

自動車遠隔運転に向けた WebRTC を用いた 映像伝送における MP-QUIC の適用

金子 直矢^{1,a)} 伊東 孝紘² 勝田 肇² 渡辺 敏暢² 岡田 和也¹ 阿部 博¹ 大西 亮吉^{1,2}

概要: 自動車の遠隔運転では、従来運転者が車内で知覚していた車の視覚・聴覚情報を遠隔地にいる運転オペレータに対して安定的に伝送しなければならない。本研究は、移動体通信回線を備えた自動車から遠隔地のオペレータ端末への映像伝送に WebRTC を用いる環境を想定し、通信品質向上への MP-QUIC プロトコルの適用可能性検証を行う。WebRTC は、インターネットにおいて端末・サーバ間で P2P 通信路を確立し通信を行うため、遠隔運転に必要な低遅延な通信を実現できる可能性がある。しかし、移動する自動車からの通信は不安定になることもあり、遠隔運転の安全性を高めるためにより安定した通信方式を確立しなければならない。そこで、本研究では複数の異なる通信回線を利用して通信を行うマルチパス通信プロトコルの一つである MP-QUIC を採用することにより、WebRTC 通信の可用性および品質の向上を試みる。MP-QUIC は、ユーザランドで動作し低遅延なトランスポートプロトコルである QUIC を対象にマルチパス通信技術を導入しており、遠隔運転に必要な通信品質を安定的に提供できる可能性がある。評価では、WebRTC を用いた映像伝送システムに対して MP-QUIC の組み込み、市街地を走行する自動車から複数の移動体通信回線を用いた映像メディアの伝送を実施し、遅延および映像品質の計測を行う。

Applying MP-QUIC Protocol for WebRTC-based Video Communication on Teleoperated Driving

1. 背景と目的

通信機能を備えた自動車（コネクティッドカー）の普及が進んでいる。コネクティッドカーは、Wi-Fi または移動体通信網（3G/4G/5G）に接続し、外部からの交通情報の受信や運行状況管理機能や遠隔オペレータサービスといった付加価値を提供している。今後は、コネクティッドカーの特性を活かし、自動車を遠隔地から操作する遠隔運転支援サービスの実現が期待されている。

遠隔運転の実現には、車両周辺の映像、音声、車両のスピードといった情報を自動車から操作拠点に低遅延でかつ安定して転送しなければならない。しかし、既存のコネクティッドカーでは同時に 1 つの通信回線しか利用できないため、場所によっては通信が不安定な状況になったり、通信の断絶が発生する可能性がある。そのため遠隔運転に必

要なデータ転送を行うための通信帯域と遅延を確保できる通信方式が必要不可欠である。

筆者らは先行研究 [1] にて、安定した通信を確保するために複数の異なる通信回線を併用するマルチパス通信技術を WebRTC 映像伝送システムに適用する手法を提案した。提案システムは、車両側端末・中継サーバ端末にそれぞれ UDP プロキシを組み込み、WebRTC を用いて通信する車両側端末・操作拠点端末からのパケットを複数のリンクに冗長・振り分けて通信を行う。プロキシが通信内容に応じて複数の通信回線へ動的に振り分けることで端末間の通信品質を安定化した。映像・音声伝送には WebRTC を採用し、他のプロトコルと比べて低遅延な伝送が可能とした。

本研究では、QUIC [2] にマルチパス通信技術を導入した Multipath QUIC (MP-QUIC) [3] を用いた WebRTC へのマルチパス通信技術の導入手法を提案する。また、提案手法を実装したシステムを自動車に搭載し、市街地を走行する自動車と遠隔地の端末との間で映像伝送を行い、MP-QUIC の適用可能性および効用の評価を行う。QUIC は、OS 内のユーザランドで動作し導入が容易であること

¹ トヨタ自動車株式会社

Toyota Motor Corporation

² ウーブン・プラネット・ホールディングス株式会社

Woven Planet Holdings, Inc.

a) naoya_kaneko_aa@mail.toyota.co.jp

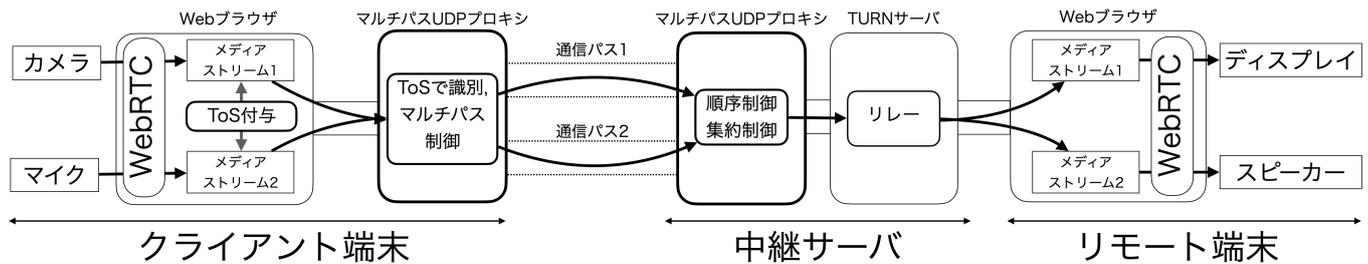


図 1 提案システムの構成図

に加えて、輻輳制御と再送制御を備えているため UDP をトランスポートプロトコルとして用いた先行研究の方式と比べてより安定的な通信環境を実現できる可能性がある。

本論文の構成は以下の通りである。第 2 章では関連研究を紹介する。第 3 章では、WebRTC と MP-QUIC を組み合わせる提案手法について述べる。第 4 章では、提案システムの実装について述べる。第 5 章では、提案システムの評価を行う。第 6 章では、第 5 章で得られた結果に基づき考察を行う。最後にまとめと今後の課題について整理する。

2. 関連研究

QUIC [2] にマルチパス通信機能を導入した研究として文献 [3, 4] が存在する。これらの研究では、スマートフォンにおける移動体通信と Wi-Fi の併用に MP-QUIC を適用し検証を行なっている。本研究では、自動車走行環境を対象として複数の移動体通信の併用に MP-QUIC を適用し検証を行う。自動車走行は公道が中心であるため、可用性が高く安価な通信手段が移動体通信サービスに限定される。

文献 [5] は、複数の移動体通信回線を対象に MP-QUIC を導入し通信の選択的冗長化手法を提案している。車載通信の遅延を模擬した環境で 2 種類の定常的な通信を行い、提案手法が通信の遅延を低減し通信の安定化に寄与することを示している。本研究では、市街地を走行する自動車から映像伝送を行う環境に対して MP-QUIC を導入する。実際の環境では、遅延以外にもジッタやパケットロスの発生により通信品質の低下が発生するため、MP-QUIC がこれらの事象に対して有効であるか検証が必要である。

映像伝送を対象に MP-QUIC を適用した研究として文献 [6] が存在する。この研究では、スマートフォンを対象とした映像配信にマルチパス通信技術を導入し、映像再生端末のバッファ状態など映像再生品質の制御と利用通信回線の制御を連携させている。これにより、通信品質が劣化した通信回線に送信した映像フレームを必要に応じて別の通信回線に再挿入可能となり、映像品質の向上を図っている。文献では Content Delivery Network (CDN) からスマートフォンへの映像配信を対象としており、低遅延映像伝送を対象としておらず、遠隔運転・操作の知覚情報伝送とは逆方向の通信へマルチパス通信技術を適用している。文献 [7] が述べているように、移動体通信では、端末から

のアップロードトラフィックと端末へのダウンロードトラフィックで通信品質が異なるため、文献が提案する映像フレーム再挿入手法の、低遅延映像伝送を必要とする遠隔運転での有効性検証が必要である。

マルチパス通信を可能とするトランスポート技術の代表例として Multipath TCP (MPTCP) [8], Multipath Datagram Congestion Control Protocol (MP-DCCP) [9], Multipath Stream Control Transmission Protocol (MP-SCTP) [10] が存在する。既に MPTCP は Linux カーネルバージョン 5.6 から、カーネル内のネットワークスタックにトランスポートプロトコルの 1 つとして実装が導入されているが、クライアントとサーバそれぞれでカーネルの対応と設定が必要である。一方で QUIC および MP-QUIC は OS のユーザランドで実装されており、文献 [11] が述べているように、カーネルでの機能追加およびバージョンアップなしに利用でき導入や最新規格追従の容易さが利点として挙げられる。また QUIC は UDP の上のプロトコルとして実装されているため、TCP/MPTCP と同様にインターネット経路上のファイヤウォール、プロキシなどのミドルボックスにより破棄される可能性が DCCP や SCTP に比べて低く、可用性が高い。

3. 提案手法

本章では、先行研究 [1] にて提案したマルチパス UDP プロキシを用いた WebRTC の通信制御手法について述べ、MP-QUIC を導入する手法について述べる。

3.1 UDP プロキシを用いたマルチパス制御手法

筆者らが、先行研究 [1] で提案した WebRTC の通信制御システムの構成を図 1 に示す。この論文では、WebRTC のマルチメディアストリームをマルチパス制御するために以下の 3 手法を組み合わせている。

- (1) IP Type of Service (ToS) 値を用いたパケットごとのメディアストリーム識別
- (2) Traversal Using Relays around NAT (TURN) サーバを用いた通信経路の制限
- (3) UDP プロキシを用いたマルチパス制御

先行研究では、WebRTC 内の映像および音声ストリームを識別するために Web ブラウザ内の API を拡張し IP ToS

値の付与を提案している。WebRTCは通信プロトコルとして、映像および音声にSRTP over DTLS over UDP、データにSCTP over DTLS over UDPを用いている。そのためパケットのペイロードがDTLSにより暗号化された同じヘッダの通信と認知され、Webブラウザ外部でメディアストリームの種類に応じた通信制御ができない。そこで先行研究ではWebブラウザ内部で、Synchronization Source (SSRC) 識別子に対応したIP ToS値をパケットごとに付与することで本問題を解決した。

またWebRTCがWebブラウザ間で確立する通信路が動的に決定され通信制御が困難である問題に対して、中継サーバの利用を必須とすることで制御のための介入を容易にした。本手法によりWebブラウザは中継サーバへの通信を固定的に行うため、リダイレクトまたはプロキシによる通信制御が可能となる。WebRTCはInteractive Connectivity Establishment (ICE)を用いて、端末が備える通信回線の組み合わせから最適な通信路を動的に決定する。どの通信回線が用いられ、P2Pまたは中継サーバを介する通信方式のいずれが利用されるかはWebRTC通信の開始時に決定されるため、通信制御のための介入が困難である。本提案ではNAT環境下にある端末間の通信をリレーする中継サーバとしてTURNサーバを導入した。TURNサーバが介在することによりWebブラウザ間はP2Pではなくなり更にTURNサーバの転送処理で最大で7ms程度の遅延増加が発生する。一方で移動体通信回線の遅延は数十から百数十msの幅で変動しておりTURNサーバの処理遅延よりも大きい。そのため、マルチパス通信技術を用いてオーバーヘッド以上の遅延低減および安定化が見込める場合には中継サーバ導入は許容可能であると考えられる。

マルチパス通信制御はUDPプロキシ(マルチパスUDPプロキシ)を用いて行う。マルチパスUDPプロキシは、端末の持つ複数の通信手段を用いて通信を行い、待受ポートまたはIP ToS値により識別されるメディアストリームに基づいて通信手段の冗長化や選択的利用を実現する。プロキシとして導入することで、WebRTCの通信に対して選択的な通信制御を可能とした。

3.2 MP-QUICを用いたマルチパス制御

本研究では、前節で述べたマルチパスUDPプロキシに代わってMP-QUICを用いたプロキシ(MP-QUICプロキシ)を導入した手法を提案する。MP-QUICプロキシの動作概要を図2に示す。

MP-QUICプロキシはマルチパスUDPプロキシと同じく、UDPのトラフィックを受けてマルチパス通信制御を行う。マルチパスUDPプロキシはUDPデータグラムによるカプセル化を行うが、MP-QUICプロキシはQUICストリーム内のメッセージへ変換する。MP-QUICプロキシは、UDPの宛先アドレスおよび宛先ポートとQUICスト

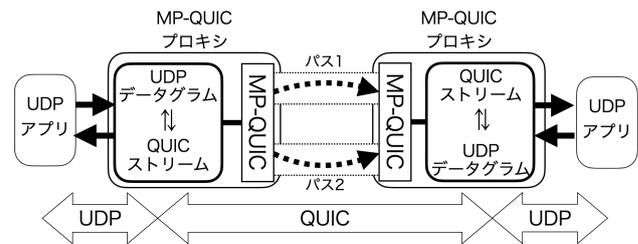


図2 MP-QUICプロキシの構成図

リームの対応関係を識別し管理する。

4. 実装

本研究では、文献[3]の研究成果として公開されている実装^{*1}(mpquic-goと呼称する)を用いてMP-QUICプロキシの実装を行う。mpquic-goの実装ではプロキシの実現に不十分であったため、以下の変更を行った。

- (1) パス検出ロジックの変更
- (2) 統計情報の収集・取得機能の追加

mpquic-goは、マルチパス機能を有効にした場合に端末の持つインタフェースのうちeth, rmmnet, wlanをプレフィクスに持つ名前前のインタフェースを利用するようにハードコードされている。本研究では移動通信回線のデバイス(モバイルルータ, モバイルUSBドングル)を用いるため、必要なインタフェース名を識別子として追加する。

マルチパス通信が行われていることを検証するためには、トランスポート層の統計情報の観測が重要である。一方でmpquic-goはパスごとの送受信トラフィックの統計値を記録していない。mpquic-goが通信に利用したパスを評価するため、パスごとに送受信パケット数、送受信バイト数を記録し、QUIC Pingフレームで計測されるRTTを取得可能なAPIを追加する。

QUICには、UDPデータグラムに相当する通信方式であるQUICデータグラム[12]が存在する。QUICデータグラムを用いることで、UDPデータグラムとの変換を容易に実装しつつマルチパス通信技術を導入できると考えられる。本研究ではmpquic-goがQUICデータグラムが実装していない点、輻輳制御および再送制御が適用されない点からQUICデータグラムを利用しないこととした。

5. 評価

提案手法の実現性および有効性を確認するために、自動車走行環境での評価を行った。自動車走行環境では、市街地を走行する試験車に搭載した端末から遠隔地の端末へWebRTCを用いた映像伝送を行う。マルチパス制御を行わず1つの通信回線を用いてTURNサーバのみを介する場合(実験構成1)と、MP-QUICプロキシを介する場合(実験構成2)の通信品質および映像品質を計測し評価を行

^{*1} <https://github.com/qdeconinck/mp-quic>

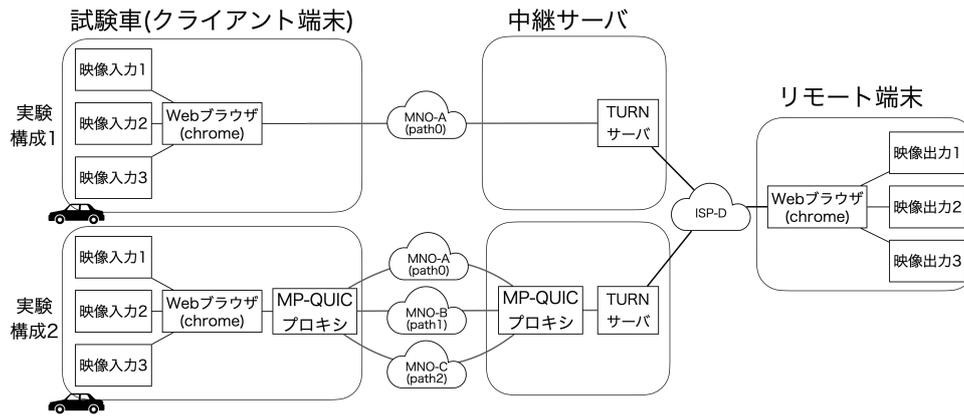


図 3 評価環境構成図

う。MP-QUIC プロキシを用いる場合は、MP-QUIC プロキシにてマルチパス通信の統計情報を計測し、複数の通信回線を用いた通信が行われていることを確認する。

5.1 評価環境

本稿で用いた評価環境のネットワーク構成を図 3 に示す。実験構成 1 と実験構成 2 は、MP-QUIC プロキシと複数移動体通信回線の有無が異なる。

クライアント端末は、リモート端末へ WebRTC を用いて 360p 30FPS の映像を 3 つ送信する。クライアント端末は 3 つのモバイルルータを用いて、MNO の異なる 3 つの移動体通信回線サービスへ接続し通信を行う。本稿では、それぞれの移動体通信回線を path0, path1, path2 と呼称する。実験構成 1 の場合、クライアント端末は path0 のみを用いる。実験構成 2 の場合、クライアント端末は MP-QUIC プロキシを通して 3 つ全ての通信回線を用いる。クライアント端末は、試験車に搭載し 12 時台に、三島駅北口(静岡県三島市文京町)から三島塚原 IC(三島市塚原新田)まで 10 分間走行する。

リモート端末は、クライアント端末から映像を受信しディスプレイに出力を行う。リモート端末は安定的な通信が可能な有線 LAN を用いてインターネットに接続する。

中継サーバは、Amazon Web Service の EC2 インスタンス上に構築され、TURN サーバ機能を提供する。また中継サーバは実験構成 2 の場合、MP-QUIC プロキシを用いてクライアントの MP-QUIC プロキシからの通信を集約し、TURN サーバに転送する。

クライアント端末、リモート端末および中継サーバは Ubuntu 20.04 を用いる。クライアント端末およびリモート端末は、WebRTC を利用するため Google Chrome 86.0.4240.198 を用いる。中継サーバで動作させる TURN サーバは coturn 4.5.1.3 を用いる。

5.1.1 MP-QUIC プロキシ統計情報の評価

本節では、MP-QUIC プロキシの統計情報からマルチパ

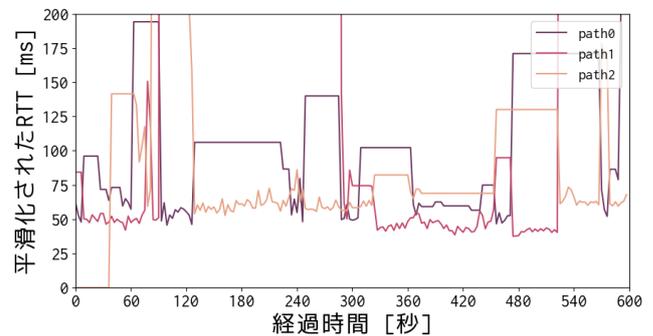


図 4 MP-QUIC プロキシ: 各通信回線の RTT (250ms 以下)

ス通信の動作を評価する。クライアント端末の MP-QUIC プロキシが、WebRTC 映像伝送を受けて行った通信を計測し、マルチパス制御の有無を確認する。統計情報は、MP-QUIC プロキシから 3 秒間隔で取得する。

MP-QUIC がプロトコル内部で各回線の品質情報として比較に用いている平滑化された RTT の推移を図 4 に示す。この図では、最小 RTT を示す回線の推移を評価するために、3 つの通信回線の RTT の中央値を中心とする範囲のみを抜粋した。評価実験中、用いた 3 つの通信回線のいずれにおいても RTT が他の 2 つの回線と比較して最小となる期間がある。いずれの通信回線の RTT も、評価実験中の一部の時間に値が変化していない期間がある。評価実験中に、最小 RTT を示した期間が最も長かった通信回線は path1 である。

MP-QUIC プロキシが各移動体通信回線へ送信したトラフィックのスループットを図 5 に、累計送信パケット数の推移を図 6 に示す。最大の送信スループットおよび最大の累計送信パケット数が計測されたのは path1 である。path1 は、最小 RTT を示した期間が最も長かった通信回線である。評価実験中、3 つの通信回線の全てで送信トラフィックが発生し、時刻ごとにそれぞれの送信スループットは変化している。また一部の時刻において複数の通信回線に同時に送信トラフィックが発生している。

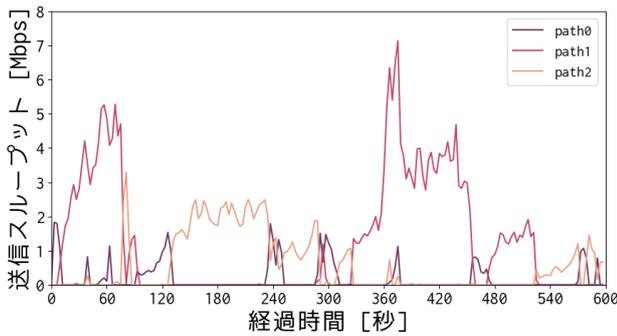


図 5 MP-QUIC プロキシ: 各通信回線への送信スループット

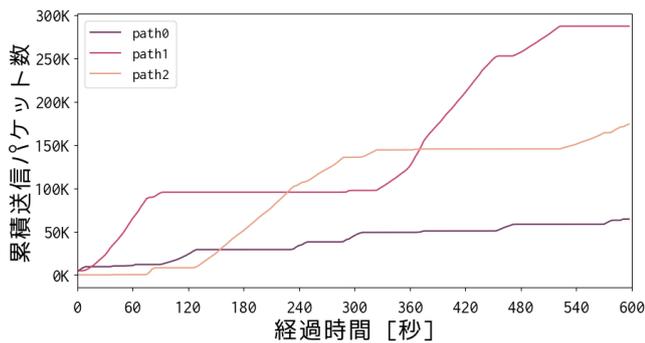


図 6 MP-QUIC プロキシ: 各通信回線への累積送信パケット数

5.1.2 WebRTC 統計情報の評価

本節では、クライアント端末およびリモート端末の Web ブラウザより WebRTC 統計情報を採取し、WebRTC レイヤから見た通信品質および映像伝送の品質を評価する。WebRTC 統計情報として、Web ブラウザ間の RTT、リモート端末の Web ブラウザの受信スループットの推移、単位時間あたりにデコードされたフレーム数、受信した映像の画質として画幅を計測する。WebRTC 統計情報は、Web ブラウザより 1 秒間隔で WebRTC Statistics API [13] を用いて取得する。本節で図示する CDF では、WebRTC の通信が行えていた期間の値を図示し、通信途絶により統計値が得られない期間を含めない。

WebRTC から見た通信品質の指標として、WebRTC が計測する RTT の統計を表 1 に示す。また文献 [14,15] が遠隔操作環境の遅延の許容値として例示する一方向遅延 100ms の RTT を想定して、200ms を上限とする範囲の RTT の CDF を図 7 に示す。WebRTC は、通信を行う Web ブラウザ間でメディアストリームごとに RTP Control Protocol (RTCP) の Sender Report と Receiver Report を用いた RTT 計測 (googRTT) を 1 秒ごとに行う。次の RTT 計測の完了までに Receiver Report の応答がない場合は、パケットロスにより RTT 計測に失敗したと判断する。表 1 の RTT 最小・最大・平均・標準偏差はパケットロスにより計測に失敗した期間の値を含まない。

WebRTC の RTT は、実験構成 2 が実験構成 1 よりも高い最大値および標準偏差を示し、最小値および平均値は同

表 1 WebRTC: RTT の統計

	実験構成 1	実験構成 2
RTT 最小 [ms]	51.0	52.0
RTT 最大 [ms]	192.0	1361.0
RTT 平均 [ms]	79.4	80.3
RTT 標準偏差 [ms]	14.9	46.5
ロス率 [%]	48.5	0.0

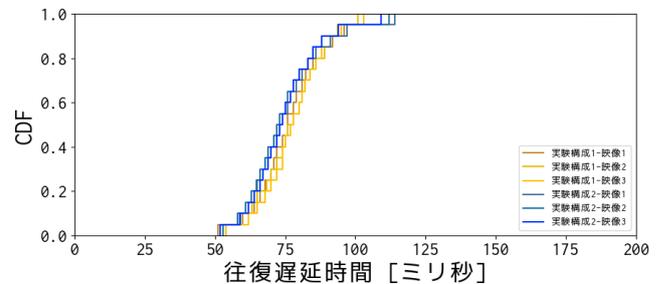


図 7 WebRTC: RTT の CDF

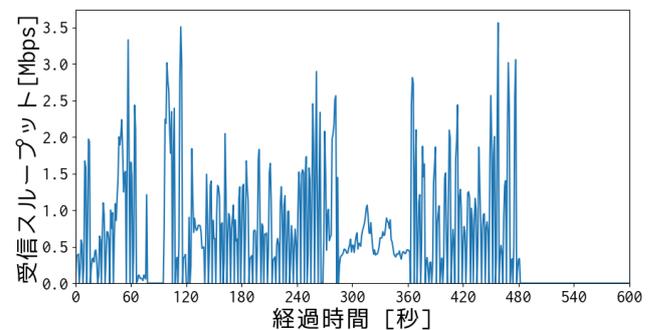


図 8 WebRTC: 受信スループットの推移 (実験構成 1)

程度である。一方で、RTT の計測失敗を表すロス率は実験構成 1 が実験構成 2 よりも高い値を示す。実験構成 2 ではマルチパス制御が回線切り替えを行う時に RTT が大きくなる傾向が見られる。例として表 1 の実験構成 2 の RTT 最大値 1361.0ms は、評価実験開始後 94 秒ごろにクライアント端末が path1 から path0 に通信回線を切り替える時に観測される。

WebRTC から見た通信の可用性・安定性の評価のため、リモート端末の Web ブラウザが観測した WebRTC の受信スループットの推移を図 8 (実験構成 1)、図 9 (実験構成 2) に示す。また受信スループットの CDF を図 10 に示す。

WebRTC の受信スループットは、実験構成 2 が実験構成 1 よりも高い値を示す。実験構成 1 では、断続的に受信スループットが 0 Mbps となり WebRTC の PeerConnection の切断および再接続が繰り返し行われる事象が発生しており、最終的に通信が切断されている。実験構成 2 では、受信スループットが 100Kbps から 8.5Mbps の幅で変動しているが、0 Mbps を示し通信が 1 秒以上にわたって途絶する事象は観測されていない。

映像伝送の安定性の評価として、リモート端末側の We-

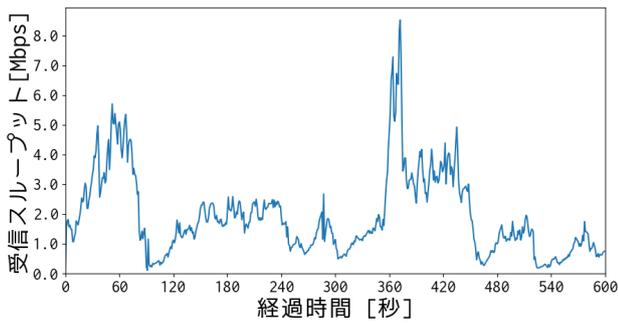


図 9 WebRTC: 受信スループットの推移 (実験構成 2)

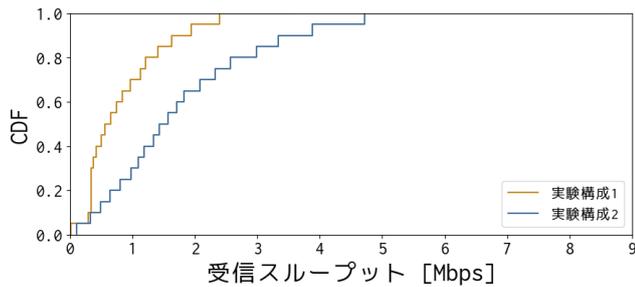


図 10 WebRTC: 受信スループット (CDF)

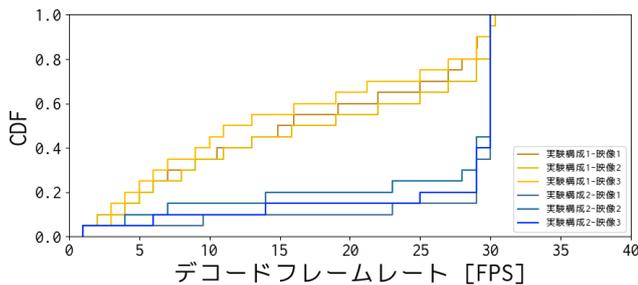


図 11 WebRTC: リモート端末でのデコードフレームレート (CDF)

WebRTC が 1 秒あたりにデコードした映像フレームの数 (デコードフレームレート) の CDF を図 11 に示す。リモート端末で映像を受信した際のデコードフレームレートは、クライアント端末の映像送出時のフレームレートと一致することが望ましい。しかしデコードフレームレートは、通信の遅延およびジッタにより変化し、映像のフリーズやフレーム飛びを発生させる。この値の変化が抑制されている場合、通信および映像伝送がより安定していると考えられる。デコードフレームレートは、実験構成 2 が実験構成 1 よりも高い値を示している。実験構成 2 では 50% 以上の期間で 30 FPS 以上を示している。一方で、実験構成 1 では 50% 程度の期間において 15 FPS 以下を示している。

映像品質の評価として、リモート端末側の WebRTC がデコードし受信した映像の画幅 (ピクセル) の分布を図 12 に示す。クライアント端末は 360p (画幅 640 ピクセル) の映像を伝送しているため、リモート端末において同サイズの映像が表示されることが望ましい。Google Chrome の WebRTC は Google Congestion Control (GCC) を用いて

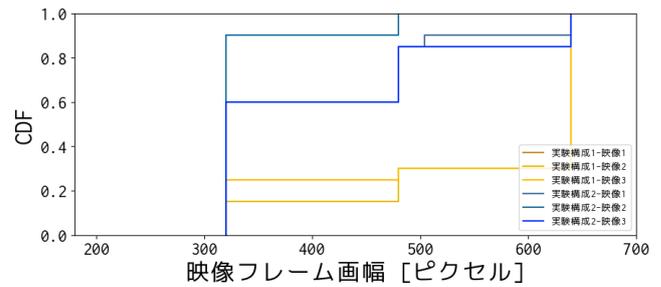


図 12 WebRTC: リモート端末が受信した映像の画幅 (CDF)

遅延・ジッタ・可用帯域に基づいた輻輳制御と映像圧縮により画質の削減を行い利用通信帯域の抑制制御を行う。そのため、リモート端末側でデコードされた画像の画幅が元画像と同じ 640 ピクセルに近いほど、十分な帯域と安定性を提供していると考えられる。図 12 より画幅は、実験構成 1 が実験構成 2 よりも高い値を示す分布になっている。実験構成 1 は通信が行っていた 70% 以上の期間において、640 ピクセルの画幅を維持していた。一方で実験構成 2 は 640 ピクセルの画幅を維持していた期間は 20% 弱にとどまる。

6. 考察

本章では、第 5 章の評価で示した実験結果をもとに MP-QUIC プロキシ導入による通信品質および WebRTC 映像伝送品質への効用を考察する。また、先行研究で提案したマルチパス UDP プロキシと本研究で用いた MP-QUIC プロキシの比較を行う。

6.1 MP-QUIC の通信品質への影響

図 5 および図 6 から、MP-QUIC プロキシを利用することで WebRTC の映像伝送にマルチパス制御が行われていることがわかる。本研究の MP-QUIC ではマルチパス選択ポリシーとして Lowest RTT First スケジューラを用いる。このため、MP-QUIC は図 4 に示される最小 RTT の通信回線を優先的に用いて通信を行う。図 5 では、最小 RTT に従って評価実験中に主たる送信スループットが計測される通信回線が変化している。試験車の移動をうけて変化した各回線の特性に、MP-QUIC のスケジューラが追従し継続的な通信がなされていたと考えられる。また、一部の期間では複数の通信回線で同時に送信トラフィックが観測されている。Lowest RTT First スケジューラが選択した回線で輻輳が発生した場合に、別の回線を併用することで通信品質の劣化を抑制していたと考えられる。

MP-QUIC でより高品質な通信を実現するためには、RTT 計測手法およびスケジューラの改善が必要と考えられる。今回の実験で用いた mpquic-go は定期的な RTT 計測を導入しておらず、UDP パケットを送信している通信回線および、輻輳制御により再送パッケージが送信される通信回線のみを対象に計測を行う。このため図 4 に示される

ように、RTT 計測結果が一定の期間に変化しない事象が発生し、輻輳が発生しない限り代替通信回線の調査が行われない。通信の状況が時々刻々と変化する自動車走行環境では、通信に利用している回線だけでなく他回線の通信状況を把握した動的な通信回線選択が肝要である。一方で文献 [16] では、MPTCP において頻繁な通信回線の切り替えが通信品質および映像品質を低下させることが示唆されており、回線の選択と切り替えには十分な検討が必要である。また、回線品質計測のために QUIC Ping フレームなどの探査パケットを用いた観測は、通信コストを増加させる可能性があり注意が必要である。

筆者らの先行研究 [1] では、回線の通信品質劣化を回避するためにマルチパス UDP プロキシを用いた複数通信回線の冗長化を提案した。しかし、この手法では複数の回線を常時利用するため通信コストが通信回線数に比例して増加する。通信途絶の可能性を抑制しつつ通信コストの低減を図るためには、文献 [5] に言及されている選択的な通信回線の冗長利用や精度の高い可用帯域および輻輳予測に基づく通信回線の選択が可能なスケジューラが必要と考える。

6.2 MP-QUIC の WebRTC 映像伝送品質への影響

表 1 および図 7 により、MP-QUIC プロキシの有無による WebRTC から見た通信遅延は、実験構成 2 が実験構成 1 よりも大きい。RTT 増加の理由は、プロキシにおけるパケットの転送処理および MP-QUIC にて発生する再送制御に起因すると考えられる。また、メディアストリームごとの QUIC ストリーム弁別を行わなかったため、メディアストリーム同士の Head of Line ブロッキングが発生し遅延を生じた可能性がある。この問題については、先行研究 [1] にて行った IP ToS 値を用いたストリームの分類機能により改善できると考える。MP-QUIC が通信回線を切り替える時に、通信遅延が大きくなる事象が発生している。原因として、通信回線を切り替える際に品質の異なる通信回線を併用した結果、再送・順序制御が行われ遅延が発生する可能性や、中継端末の MP-QUIC プロキシが切り替えに追従せず通信品質が低下した従前の通信回線に送信し続けている可能性がある。通信回線切り替え時の一時的な通信回線冗長化や、MP-QUIC プロキシ間の利用回線の同期により改善できると考える。

図 8 および図 9 の比較から、実験環境 1 では受信スループットが 0 Mbps となり通信途絶する事象が頻繁に見られるのに対して、実験環境 2 では MP-QUIC のもつ再送・輻輳制御により通信途絶を抑制できている様子がわかる。表 1 のロス率も、実験構成 1 は実験構成 2 よりも高い値を示しており、通信途絶がより多く発生したことを示している。単一通信回線では走行時に基地局間ハンドオーバーや基地局との位置関係の変化により通信が不安定になるが、マルチパス通信では通信品質の悪化を検知した時に別の通信回

線に切り替えることで通信を安定化できたと考える。

一方で、図 12 に示されるように、実験構成 2 は実験構成 1 に比べて画幅の値が低い傾向を示している。また画幅 640 ピクセルの画像が 320 ピクセルへ圧縮された通信が観測されており、通信品質の劣化が発生していたと考えられる。実験構成 2 は実験構成 1 に比べ通信を継続できていた期間が長く、通信品質悪化時にも GCC の映像品質制御および MP-QUIC の通信制御により画質を落とすつつも映像伝送が継続できていたと考えられる。一方で、実験構成 1 では高い画質で通信を開始し映像品質制御が行われることなく WebRTC が切断を繰り返していたため、実験構成 2 よりも値が高い傾向となったと考えられる。評価に用いた 360p 30FPS の映像 3 本の伝送は、合計約 3.0Mbps の帯域を必要とする。評価実験中、図 9 に示されるようにこの値を下回る受信スループットをリモート端末側で観測している。受信スループットが低下した原因として、用いた通信回線で十分な可用帯域が得られない場合や、GCC によって映像の圧縮が行われる場合が挙げられる。通信回線の冗長利用による可用帯域の確保、通信品質の推定・予測に基づく投機的な通信、および GCC に影響を与えるパラメータである遅延・ジッタを Head of Line ブロッキング回避により抑制することが効果的であると考えられる。

図 11 におけるデコードフレームの比較からは、MP-QUIC の導入により映像の途絶や欠落を抑制できていることがわかる。デコードフレームレートは、遠隔運転システムがオペレータに表示する視覚情報の画像表示間隔に影響し、デコーダおよび再生システムの実装によっては、映像の停止、映像フレームの欠損を発生させオペレータの操作に影響を与える。実験構成 1 と比較して、実験構成 2 はデコードフレームレートの平均値を高く変動を狭く維持できている。GCC による映像圧縮の効果が含まれると考えられるが、単一通信回線のみを利用する場合と比較して MP-QUIC プロキシが安定した映像伝送に寄与していた可能性がある。

6.3 先行研究の手法との比較

筆者らの先行研究で提案したマルチパス UDP プロキシ方式はトランスポートプロトコルに UDP、本研究の MP-QUIC プロキシ方式は QUIC を用いてマルチパス制御を実現する。マルチパス UDP プロキシ方式は、回線の冗長利用・選択的利用のためのマルチパス制御メカニズムのみを提供し、輻輳制御は WebRTC に依存していた。MP-QUIC は、トランスポート層での輻輳制御・再送制御を備えており、Lowest RTT First または Round Robin スケジューラによる複数通信回線の選択的利用ポリシーを提供する。そのため MP-QUIC プロキシ方式では、QUIC トランスポート層が備える通信の輻輳制御と WebRTC が備えるメディアストリームと通信を連携させた輻輳制御とが重複する。表 1 に示す実験構成 2 の RTT 最大値および標準偏差は実験

構成1よりも高い値を示しており、これはQUICによる制御に起因すると考えられる。一方、評価で示された結果では、QUICによる輻輳制御・再送制御がなされたことで通信の可用性向上およびパケットロス率の低減ができています。安定的な映像伝送を必要とする遠隔運転システムでは、メディアストリームと通信の双方が連携した輻輳制御が必要である。先行研究および本研究で導入したマルチパスプロキシに相当するトランスポート層が、遠隔運転システムに用いられるメディアストリーム制御層に対して提供する機能の充分性の検証が必要である。

7. まとめと今後の課題

本研究ではWebRTCの映像伝送を対象としてマルチパス通信技術であるMP-QUICを導入する手法を提案した。提案手法により複数の通信回線を利用した通信を可能とし、単一回線の場合と比べて通信を安定的に継続できること、映像品質の劣化を抑制できることを示した。

本研究ではプロキシ方式を採用し、そのトランスポートプロトコルにMP-QUICを採用した。WebRTCのトラフィックはSRTP/SCTP over DTLS over UDPであり、そのさらに外側にQUICでカプセル化を行なっている。今後は、方式のプロキシとカプセル化のオーバーヘッドをより詳細に分析する必要があると考える。

謝辞 本研究はウーブン・プラネット・ホールディングス株式会社との共同研究により実施されたものである。

参考文献

- [1] 金子直矢, 伊東孝紘, 渡辺敏暢, 阿部 博, 大西亮吉: マルチパス通信技術を活用したWebRTCにおけるメディアストリームトラフィックの柔軟な制御の実現, マルチメディア, 分散協調とモバイルシンポジウム 2021 論文集 (2021).
- [2] Iyengar, J. and Thomson, M.: QUIC: A UDP-Based Multiplexed and Secure Transport, RFC 9000 (2021).
- [3] De Coninck, Q. and Bonaventure, O.: Multipath QUIC: Design and Evaluation, *Proceedings of the 13th International Conference on Emerging Networking Experiments and Technologies*, CoNEXT '17, New York, NY, USA, Association for Computing Machinery, p. 160–166 (online), DOI: 10.1145/3143361.3143370 (2017).
- [4] De Coninck, Q. and Olivier, B.: Multipath Extension for QUIC draft-deconinck-quic-multipath-07 (2017).
- [5] Mogensen, R. S., Markmoller, C., Madsen, T. K., Kolding, T., Pocovi, G. and Lauridsen, M.: Selective Redundant MP-QUIC for 5G Mission Critical Wireless Applications, *2019 IEEE 89th Vehicular Technology Conference (VTC2019-Spring)*, pp. 1–5 (online), DOI: 10.1109/VTCSpring.2019.8746482 (2019).
- [6] Zheng, Z., Ma, Y., Liu, Y., Yang, F., Li, Z., Zhang, Y., Zhang, J., Shi, W., Chen, W., Li, D., An, Q., Hong, H., Liu, H. H. and Zhang, M.: XLINK: QoE-Driven Multi-Path QUIC Transport in Large-Scale Video Services, *Proceedings of the 2021 ACM SIGCOMM 2021 Conference*, SIGCOMM '21, New York, NY, USA, Association for Computing Machinery, p. 418–432 (online),

- DOI: 10.1145/3452296.3472893 (2021).
- [7] Raida, V., Svoboda, P., Lerch, M. and Rupp, M.: Crowdsensed Performance Benchmarking of Mobile Networks, *IEEE Access*, Vol. 7, pp. 154899–154911 (online), DOI: 10.1109/ACCESS.2019.2949051 (2019).
 - [8] Ford, A., Raiciu, C., Handley, M. J., Bonaventure, O. and Paasch, C.: TCP Extensions for Multipath Operation with Multiple Addresses, RFC 8684 (2020).
 - [9] Amend, M., Bogenfeld, E., Cvjetkovic, M., Rakocevic, V., Pieska, M., Kassler, A. and Brunstrom, A.: A Framework for Multiaccess Support for Unreliable Internet Traffic using Multipath DCCP, *2019 IEEE 44th Conference on Local Computer Networks (LCN)*, pp. 316–323 (online), DOI: 10.1109/LCN44214.2019.8990746 (2019).
 - [10] Shailendra, S., Bhattacharjee, R. and Bose, S. K.: MPSCPT: A multipath variant of SCTP and its performance comparison with other multipath protocols, *2012 IEEE International Conference on Communication Systems (ICCS)*, pp. 280–284 (online), DOI: 10.1109/ICCS.2012.6406154 (2012).
 - [11] Viernickel, T., Froemmgen, A., Rizk, A., Koldehofe, B. and Steinmetz, R.: Multipath QUIC: A Deployable Multipath Transport Protocol, *2018 IEEE International Conference on Communications (ICC)*, pp. 1–7 (online), DOI: 10.1109/ICC.2018.8422951 (2018).
 - [12] Tommy Pauly, E. K. and Schinazi, D.: An Unreliable Datagram Extension to QUIC (2022).
 - [13] Singh, V., Alvestrand, H. and Boström, H.: Identifiers for WebRTC's Statistics API (2021).
 - [14] Kang, L., Zhao, W., Qi, B. and Banerjee, S.: Augmenting Self-Driving with Remote Control: Challenges and Directions, *Proceedings of the 19th International Workshop on Mobile Computing Systems & Applications*, HotMobile '18, New York, NY, USA, Association for Computing Machinery, p. 19–24 (online), DOI: 10.1145/3177102.3177104 (2018).
 - [15] Pantel, L. and Wolf, L. C.: On the Impact of Delay on Real-Time Multiplayer Games, *Proceedings of the 12th International Workshop on Network and Operating Systems Support for Digital Audio and Video*, NOSSDAV '02, New York, NY, USA, Association for Computing Machinery, p. 23–29 (online), DOI: 10.1145/507670.507674 (2002).
 - [16] Nagayama, S., Cavendish, D., Nobayashi, D. and Ikenaga, T.: Managing Path Switching in Multipath Video Streaming, *INTERNET 2019; The Eleventh International Conference on Evolving Internet*, pp. 42–47 (online), available from (<https://ci.nii.ac.jp/naid/120006800702/>) (2019).