 1st ACM SIGCOMM Workshop on Internet Measurement*, pp. 281–293, November 2001.

- [28] Z. Wang, S. Banerjee, and S. Jamin. Studying Streaming Video Quality: From An Application Point of View. In *Proceedings of the Eleventh ACM International Conference on Multimedia*, pp. 327–330, Berkeley, CA, November 2003. ACM Press.
- [29] Methodology for the subjective assesment of the quality of television pictures, ITU–R Recommendation BT.500–11. June 2002.
- [30] Objective perceptual video quality measurement techniques for digital cable television in the presence of a full reference, ITU-T Recommendation J.144.
- [31] 間伸一, 渡部優, 片山頼明. ネットワーク配信映像の視聴品質推定技術. NTT 技術ジャーナル 2006 年 7 月号, NTT コミュニケーションズ, 2005.
- [32] NIST Net: A Linux-based Network Emulation Tool, Mark Carson and Darin Santay. <http://www-x.antd.nist.gov/nistnet/>, Jul. 2003.

著者研究業績

論文誌

1. 寺田 直美, 河合 栄治, 砂原 秀樹. “クライアント品質推定のための詳細なストリーミングサーバログ分析”. 情報処理学会論文誌, 第 47 巻第 7 号, pp.2019–2029, Jul. 2006.

国際会議

1. Naomi TERADA, Eiji KAWAI and Hideki SUNAHARA. “Extracting Client-Side Streaming QoS Information from Server Logs”. in Proceedings of the IEEE Pacific Rim Conference on Communications, Computers and Signal Processing(PacRim2005). pp.621–624, August 2005.

国内研究会

1. 寺田 直美, 河合 栄治, 砂原 秀樹. “ストリーミングサービスにおけるクライアント上のサービス品質推定のためのサーバログ分析手法の提案”, 情報処理学会 マルチメディア, 分散, 協調とモバイル (DICOMO2005) シンポジウム. pp.769–772, July 2005.
2. 寺田 直美, 中山 貴夫, 中村 豊, 砂原 秀樹. “ストリーミング配信サーバにおける管理支援システムの設計”. 電子情報通信学会 技術研究報告, Vol.102, No.497, pp.13–18, December 2002.
3. 寺田 直美, 中山 貴夫, 中村 豊, 砂原 秀樹. “ストリーミング配信サーバにおける管理支援システムの設計”. インターネットコンファレンス 2002 論文集 (Work in Progress), pp.118, Oct, 2002.