

# Android 端末における無線および有線 LAN を考慮した通信制御の一検討

松野 瑛南<sup>1</sup> 山口 実靖<sup>2</sup> 神山 剛<sup>3</sup> 小口 正人<sup>1</sup>

**概要:** 近年, 無線 LAN 通信の高速化の技術が進歩している. しかし, スマートフォン市場の拡大に伴い, 無線端末の使用台数の増加に加え, ライブ配信の視聴などデータの通信量が増加している為に, 使用可能な帯域を広げても輻輳は発生してしまう. また, 無線 LAN 通信の高速化に伴い, 有線側で古い機器を使用している環境では, 無線側だけでなく有線部分がボトルネックになる事例がある. そこで本研究では, 様々な原因で発生する輻輳を回避するために, 先行研究の輻輳制御ミドルウェアを基に改良することを目標としている. 輻輳制御ミドルウェアの構築を実現するために, 現在使用している回線の混雑状況や通信状況を知る必要があり, 本稿では, 通信性能の異なる複数 Android 端末の同時通信時の TCP パラメータに着目したトラフィック解析を行う. 解析結果から, 評価指標の Fairness Index は通信性能の差による公平性を考慮していないことが確認できたため, 改良版 Fairness Index を提案, 評価した.

## A Study of Communication Control Considering Wireless and WLANs on Android

ENA MATSUNO<sup>1</sup> SANEYASU YAMAGUCHI<sup>2</sup> TAKESHI KAMIYAMA<sup>3</sup> MASATO OGUCHI<sup>1</sup>

### 1. はじめに

近年, スマートフォンの登場により, 無線通信環境で使用する端末の市場が急速に拡大している. 全世界のスマートフォン市場は, 2020 年の段階で約 12 億台, 2024 年には 14 億台越えと今後さらに増加していくと予想されている [1]. 高性能, 高機能の端末であるため, スマートフォン端末でライブ配信の視聴, オンライン会議や大容量のデータ送信など使用方法にも変化が見られた. さらに, 無線規格の進化により, より高性能になったスマートフォンが高頻度で更新され, 端末の種類も豊富になった.

また, 無線通信の高速化の技術も進歩している. 現在, 無線通信環境では, 5G を代表とした公衆無線通信と, 無線 LAN 通信の高速化が進んでいる. 無線 LAN 通信は WiFi6 など通信の高速化に向けて新しい規格が出ている. 公衆無

線通信では, 5G という第 5 世代移動通信システムがある. 高速大容量, 高信頼・低遅延通信, 多数同時接続の主に 3 つの軸がある. このような技術の進歩により, 快適に通信ができる環境になった.

一方, 有線環境も無線環境同様変化が見られた. 前提として, 有線環境は無線に比べ, 機器の交換頻度が少なく, 取り替えが困難である事例が発生している. 一方で, 無線通信側は, アクセスポイント (以下, AP) やスマートフォンなど簡単に交換できる. また, スマートフォンは毎年のように新しい機種が発売されており, 機種変更すると安く買える購入パッケージもあるため, 取り替えは比較的簡単である. このような有線環境の事例に加えて, 無線通信の高速化や機種の交換が進むと, 無線環境側では, 通信速度や使用可能帯域が広がっていくのに対し, 有線環境側では従来の通信速度の環境が引き続き使用される可能性が高い. 結果として, 従来はエンドツーエンドの通信の中で有線部分の方が圧倒的に速く, 無線通信が常にボトルネックになっていたのに対し, 近年では有線環境よりも無線環境の方が速くなり, 有線側のボトルネックの影響で輻輳が発

<sup>1</sup> お茶の水女子大学  
Ochanomizu University

<sup>2</sup> 工学院大学  
Kogakuin University

<sup>3</sup> 長崎大学  
Nagasaki University

生ずる複雑な環境が一般的になりつつある。

先行研究では、Android 端末を用いて無線 LAN の AP における ACK パケットの蓄積を回避し、端末が同一の AP に接続し通信するときの全体の通信速度と公平性の向上を可能にするツールを提案、実装した [2][3]。先行研究で開発したツールをスマートフォンとタブレットに導入し、実験を行った結果、公平性と全体通信速度の向上が確認できた [4]。実験で使用した端末の通信性能は均一である。この実験環境では、公平性を高く保ちつつ、合計通信速度も向上する制御を目標としていた。しかし現在は、先述したように端末の高性能、輻輳の発生原因の増加により、以前より複雑な環境に変化したため、以前と同じ目標での制御が適切かどうか確認する必要がある。有線通信と無線通信の性能バランスが大きく変化した事に加え、端末であるスマートフォンも急激に高性能化が進み、最新の端末と少し古い端末の間の通信性能差が大きくなった結果、以前のように端末の通信性能が均一ではなくなり、極めて不均一な性能を持つ端末の混在する環境が一般的になっている。

そこで、本研究では以上の環境変化が起きた場合、どのような制御を行うか、目指す目標は何かを検討する。そして、検討から得た目標を達成するために制御を行うことを目的としている。

本稿では、通信性能の異なる複数の Android 端末の同時通信を行い、また通信速度が異なる 2 つの有線環境を構築して、実験を行った。各端末の振舞、スループットなどを合計通信速度と公平性の観点から評価した。実験の結果から、先行研究の輻輳制御ミドルウェアで使用した制御手法とは異なる制御が必要であることがわかった。さらに、トレードオフの関係である評価指標に基づく評価実験から、公平性の指標である Fairness Index の問題点を見出し、その改善を目指すために、通信性能によって重みをつけた改良版 Fairness Index を提案、評価を行った。

## 2. 関連研究

### 2.1 カーネルモニタ

カーネルモニタとは、バックグラウンドで処理されるカーネル内部の処理を監視するためのツール (図 1) である [5]。このツールは、TCP ソースコードにモニタ関数を挿入しカーネルで再コンパイルすることで、TCP パラメータをモニタ可能にする。モニタできる値として、輻輳ウィンドウ、ソケットバッファのキュー長、重複 ACK、SACK 受信、タイムアウト検出など各種エラーイベントの発生タイミング等がある。

このツールを組み込み機器である Android 端末上で実装することで、TCP 通信におけるカーネル内部の処理をユーザ空間からリアルタイムにモニタや解析を行い、TCP パラメータの値の変化を記録することが可能となる。

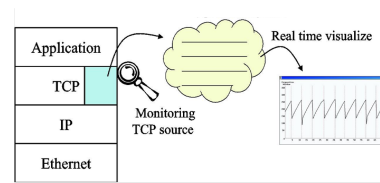


図 1: カーネルモニタ

### 2.2 スマートフォンにおける輻輳制御アルゴリズム

Android OS はカーネルに Linux カーネルを用いており、輻輳制御アルゴリズムに CUBIC TCP[6] を用いている。また、近年は新しい TCP 輻輳制御アルゴリズムとして TCP BBR[7] が提案され今後の普及が期待されている。ただし、CUBIC TCP と TCP BBR が共存して通信を行った場合は両 TCP のスループットの公平性が低くなることが示されており [8][9][10]、今後は複数種類の輻輳制御手法や端末が混在する環境における性能の公正性の考察が重要であると考えられる。特にモバイル環境における不公平性 [11] が示されており、スマートフォン端末などにおける性能公平性の改善が重要であると考えられる。

### 2.3 輻輳制御ミドルウェア

先行研究で開発された輻輳制御ミドルウェア [2][3] は、カーネルモニタをベースとしたシステムである (図 2)。同一の AP に接続した Android 端末間を連携し、各端末の接続状況を把握することで、通信速度の高速化と公平性の向上を可能にする。具体的には、各端末の通信状況や RTT を計測する。その情報から利用可能な帯域幅を計算し、帯域幅を上回るパケットを送信するのを防ぐために輻輳ウィンドウの上限値を自動で算出、補正する。これにより、端末間で可用帯域を公平に分け合い、無線 LAN の AP における ACK パケットの蓄積を回避する。

また、スマートフォン端末 (Nexus5) とタブレット端末 (Nexus7) にナイーブに適用し、その性能向上の程度を示す実験を行った。その結果に基づき、既存手法 [2][3] に改善を加えスマートフォンとタブレット端末の両方においてさらなる性能向上を示す手法を提案し、評価した。

本稿では、以下 2 つの環境変化に着目し、本実験環境において調査を行う。1 つ目は、端末がより高性能になったことにより、ヘテロな環境が発生したことである。2 つ目は、有線側の設備交換は時間がかかることに加え、無線側の交換は容易なため性能向上が進み、無線側と有線側双方がボトルネックになり得る輻輳の発生原因が増えたことにより、有線側の影響で発生する輻輳の制御を考慮することである。

## 3. 基礎実験

### 3.1 実験概要

まず基礎実験として、本研究で用いる Android 端末

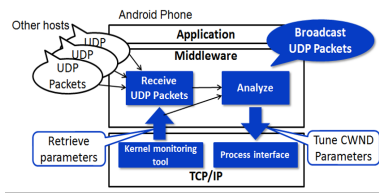


図 2: 輻輳制御ミドルウェア

(Pixel4, Pixel5, Nexus7)1台で iperf 通信 [12] を行い, 通信性能を評価した. また, スマートフォン端末 (Pixel4, Pixel5) とタブレット端末 (Nexus7) の計 4 台の Android 端末とサーバ間で iperf を用いた同時通信を行う.

### 3.2 実験環境

表 1 に実験に使用した機器の性能を, 図 3 に本研究で構築した実験環境を示す. Android 端末と AP 間は無線接続環境, AP とサーバ間は有線接続環境である. また, AP とサーバ間に転送速度 100Mbps のスイッチングハブを設置し, 1Gbps 環境と 100Mbps 環境の通信速度が異なる環境を構築した. 本研究では, クライアントとしてスマートフォン 2 台, タブレット 2 台, サーバとして Ubuntu Linux を OS とするデスクトップ機を実験に用いた. 本稿では, Pixel4 はスマートフォン 1, Pixel5 はスマートフォン 2, Nexus7 の 2 台のタブレット 1, 2 と記す.

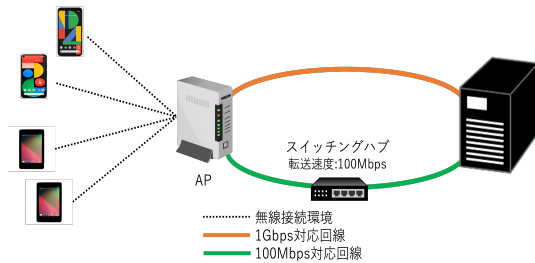


図 3: 実験環境

表 1: 実験機器の性能

Android	Model number	Pixel4	Pixel5	Nexus 7(2013)
	OS	Android 10	Android 11	Android6
	CPU	Snapdragon 855	Snapdragon 765G	Snapdragon S4 Pro
	Memory	6 GB	8 GB	2 GB
	WLAN	Wi-Fi 802.11 a/b/g/n/ac	Wi-Fi 802.11 a/b/g/n/ac	Wi-Fi 802.11 a/b/g/n
Server	OS	Ubuntu 20.04 LTS		
	CPU	Intel(R)Core(TM) i7-10400		
	Main Memory	32GB		
AP	Model	WSR-2533DHP2		
	Support Format	IEEE 802.11a/b/g/n/ac		
	Frequency Band	5 GHz		

### 3.3 実験結果

図 4,5 は, 各端末 1 台ずつの平均スループットを測定した結果を示し, 有線環境はそれぞれ 100Mbps と 1Gbps である. スマートフォン 2 台とタブレットの平均スループットの値が大幅に異なる理由の 1 つに無線規格の違いが挙げられる. タブレットの無線規格は 802.11n で, 最大通信速度は 600Mbps である. スマートフォン 2 台の無線規格は 802.11ac で, 802.11n と比較し, 帯域幅拡大, 変調信号の多値化, MIMO 方式の拡張に伴い, 最大通信速度は 6.9Gbps である.

図 6,7 は, 端末 4 台で実験を行った際の平均スループットを測定した結果を示し, 有線環境はそれぞれ 100Mbps と 1Gbps である. 100Mbps 環境, 1Gbps 環境ともにスマートフォン 2 台のスループット値が高く, タブレット 2 台は低い値となった. 1Gbps 環境のスマートフォン 2 台とタブレット 2 台のスループットの差は, 100Mbps 環境の差よりも大きい.

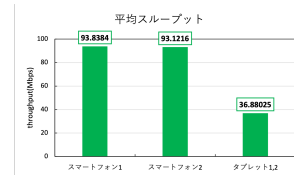


図 4: 有線環境 100Mbps (各端末 1 台)

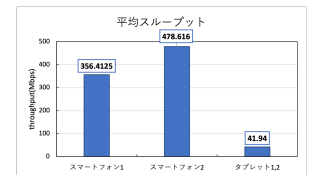


図 5: 有線環境 1Gbps (各端末 1 台)

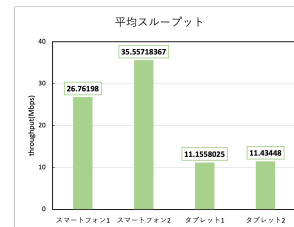


図 6: 有線環境 100Mbps (4 台同時通信)

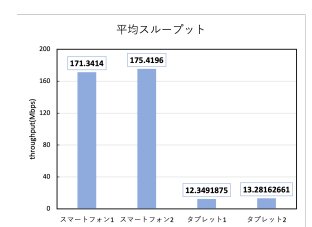


図 7: 有線環境 1Gbps (4 台同時通信)

## 4. 輻輳制御ミドルウェアの性能評価実験

### 4.1 実験概要

続いて, 先行研究によって開発された輻輳制御ミドルウェアによる性能評価を行うため, スマートフォン端末及びタブレット端末にミドルウェアを導入した. 前章と同様の実験環境下において, 公平性と合計通信速度の評価指標を用いてそれぞれの端末の振舞いを検証, 評価した.

公平性を評価する評価指標として, Jain の Fairness Index [13] が主に使用されている. 次式で定義されており,  $x$  は取得したスループット,  $n$  は標本数である.

$$F_j(X_n) = \frac{(\sum_{i=1}^n x_i)^2}{n \sum_{i=1}^n x_i^2}$$

さらに、輻輳制御ミドルウェアの制御の効果を解析するため、カーネルモニタから取得した TCP パラメータを取得し各端末の動作を解析、比較を行った。本実験で比較した TCP パラメータは、エラー状態 (ca-state) と輻輳ウィンドウ (cwnd) である。実験で使ったカーネルモニタのエラーコードを表 2 に示した。

表 2: カーネルモニタのエラーコード

番号	発生条件
1 番	パケット順序の入れ替わり検出
2 番	送信側のバッファが溢れることによるエラーの検出
3 番	重複 ACK, SACK などの受信
4 番	タイムアウトによる検出

## 4.2 実験結果

### 4.2.1 100Mbps 環境

有線環境が 100Mbps での平均スループットを図 8 に示す。図 9 は、通信速度と公平性を散布図で評価した。縦軸が公平性 (Fairness Index), 横軸が合計通信速度 (Mbps) を示す。また、色によって、デフォルト通信と、先行研究のミドルウェアにより制御した場合を区別する。

実験結果から、デフォルト通信では、図 4 の通信性能の高い順にスループットの値が高くなる結果となった。一方で、先行研究の輻輳制御ミドルウェアにより制御を行った場合は、全端末の平均スループットの値の差が縮まった。図 9 ではデフォルト通信に比べ、先行研究のミドルウェアがグラフの右上に分布している。これは、先行研究の輻輳制御ミドルウェアが合計通信速度、Fairness Index による公平性ともに向上していることが確認できた。先行研究の輻輳制御ミドルウェアは、合計通信速度と公平性の両方の向上を目標としているため、100Mbps 環境では輻輳制御ミドルウェアの制御が適切に動作していることが確認できた。

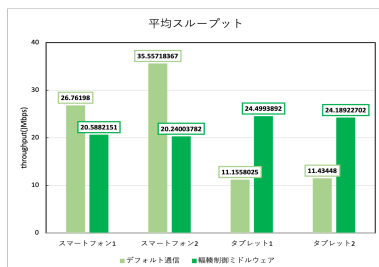


図 8: 有線環境 100Mbps の平均スループット

カーネルモニタを用いてさらに解析を行う。図 10,12,14,16 は、デフォルト通信の際の各端末の輻輳ウィンドウ (cwnd) とエラー状態 (ca-state) を示している。図 11,13,15,17 は、先行研究の輻輳制御ミドルウェアを導入した際の各端末の輻輳ウィンドウ (cwnd) とエラー状態

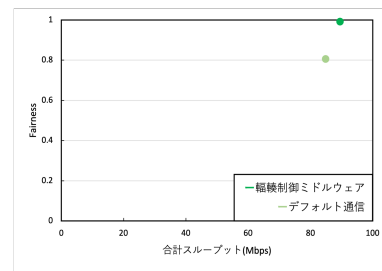


図 9: 有線環境 100Mbps の公平性と合計通信速度の評価

(ca-state) を示している。縦軸の左側はエラー状態、右側は輻輳ウィンドウ、横軸は時間を示す。

デフォルト通信と輻輳制御ミドルウェアを導入した場合の振る舞いを比較すると、全端末輻輳ウィンドウの値が大幅に上がり、エラーが大量発生していることがわかる。これにより、輻輳制御ミドルウェアが輻輳ウィンドウの値を最適化することで、結果として合計通信速度の向上に繋がったが、エラーも頻繁に発生することがわかった。

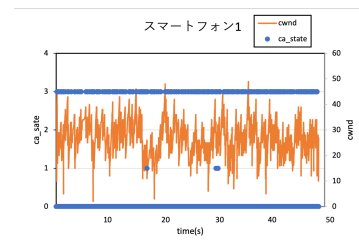


図 10: スマートフォン 1 (デフォルト通信)

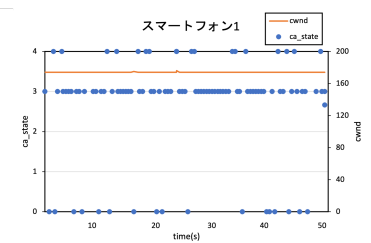


図 11: スマートフォン 1 (輻輳制御ミドルウェア)

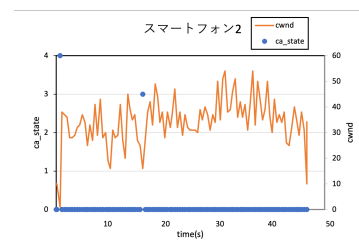


図 12: スマートフォン 2 (デフォルト通信)

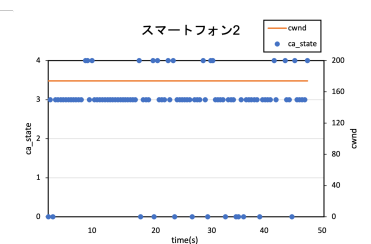


図 13: スマートフォン 2 (輻輳制御ミドルウェア)

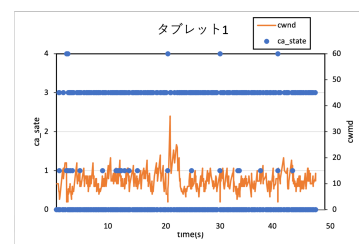


図 14: タブレット 1 (デフォルト通信)

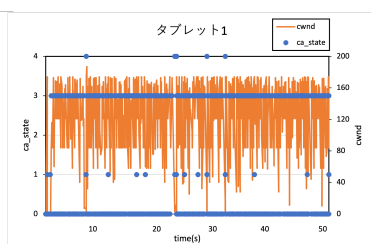


図 15: タブレット 1 (輻輳制御ミドルウェア)

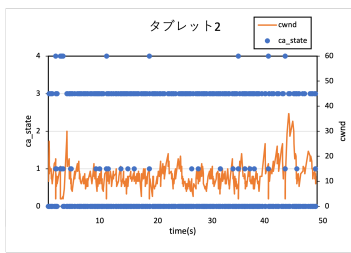


図 16: タブレット 2  
(デフォルト通信)

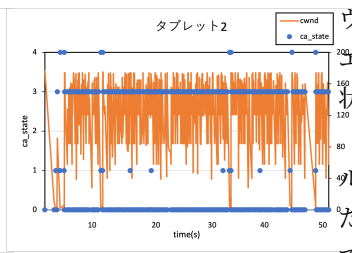


図 17: タブレット 2  
(輻輳制御ミドルウェア)

ウェアを導入した際の各端末の輻輳ウィンドウ (cwnd) とエラー状態 (ca-state) を示している。縦軸の左側はエラー状態、右側は輻輳ウィンドウ、横軸は時間を示す。

100Mbps 環境と比較し、デフォルト通信と輻輳制御ミドルウェアを導入した場合のグラフにほぼ変化がない。そのため輻輳制御ミドルウェアの制御に効果がなく、結果としてスループットも変化しなかった。

#### 4.2.2 1Gbps 環境

有線環境が 1Gbps での平均スループットを図 18 に示す。本実験結果でも、通信速度と公平性の両方に着目するために、散布図で実験結果を評価する (図 19)。縦軸が公平性 (Fairness Index)、横軸が合計通信速度 (Mbps) を示す。また、色によって、デフォルト通信と、先行研究のミドルウェアにより制御した場合を区別する。

図 19 から、デフォルト通信、先行研究の輻輳制御ミドルウェアともにほぼ同じ位置に分布している。これは、先行研究の輻輳制御ミドルウェアとデフォルト通信が合計通信速度、Fairness Index による公平性がほぼ変わらない結果であること示す。先行研究の輻輳制御ミドルウェアは、通信速度と公平性の両方の向上を目指し制御を行っているが、1Gbps 環境下では輻輳制御の効果がなく、期待通りに動作してない結果となった。

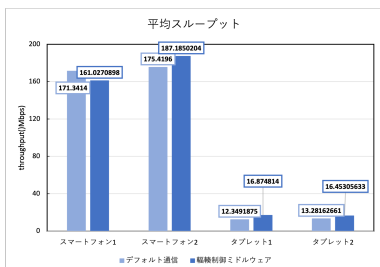


図 18: 有線環境 1Gbps の平均スループット

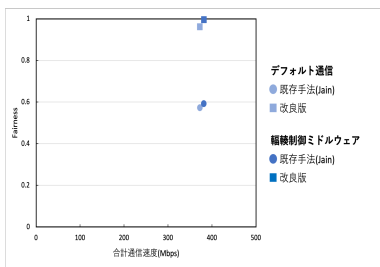


図 19: 有線環境 1Gbps の公平性と合計通信速度の評価

100Mbps 環境同様に、カーネルモニタから取得した値で解析を行う。図 20,22,24,26 は、デフォルト通信の際の各端末の輻輳ウィンドウ (cwnd) とエラー状態 (ca-state) を示している。図 21,23,25,27 は、先行研究の輻輳制御ミドル

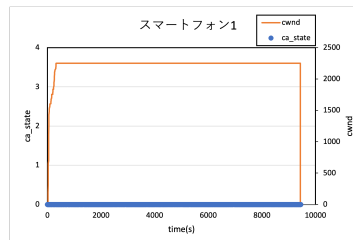


図 20: スマートフォン 1  
(デフォルト通信)

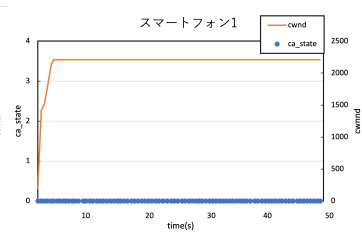


図 21: スマートフォン 1  
(輻輳制御ミドルウェア)

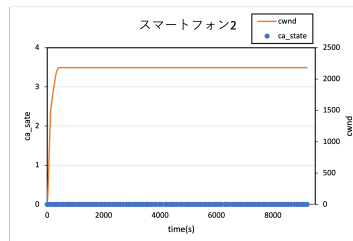


図 22: スマートフォン 2  
(デフォルト通信)

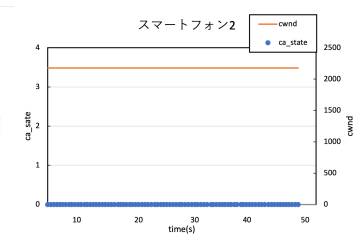


図 23: スマートフォン 2  
(輻輳制御ミドルウェア)

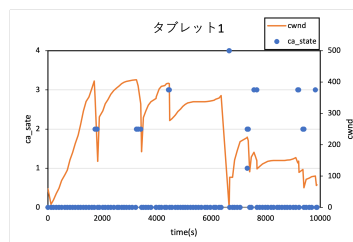


図 24: タブレット 1  
(デフォルト通信)

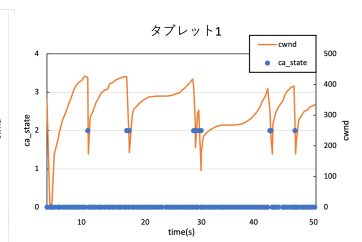


図 25: タブレット 1  
(輻輳制御ミドルウェア)

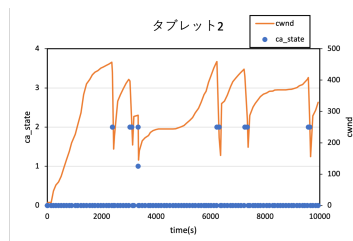


図 26: タブレット 2  
(デフォルト通信)

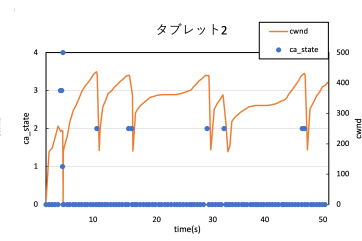


図 27: タブレット 2  
(輻輳制御ミドルウェア)

## 5. 評価指標に基づく評価実験

先行研究の輻輳制御ミドルウェアのツールを構築する上での目標は、公平性を高く保ちつつ、全体の通信速度を向上することであった。この二つの指標は、どちらも最高値を目指す最適化を行った場合には、一つの評価指標が向上すると、もう一つの評価指標の値が下がってしまうトレードオフの関係である。本研究も先行研究と同じく、二つの評価指標が高くなるにはどのように制御すべきか、評価を行った。4. 章から、100Mbps 環境は2つの評価指標の観点からうまく制御出来たため、本章でも有線環境が1Gbps 環境での評価実験を行う。実験環境は、3. 章と同様である。

### 5.1 合計通信速度重視

本研究の実験環境で合計通信速度を高めるためには、通信性能の高いスマートフォン2台のスループットを高めるよう制御を行う。これは、取得したスループット値(図18)と最大スループット値(図5)の差が、タブレットよりスマートフォンの方が大きいことから、制御を行うと合計通信速度が高くなると予測した。

TCP チューニングを行い、タブレット2台の輻輳ウィンドウの値を抑える制御を行うことにより、前章と同様の実験を行った。具体的には、タブレット2台の輻輳ウィンドウ値を10, 50, 100と三段階で実験を行った。本実験の有線環境は、1Gbps 環境である。図28は、TCP チューニングを行った際の実験結果の各端末の平均スループットの値を示した。図29は、図9と同様に通信速度と公平性を散布図で評価した。縦軸が公平性(Fairness Index)、横軸が合計通信速度(Mbps)を示す。また、青い点はミドルウェアを用いずにデフォルト通信を行った結果、オレンジの点は先行研究のミドルウェアを用いた結果、緑の点はTCP チューニング変更を行った結果を示す。またマークによって、輻輳ウィンドウ値10, 50, 100を区別する。

図29から、合計通信速度を重視しTCP チューニングを行った結果、デフォルト通信と先行研究の輻輳制御ミドルウェアに比べ、合計通信速度は向上しているが、公平性が著しく下がっていた。このことから、タブレット2台の輻輳ウィンドウ値を下げ、平均スループットを減少させることで、合計通信速度が向上することが確認できた。しかし、公平性の面でFairness indexの値が低く、あまり良い制御とはいえない。輻輳ウィンドウの値を下げるとFairness indexは0.5に近づいている。これはスマートフォン2台とタブレット2台の値が極端に異なることを示す。

### 5.2 公平性重視

図18から、スマートフォン2台のスループット値を下げ、タブレット2台の値を上げるよう制御を行うと公平性は高くなると推測した。

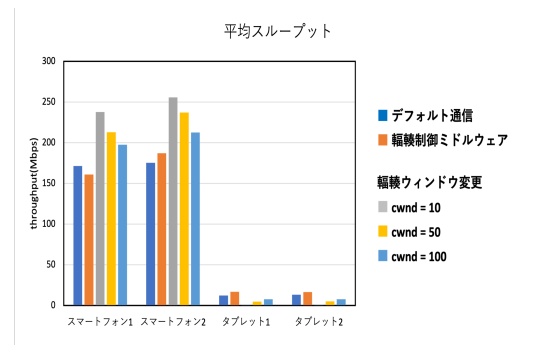


図 28: 合計通信速度を重視した制御の平均スループット

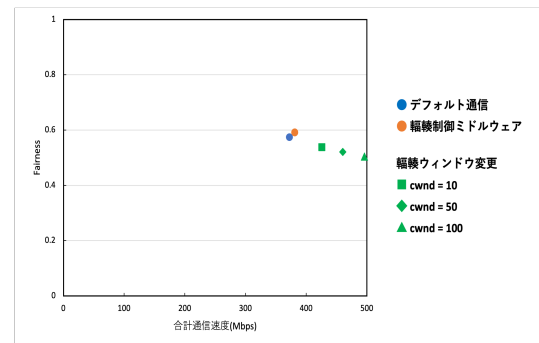


図 29: 合計通信速度を重視した公平性と合計通信速度の散布図

実際に上記のような制御をTCP チューニングで環境を構築し、実験を行った。具体的には、スマートフォン2台の輻輳ウィンドウの値を下げると、タブレット2台が取れる帯域が広がり、公平性が高くなる制御を行った。スマートフォン2台の輻輳ウィンドウ値を100, 500, 1000, 1500と四段階に設定して実験を行った。前節同様、本実験の有線環境は、1Gbps 環境である。実験結果は、図30に示した。図31は、図9と同様に通信速度と公平性を散布図で評価した。色によって、デフォルト通信、先行研究のミドルウェア、輻輳ウィンドウ変更を区別する。またマークによって、輻輳ウィンドウ値100, 500, 1000, 1500を区別する。

いずれの結果も、デフォルト通信や先行研究のミドルウェアはグラフの右下に分布しているのに対し、輻輳ウィンドウ値を変更した場合は、徐々に左上に分布し、公平性は向上したが合計通信速度が減少した結果となった。このことから、公平性を重視した制御は、スマートフォン2台の輻輳ウィンドウの値を下げていくとスマートフォン2台の平均スループット値が減少し、結果として公平性の向上が確認できた。しかし、使用可能な帯域をほぼ使わない制御手法となり、前節同様より良い制御を行うためには、合計通信速度も考慮する必要があることがわかった。

## 6. 考察

これらの実験から、先行研究の実験環境でうまく制御す

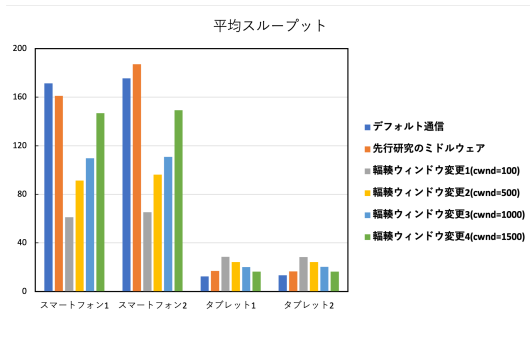


図 30: 公平性を重視した制御の平均スループット

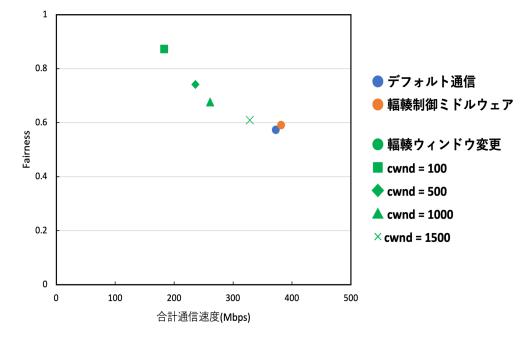


図 31: 公平性を重視した公平性と合計通信速度の散布図

る手法と本実験環境で最適に制御する手法は異なることが示唆された。異なる要素の一因として、端末の通信性能に大きな差があることが挙げられる。既存手法は、各端末のスループットを揃えて公平性を高く保ちつつ、全体の通信速度が高くなるように制御していた。しかし、本実験環境では通信性能を考慮する必要があり、既存手法と同様の制御では効果が見られなかった。さらに、前節の実験結果から、公平性と合計通信速度それぞれのみを重視した制御手法は好ましくないことも確認できた。

そこで本研究では、評価指標のうち Fairness Index に着目した。この指標には、端末の通信性能に大きな差がある実験環境で発生する問題点が見つかった。問題点の詳細は 7 節に示す。よって、本研究では改良版の Fairness Index を提案し、改良版 Fairness Index と合計通信速度を評価指標とし、評価を行っていく。改良版 Fairness Index の値が高く、さらに合計通信速度も高くなるような制御を目標とする。目標を達成するために、制御手法を再構築するなどの改良が必要だと考えられる。

## 7. 既存 Fairness Index の改良

### 7.1 改良版 fairness Index の提案

既存の Fairness Index を本実験環境で使用する場合、1 つの問題が発生した。Fairness Index が端末数と取得したスループットから公平性を求めるという点である。この指標は、性能の違いに関わらず複数端末を同時通信した際に

各端末が同等のスループットを取得しているかどうかを評価する指標となっている。しかし本実験環境では、前述したように、通信性能が異なる端末 4 台をクライアントとし、実験を行っている。図 5 からスマートフォン 1 とスマートフォン 2 の平均スループットの差も大きい、スマートフォン 2 とタブレットの平均スループットの差はおよそ 10 倍である。この環境で同時通信をした場合、スマートフォン 2 とタブレットが同じスループット値を取れないのは明白である。つまり、既存の評価指標は、通信性能が異なる端末で同時通信を行うと、様々な制御を行っても公平性が向上しない問題が発生することが確認できた。

前節を踏まえて本実験のような環境にも適応する公平性の指標について、通信性能に重みをつけた改良版の Fairness Index を提案する。この指標は前述した Fairness Index の式に取得したスループット値をそのまま代入せず、取得したスループットを最大スループットの値で割った値を代入する手法である。具体的には、各端末の最大スループットに対して取得したスループットがどのくらいの割合なのかを算出し、その割合から公平性を導いていく。本研究では、各端末の最大スループットの値は図 4 と図 5 の値を使用した。Max を各端末の最大スループット値とした場合、改良版の Fairness Index は下記の式ようになる。

$$F_j(X_n) = \frac{\left(\sum_{i=1}^n \frac{x_i}{Max_i}\right)^2}{n \sum_{i=1}^n \frac{x_i}{Max_i}}$$

この改良版 Fairness Index は、通信性能が異なる端末の同時通信した時も重みの割合で比較するため、5 節で述べた二つの指標のうち片方重視した制御と比較し、公平性合計通信速度共により良い値になる。本研究では、この改変によって、有線環境の通信速度が 1Gbps の場合の公平性の問題を解決することを目指した。

## 7.2 改良版 Fairness Index の評価

### 7.2.1 100Mbps 環境

4. 章と同様の実験結果を用いて改良版 Fairness Index の評価を行った。本実験結果のうち、各端末の平均スループットは、図 8 と同様である。図 32 は、合計通信速度と公平性の散布図で実験結果を評価する。また、薄緑色の点はミドルウェアを用いずにデフォルト通信を行った結果、緑の点は先行研究のミドルウェアを用いた結果を示す。また、マークによって、Jain の Fairness Index, 改良版 Fairness Index を区別する。

100Mbps 環境は、先行研究の輻輳制御ミドルウェアを導入した場合、既存の Fairness Index が向上した。しかし、改良版の Fairness Index では値が輻輳制御ミドルウェアを導入した場合よりもデフォルト通信の方が値が高くなった。このことから、100Mbps 環境でも改良版の Fairness Index の値が高くなるように輻輳制御ミドルウェアの改良を行っ

ていきたい。

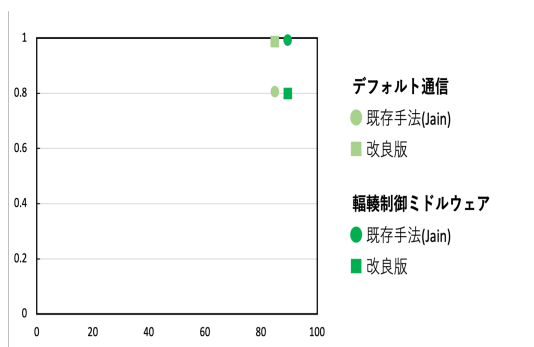


図 32: 改良版 Fairness Index の公平性と合計通信速度の散布図 (100Mbps 環境)

## 7.2.2 1Gbps 環境

4. 章と同様の実験結果を用いて改良版 Fairness Index の評価を行った。本実験結果のうち、各端末の平均スループットは、図 18 と同様である。通信速度と公平性の両方に着目するために、散布図で実験結果を評価する (図 33)。縦軸が公平性 (Fairness Index)、横軸が合計通信速度 (Mbps) を示す。また、水色の点はミドルウェアを用いずにデフォルト通信を行った結果、青い点は先行研究のミドルウェアを用いた結果を示す。また、マークによって、Jain の Fairness Index、改良版 Fairness Index を区別する。

まず、既存手法の Jain の Fairness Index のマークはグラフの中央に分布しており、デフォルト通信、先行研究の輻輳制御ミドルウェアともに公平性はほぼ変わらずあまり高くない結果となった。一方で、改良版 Fairness Index は、デフォルト通信、先行研究の輻輳制御ミドルウェアともにグラフの右上に分布している。これは、改良版 Fairness Index で通信性能を考慮した影響により、公平性が向上した。

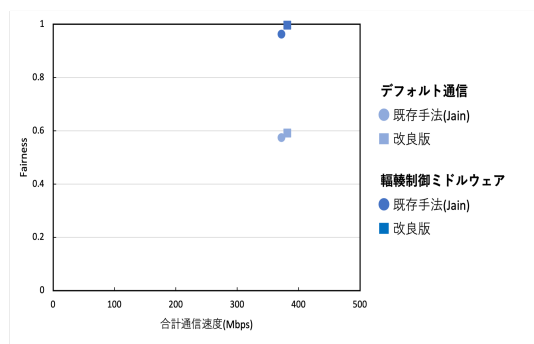


図 33: 改良版 Fairness Index の公平性と合計通信速度の散布図 (1Gbps 環境)

## 8. まとめと今後の課題

本稿では、通信性能の異なる複数の Android 端末の同時通信を行い、各端末の振舞、スループットなどを合計通信速

度と公平性の観点から評価した。これは、スマートフォン市場の成長と無線通信の高速化に伴い、通信環境に変化が現れたためである。実験結果から、既存の Fairness Index に問題が見つかり、改良版の Fairness Index を提案、評価を行った。

100Mbps 環境下では、輻輳制御ミドルウェアを導入した結果、既存の Fairness Index では高い公平性が得られ、合計通信速度も向上した。しかし、改良版 Fairness Index で評価を行うと、公平性が下がった。これは、輻輳制御ミドルウェアの制御により全端末のスループットをほぼ同じ値になったためである。

一方で、1Gbps 環境で輻輳制御ミドルウェアを導入した場合、Fairness Index、合計通信速度ともに制御していない場合とほぼ変化はなかった。改良版 Fairness Index で評価すると、既存の Fairness Index と比較して、値が高くなっていった。これは、通信性能の違いによって、取得した平均スループットの値は異なるが、端末ごとに最大スループット値に対する割合はほぼ変わらない。つまり、改良版 Fairness Index で定義した公平性が高いということが分かった。

今後の課題としては、公平性も保ちながら合計通信速度の向上も行う制御の構築を目指している。本稿では、有線環境が 1Gbps の場合改良版 Fairness Index の値が高くなったことは確認できたが、合計通信速度の影響はない。合計通信速度と公平性はトレードオフの関係なので、現在の制御が端末 4 台の限界点なのか、もしくは現在の公平性を保ちつつ、より高い合計通信速度を取得できる制御手法があるのか TCP チューニングなど実験を行っていきたい。

合計通信速度の向上する一つの案として、AP の制御が挙げられる。AP は詳細な範囲まで設定を変更できる。今後は、端末だけでなく AP による制御ではどのような影響があるのかも実験を行っていきたい。具体的には、バンドステアリング機能という AP が周波数帯を自動で決める設定がある。AP のデフォルトは、5GHz か 2.4GHz のどちらかの周波数帯を端末からユーザが選択できるようになっているが、この機能を使うと SSID が 1 つになり、ユーザ側からは選択できない設定になる。このような環境下では、周波数帯の幅が広がることによって合計通信速度が向上するのか、もしくは減少するのかを解析し、評価を行ってきたい。

## 参考文献

- [1] Worldwide smartphone shipment. <https://www.idc.com/promo/smartphonesmartphone-market-share/os>. Accessed: 2021-1.
- [2] Hiromi Hirai, Saneyasu Yamaguchi, and Masato Oguchi. A Proposal on Cooperative Transmission Control Middle-ware on a Smartphone in a WLAN Environment. Proc. IEEE WiMob2013, pp. 710-717, October 2013.



- [3] Ai Hayakawa, Saneyasu Yamaguchi, and Masato Oguchi. Reducing the TCP ACK Packet Backlog at the WLAN Access Point. Proc. ACM IMCOM2015, 5-4, January 2015.
- [4] Ayumi Shimada, Masato Oguchi. A Study of Android Tables Performance. Proc.DEIM2017,H2-3, March 2017.
- [5] Kaori Miki, Saneyasu Yamaguchi, and Masato Oguchi. Kernel Monitor of Transport Layer Developed for Android Working on Mobile Phone Terminals. Proc. ICN2011, pp.297-302 January 2011.
- [6] Sangtae Ha, Injong Rhee, and Lisong Xu. Cubic: A new tcp-friendly high-speed tcp variant, July 2008.
- [7] Bbr: Congestion-based congestion control. ACM Queue, 14(5):50, 2016.
- [8] M. Hock, R. Bless, and M. Zitterbart. Experimental evaluation of bbr congestion control. In *2017 IEEE 25th International Conference on Network Protocols (ICNP)*, pages 1–10, 2017.
- [9] K. Miyazawa, K. Sasaki, N. Oda, and S. Yamaguchi. Cycle and divergence of performance on tcp bbr. In *2018 IEEE 7th International Conference on Cloud Networking (CloudNet)*, pages 1–6, 2018.
- [10] K. Sasaki, M. Hanai, K. Miyazawa, A. Kobayashi, N. Oda, and S. Yamaguchi. Tcp fairness among modern tcp congestion control algorithms including tcp bbr. In *2018 IEEE 7th International Conference on Cloud Networking (CloudNet)*, pages 1–4, 2018.
- [11] Jae Won Li, Fengand Chung, Xiaoxiao Jiang, and Mark Claypool. Tcp cubic versus bbr on the highway. In Robert Beverly, Georgios Smaragdakis, and Anja Feldmann, editors, *Passive and Active Measurement*, pages 269–280, Cham, 2018. Springer International Publishing.
- [12] iperf. <https://iperf.fr/iperf-download.php>. Accessed: 2021-1.
- [13] D.-M. Chiu and R. Jain. Analysis of the increase and decrease algorithms for congestion avoidance in computer networks. *Computer Networks and ISDN Systems*, vol. 17, pp. 1–14, 1989.