

お茶の水女子大学大学院 人間文化創成科学研究科
博士前期課程 修士論文

無線LAN環境における帯域公平性の検討と QoS保証TCPの実環境における性能評価



著者氏名：理学専攻 情報科学コース 2年 新井 絵美

指導教官：人間文化創成科学研究科 教授 小口 正人

平成 21 年 3 月

要旨

近年, マルチメディア通信の需要の高まりにより, QoS (Quality of Service) の保証が重要となっている. ここで定義する QoS とは指定された帯域を確保することである. これを実現する TCP-AV という帯域確保 TCP がこれまでに提案, 実装され, 有線環境における評価がすでに行われてきた. しかし, 無線環境においては有線環境における場合とは異なる問題が存在する. たとえば複数台の端末で通信を行ったときの送信権の制御メカニズムなどが原因となって, 端末ごとのスループットに不公平が生じる場合があるといわれており, その結果有効な帯域制御が行えない可能性がある. そこで本研究では, 実機を用いて公平性に関する実験および考察を行い, 無線環境において TCP-AV を用いてマルチメディア通信を実行する際の性能を評価し, 問題点を改善することによって, より良い QoS 制御を行うことを目指す.

キーワード : TCP, 無線 LAN, QoS, 公平性

abstract

In late years, the demand for multimedia communication has been raised extensively. To guarantee the QoS is extremely important. QoS defined here is to guarantee target bandwidth. In order to realize it, TCP-AV has been proposed and implemented, and it is achieved in a wired environment. However, wireless environment has problems different from the wired environment. When multiple terminals are communicating, it is said that unfairness of the throughput among terminals is observed caused by control mechanism of the data transmission right. As a result, there is a possibility that it cannot control the bandwidth effectively. Therefore we invested and considered about fairness in this paper. We have used TCP-AV on wireless environment and evaluated performance of multimedia communication, for aiming at better QoS controls by resolving problems.

Kerwords : TCP,wirelessLAN,QoS,fairness

目次

第 1 章	序論	1
1.1	研究背景	1
1.2	研究目的	2
1.3	論文構成	3
第 2 章	無線 LAN 環境における公平性	4
2.1	無線環境における公平性	4
2.1.1	CSMA/CA による送信機会の平等と自律分散制御	4
2.1.2	CSMA/CA における TCP の振舞の特徴	6
2.1.3	MAC 層の実装の違い	8
2.2	トランスポート層プロトコル	9
2.2.1	UDP	9
2.2.2	TCP-Reno	10
2.2.3	TCP-AV	11
第 3 章	公平性に関する実機評価	13
3.1	同機種無線 LAN 環境における TCP フローの公平性	13
3.1.1	実験概要	13
3.1.2	実験結果と考察	14
3.2	不公平の傾向についての検証	15
3.2.1	基礎実験（前提条件）	16
3.2.2	マシンのスペックや OS	18
3.2.3	端末の組み合わせによる傾向	19
3.3	原因の検証	20

3.3.1	AP のバッファサイズの検証	21
3.3.2	UDP を用いた環境	22
3.3.3	一般的な環境における TCP を用いた環境	25
3.4	異機種無線 LAN 混在環境における TCP フローの公平性	26
3.4.1	基礎実験	26
3.4.2	UDP	27
3.4.3	TCP フロー	29
第 4 章	TCP-AV の無線環境における評価	33
4.1	TCP-AV の無線 LAN 環境における実装	33
4.2	端末端末が 3 台の場合	34
4.2.1	実験概要	34
4.2.2	実験結果および考察	35
4.3	送信端末が 5 台の場合	36
4.3.1	実験概要	36
4.3.2	実験結果および考察	36
第 5 章	まとめと今後の課題	38
5.1	まとめ	38
5.2	今後の課題	39
	謝辞	40
	参考文献	42
	発表文献	44

目 次

2.1	公平および不公平なスループット	5
2.2	シミュレーションにおける TCP フローの不公平	7
2.3	シミュレーションに環境におけるスループット	8
2.4	実機環境における TCP フローの不公平	9
2.5	実機環境におけるスループット	10
2.6	無線子機と MAC 層の違い	11
2.7	輻輳ウィンドウの状態変化	12
3.1	同機種無線 LAN による実験環境	14
3.2	TCP フローが公平な場合 (送信端末数 3)	14
3.3	TCP フローが公平な場合 (送信端末数 4)	15
3.4	TCP フローが不公平な場合 (送信端末数 5)	15
3.5	TCP フローが不公平な場合 (送信端末数 6)	16
3.6	TCP フローが不公平な場合 (送信端末数 7)	16
3.7	端末の故障がないことを確かめる実験環境	17
3.8	端末の故障がないことを確かめる実験	17
3.9	無線 LAN クライアントの故障がないことを確かめる実験環境	17
3.10	無線 LAN クライアントの故障がないことを確かめる実験	18
3.11	無線 LAN クライアントの配置	18
3.12	マシンのスペックや OS の違いに関する実験環境	19
3.13	マシンのスペックや OS の違いに関する実験 1	19
3.14	マシンのスペックや OS の違いに関する実験 2	20
3.15	5 台の組み合わせを変えた実験 1 (Windows 3 台, Linux2.4 1 台, Linux2.6 1 台)	20
3.16	5 台の組み合わせを変えた実験 2 (Windows 2 台, Linu2.4 1 台, Linux2.6 1 台)	21

3.17 5 台の組み合わせを変えた実験 3(Windows 3 台,Linux2.6 2 台)	21
3.18 5 台の組み合わせを変えた実験 4(Windows 4 台,Linux2.6 1 台)	22
3.19 5 台の組み合わせを変えた実験 5(Windows 4 台,Linux2.4 1 台)	22
3.20 AP のバッファサイズ検証実験	23
3.21 UDP を用いた際の基本性能	23
3.22 同機種無線 LAN 環境 (UDP)	25
3.23 同機種無線 LAN 環境のスループット (UDP)	25
3.24 一般的な無線環境における TCP フローを流す実験	26
3.25 一般的な無線環境における TCP フローのスループット	26
3.26 USB タイプ無線 LAN のみを用いた実験環境	27
3.27 USB タイプ無線 LAN のみを用いた環境のスループット	27
3.28 異機種無線 LAN 端末混在環境を用いた実験環境 (UDP)	28
3.29 異機種無線 LAN 端末混在環境のスループット (UDP)	28
3.30 異機種無線 LAN 端末混在環境において TCP フローを流す実験	29
3.31 異機種無線 LAN 端末混在環境における公平性 (遅延 2.5 ミリ秒)	29
3.32 異機種無線 LAN 端末混在環境における公平性 (遅延 20 ミリ秒, パケットロス 0.5%)	30
3.33 イーサネットコンバータの動作を解析する実験環境	30
3.34 TCP モニタの実行結果	31
3.35 tcpdump の実行結果	31
4.1 TCP-AV の無線 LAN 環境における実装	33
4.2 AVR を評価する実験環境	34
4.3 AVR による TCP-AV の評価 1 (送信端末数 3)	35
4.4 送信端末が増加した場合の AVR の評価	36
4.5 AVR による TCP-AV の評価 2 (送信端末台数 5)	37

第1章 序論

1.1 研究背景

近年、動画ストリームや音声（VoIP）などのマルチメディア通信の需要が高まっている。そのような通信において QoS（Quality of Service）は大変重要である。しかし一口に QoS と言ってもそこで要求される品質はメディアやアプリケーションにより異なる。そのため、マルチメディア通信のための「QoS」を定義し、これが保証される仕組みを作る必要がある。また、インターネット（TCP/IP に基づくネットワーク）の本質は「ベストエフォート」であるが、QoS が必要とされる場面が増えてきたため、インターネットのプロトコルに QoS 保証の仕組みを組込む方法が検討されてきている。この 1 つの例が TCP-AV[1] である。

TCP-AV は、ルータなどに QoS 制御を実装するのではなく、既存の TCP をこれに載せかえて通信を行うというものである。TCP-AV では、目標帯域を確保するように輻輳ウィンドウ制御パラメータを変更する。これにより、ストリーミング通信などを効果的に行う QoS を実現できる。TCP-AV により有線環境における帯域確保などの QoS 保証が達成されるようになった。

一方で、無線 LAN が広く普及したことから、無線環境においても有線環境と同様の通信の品質が保証されることが望まれている。その例は、前述の VoIP などである。ユーザは常に同じ品質でアプリケーションが利用可能であることを要求するが、システムとしては無線環境において、有線環境の場合と同様に品質保証を行うことは、より困難である。そこで本研究では TCP-AV のような帯域確保型 TCP が無線環境においても想定しているように振舞うかどうかを検証する。特に実機を用いて様々な評価を行うことにより、シミュレーションではモデル化が困難な端末ごとの機器の差異やバッファの大きさなどによる影響を明らかにし、実環境で有効な QoS 手法について検討する。

また、トランスポート層プロトコルの相違による影響に着目して評価を行った。例えば VoIP では、リアルタイム性を重視し再送信を行わない UDP を使用して音声パケットを送っている。このよ

うにストリーム通信などではUDPの使用が一般的となっている。しかし、UDPはファイアウォールなどを通過できない場合がある。ファイアウォールは基本的に許可したパケット以外を通さない方向で働くため、突然送られてくるUDPは落とされやすいといえる。このことから、UDPだけでなくTCPを用いたQoS手法を検討していく必要があると考えられる。

無線環境におけるQoS保証を実現するために、本論文では無線環境における帯域公平性に焦点をおく。この問題は端末の台数が増えたときに顕著になり [2], すでにシミュレーションにおいて、無線端末から送信される複数のTCPフロー間におけるスループットの不公平の発生が示されている [3]。これは、有線と無線のネットワーク速度の非対称性と無線LANのMAC制御における送信権の平等性により起こるといわれている。上記特性の下では、無線LAN AP (Access Point) のバッファにおけるオーバーフローによるTCP-ACKの破棄が不公平につながるといわれている。しかし、無線上のノイズや電波干渉、デバイスドライバの構築法やOSの構成など、シミュレーションでは考慮されない点が実機には存在するため、このような不公平が実機においてどの程度起こるのかどうかは明らかではない。例えば実機のAPのバッファサイズはメモリや装置の低価格化などの要因で変動していると考えられ、現状の実システムでバッファが問題になりうるかどうかを把握することは非常に重要である。上記実機における諸要因を考慮しても、なおTCP-ACKの破棄が原因となることを示すために、UDPを用いた実験を行い原因を明確化する必要があると考えられる。さらに、これらの環境とは逆向きの環境においても実験を行うことで、有線側のAPにおけるバッファあふれ、つまりACKの破棄に起因することを確認する。公平性の検証のためには、同機種種の無線LAN環境および異機種種の無線LAN混在環境においてそれぞれ公平性を検証する。これらの実験により実機の無線環境における課題を議論した上で、有線環境においては品質保証を実現することができるTCP-AVが実機の無線環境で期待通りの働きができるかについて検証を行った。

1.2 研究目的

本研究では、無線LANのQoS実現に必要な要素技術のうち以下の二点に焦点を当てた。

1点目は、無線LAN環境における公平性の検証である。具体的には、イーサネットコンバータタイプ無線LAN (11g) を用いてTCPフローを流した際に不公平が起こるかどうかを調べた。シ

ミュレーションやカードバスタイプの無線 LAN (11b) では不公平が示されているため、イーサネットコンバータタイプの無線 LAN を使用した。さらに、TCP-AV を無線 LAN 環境で使用する際にイーサネットコンバータを用いるのが便利である。そのため、このような環境で公平性を調査することは重要であると考えられる。

2 点目は、有線環境における QoS 保証 TCP の性能評価を行う。本研究では、QoS 保証 TCP として TCP-AV を用いる。この TCP-AV は有線環境において良好な振舞をした。しかし、無線 LAN 環境には不安定であり、不公平が起こりうる。そこで、無線 LAN 環境において TCP-AV の振舞を調査し、無線 LAN 環境における QoS 保証を考える。

1.3 論文構成

次章以降の構成は以下の通りである。

第2章 無線 LAN 環境における公平性

無線 LAN の特徴、トランスポート層プロトコルなどを紹介し、既存研究における公平性とその原因について紹介する。

第3章 公平性に関する実機評価

無線 LAN クライアントを統一した環境における公平性の検証およびその原因について考察を行う。次に、イーサネットコンバータと USB タイプ無線 LAN の混在環境における公平性の検証および原因について考察を行う。

第4章 TCP-AV の無線環境における評価

TCP-AV の無線環境における評価を行う。その際に独自の評価方法である AVR を用いる。

第5章 まとめと今後の課題

本研究のまとめを述べる。

第2章 無線LAN環境における公平性

2.1 無線環境における公平性

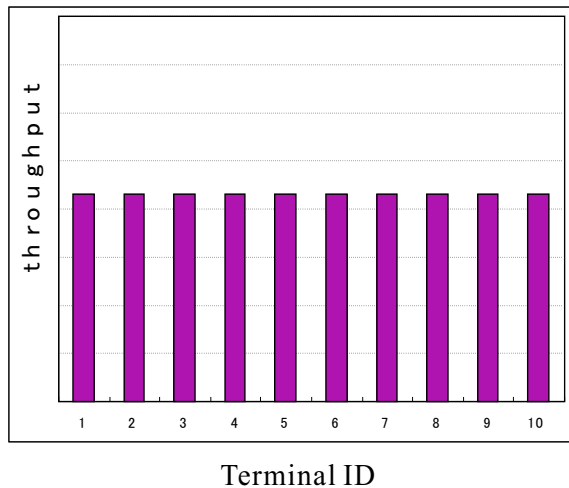
まず、無線LAN環境におけるTCPフローのスループットの公平と不公平の定義を示す。本研究における無線LAN環境におけるTCPフローのスループットの公平性の問題とは、同じ条件で通信しているにも関わらず端末間でスループットが不公平になることである。これはつまり複数の無線端末から同時に有線端末にデータを送信するときに、ある端末はほとんどスループットが上がらず（0Mbpsに近い）、ある端末は全帯域をすべての端末で均等に分けた値（fair-share）よりも高いスループットが出るという様に極端にするスループットの違いが現れる状態である。このときの一般的なスループットの特徴を表すグラフの例を図2.1に示す。横軸に端末番号、縦軸にスループットをとった。左のグラフがここで定義する公平な場合のグラフであり、右のグラフが不公平な場合を示している。

以下、無線LANの制御の特徴を述べ、この問題について考察する。

2.1.1 CSMA/CAによる送信機会の平等と自律分散制御

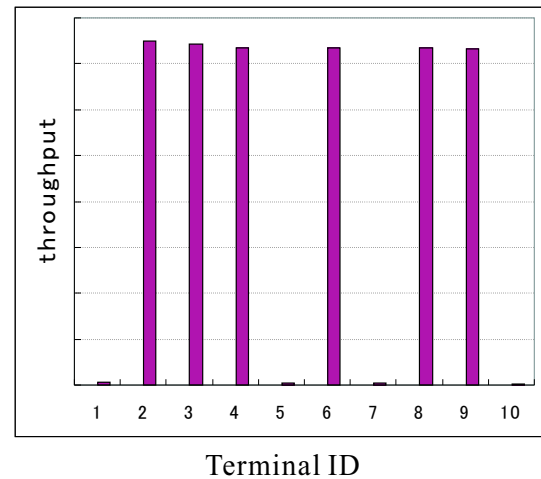
CSMA/CA（Carrier Sense Multiple Access with Collision Avoidance）は、IEEE802.11無線LANで標準的に適用されているアクセス方式である。この方式は、衝突を回避するために、他の無線端末が電波を出していればある一定期間（IFS: Inter Frame Space）だけ待ち、再び搬送波を調べて周囲が電波を出していなければランダム時間後に自分が送信するというものである。この方式の特徴はすべての無線端末が対等の関係にあり、しかも分散制御している点にある。

• faircase



throughput of all terminals
are almost equal

• Unfaircase



There are huge difference on throughput
among terminals

図 2.1: 公平および不公平なスループット

(1) 送信機会の平等

CSMA/CA の 1 つ目の特徴は送信機会の均等性である。バックオフ制御は、キャリアセンスに加えて衝突を回避するための方法として 802.11 規格に定められている。バックオフ制御では DIFS (Distributed Coordination Function IFS) 時間アイドルになった後、フレームを送信しようとする無線端末は、規定の CW (Contention Window) 範囲内で乱数を発生させ、そのランダム時間 (バックオフ時間) が決められる。キャリアセンスをランダム時間だけ行うことにより、各無線端末には公平な送信機会が与えられることになる。フレームが衝突した場合には、再送ごとにバックオフ制御の CW の範囲を 2 倍に増加させる 2 進数バックオフにより、フレームの再衝突する確率を低減させる。

(2) 自律分散制御

2 つ目の特徴は自律分散制御である。802.11 無線 LAN の基本アクセス手順とされている DCF (Distributed Coordination Function) では、各無線端末がチャネルの使用状況を検査して自律的

にフレームの送信タイミングを決定する。このときのアクセス制御プロトコルに CSMA/CA を使用している。

ただし CSMA/CA では偶然同時にパケットを送信してしまう可能性、すなわちパケット同士が無線上で衝突してしまう可能性がある。衝突によって、パケットが正常に受信されなかった場合、パケットは再送されるが、衝突の発生する確率は無線端末数と通信トラフィックの増加につれて大きくなり、スループットの低下を招く。このとき、端末ごとの MAC (Media Access Control) ドライバや TCP の実装に起因するリソース獲得の振舞の違いは考慮に入れられず、その結果送信権の平等性を目指しながら、不公平が起こってしまう。以下、その問題を議論する。

2.1.2 CSMA/CA における TCP の振舞の特徴

TCP において 1 セグメントごとに確認応答を行った場合、RTT (Round Trip Time) が大きくなったときにスループットが低下してしまう。そこで、確認応答を待たずに複数のセグメントを送信できるようにすることで性能を低下させないようにしている。このとき確認応答を待たずに送信できるデータの大きさをウィンドウサイズという。

(1) ウィンドウ制御による TCP-ACK ロスの影響

ウィンドウ制御をするときに確認応答 (ACK) が失われてしまった場合、データは届いているので再送する必要はない。そのため、ウィンドウサイズがある程度以上の大きさがある場合には少しぐらい確認応答が失われても再送処理は行わず、次の ACK パケットでデータの到着が確認される。例えば、ウィンドウサイズが 10 で通信している端末は 10 のうち 1 以上の ACK が返ってくれば、ウィンドウサイズを少し大きくして、さらに、そのまま通信を続けることができる。しかし、ウィンドウサイズが 1 で通信している端末は、その 1 つの ACK が返ってこなければデータは再送となり、ウィンドウサイズを増加させることができず、なかなかウィンドウサイズが大きな値にならない。単純に述べると、前者は、後者に比べて 10 倍有利に通信をすることができる。

このように同じ ACK のロスでもウィンドウサイズによって影響が異なる。すなわちウィンドウサイズが大きい端末は、ACK ロスの影響をほとんど受けることなしに、ウィンドウを増加させ、スループットを高くすることができるが、ウィンドウサイズが小さい端末はいつまでもデータの送信ができず、スループットをあげることができない。

(2) フローごとのスループットの公平性

ここで、シミュレーションで示されている不公平の原因についての議論を紹介する。シミュレーションにおいてはこの原因は AP のバッファにおける ACK のあふれであると述べられている [3]。これは端末の台数が増えたときに特に問題となる。例えば、10 台の無線端末から AP を経由して 1 台の有線端末にデータを送信する場合を考える（図 2.2 参照）。このとき、データの送信は高々 11Mbps であるのに対し、ACK は 100Mbps の速さで AP のバッファにたまる。AP は無線の端末に ACK を返したいが、AP も無線端末のひとつとして送信権を得なければならない。しかし、10 台分の ACK を $1/11$ の確率でしか送信できないため、有線リンクの速さとの相乗効果によって AP のバッファにどんどん ACK がたまっていく。その結果バッファに入りきらない ACK は破棄される。この ACK の破棄が上記のようにウィンドウサイズの違う端末に影響を与え、スループットの不公平を招く。図 2.3 にそのときのスループットを示した。

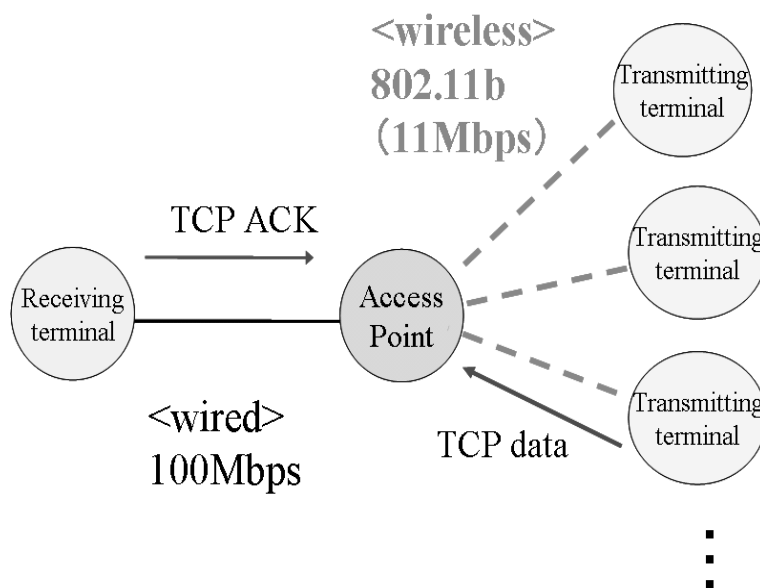


図 2.2: シミュレーションにおける TCP フローの不公平

また、実機環境において、図 2.4 のようなパラメータとカードバスタイプの無線 LAN を用いて同様の実験環境において実験を行った結果は図 2.5 のようになり、実機の環境においても不公平が示されている [4]。

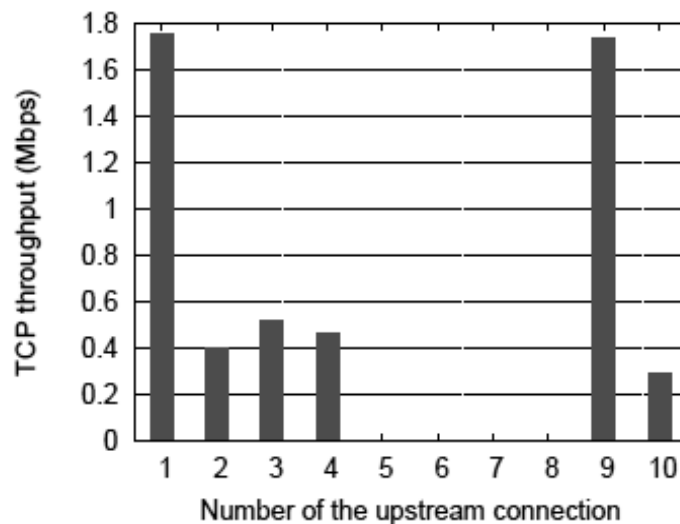


図 2.3: シミュレーションに環境におけるスループット

2.1.3 MAC 層の実装の違い

無線子機の MAC 層の違いはシミュレーションでは表せない影響を与え、実機特有の問題を引き起こす可能性がある。

データリンク層内の MAC 層の代表としては、有線の規格である IEEE802.3 の CSMA/CD (CSMA with Collision Detection)、無線の規格である IEEE802.11 の CSMA/CA の 2 種類がある。CSMA/CD においては、データを送信したい端末はケーブル上の搬送波を調べ通信状況を監視し (Carrier Sense)、ケーブルが空くとデータの送信を開始する。このとき、複数の端末が同時に送信を開始するとケーブル内でデータが衝突して壊れるため (Collision Detection)、両者は送信を中止し、ランダムな時間待機してから送信を再開する。この方法に従うと、1 本のケーブルを複数の端末が共有して、互いに通信する (Multiple Access) ことができる。

次に本研究で使用する無線子機を紹介する。本研究では無線子機として、USB タイプとイーサネットコンバータタイプの 2 種類を使用する。この 2 種類は MAC 層の実装が異なる。USB タイプは一般的なカードタイプの無線子機同様 MAC 層から直接無線となり、CSMA/CA の仕組みで動作している。一方、イーサネットコンバータタイプの無線子機は端末からイーサネットケーブルを通り、その先で無線に変換してから通信を行っている。そのため、端末からは CSMA/CD の仕組みで動作していると認識される。この違いを図 2.6 に示す。

Hardware		
1x AP	Dell GX 280	2.8Ghz P4
12x node	Soekris net4801	266Mhz 586
WLAN	D-Link DWL-G520	Atheros AR5212
Buffers	default	used
TCP	64KB	1MB
interface tx	199 packets	10 packets
driver tx	200 packets	10 packets



図 2.4: 実機環境における TCP フローの不公平

本稿では, USB タイプの無線子機として, BUFFALO WLI-UC-AG[5] を使用した. また, イーサネットコンバータタイプ無線子機として, BUFFALO WLI3-TX1-G54[6] と BUFFALO WLI-TX4-G54HP[7] を使用した.

2.2 トランスポート層プロトコル

無線 LAN 環境における公平性の問題について, トランスポート層プロトコルの相違という観点から議論する. そしてまず, トランスポート層プロトコルである UDP と TCP について説明する. TCP については広く実装されている従来型のトランスポート層プロトコルである TCP-Reno と, 提案手法であり QoS 実現を目指す TCP-AV について説明する.

2.2.1 UDP

UDP はデータ転送の信頼性を提供しない, コネクションレス型 (データグラム型) である. UDP は複雑な処理を行わない代わりに, 高速で動作する. そのため, ネットワークが混雑していても送信量を制御せず, パケットが失われても再送制御しない. 通信相手にパケットを届けるだけの仕事を行い, 届いたパケットにエラーが含まれていないことだけを保証する. したがって動画や音声な

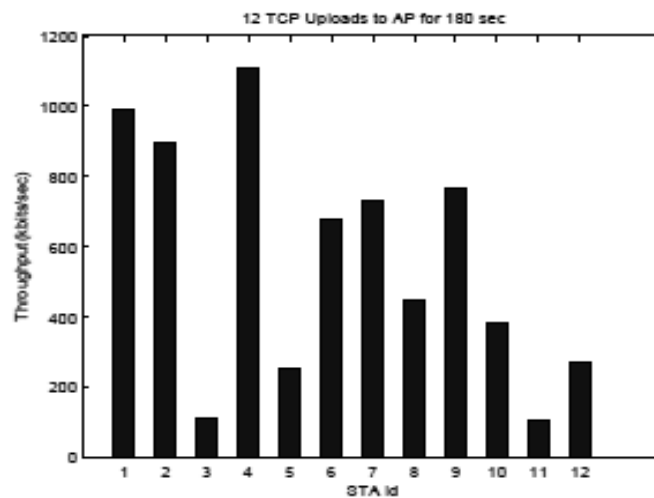


図 2.5: 実機環境におけるスループット

どのマルチメディア通信のように高速に処理を行いたい通信には UDP がよく用いられる。

2.2.2 TCP-Reno

TCP-Reno はトランスポート層で広く実装されているプロトコルである。ここで Linux の TCP 実装 [8] について詳しく説明する。まず通常は正常状態（状態 `TCP_CA_OPEN`）で動作している。このときには通常の TCP 輻輳ウィンドウ制御アルゴリズム（スロースタート、輻輳回避）に基づき輻輳ウィンドウは単調に増加する。これを続けると、一度に送り出すデータはその分だけ大きくなっていく。

しかし、何かエラーが起こると異常状態に遷移し、輻輳ウィンドウは急激に減少する。異常状態には `TCP_CA_RECOVERY`、`TCP_CA_CWR`、`TCP_CA_LOSS` の 3 つの状態があり、それぞれ重複 ACK または SACK の受信、Local Device Congestion、タイムアウトなどの原因で遷移する（図 2.7 参照）。Local Device Congestion とはデバイスドライバのバッファ（NIC ディスクリプタ値）があふれるというエラーであり、データ送信時に TCP が下位層へ送り出すデータ量がバッファの大きさを超えると発生する。これが起こると、輻輳ウィンドウが急激に減少するが、その後正常状態に戻り、輻輳ウィンドウは再び単調増加を始める。またネットワークが混雑している場合には重複 ACK または SACK の受信、さらにはタイムアウトによるエラーも頻発する。これらのエラーの場

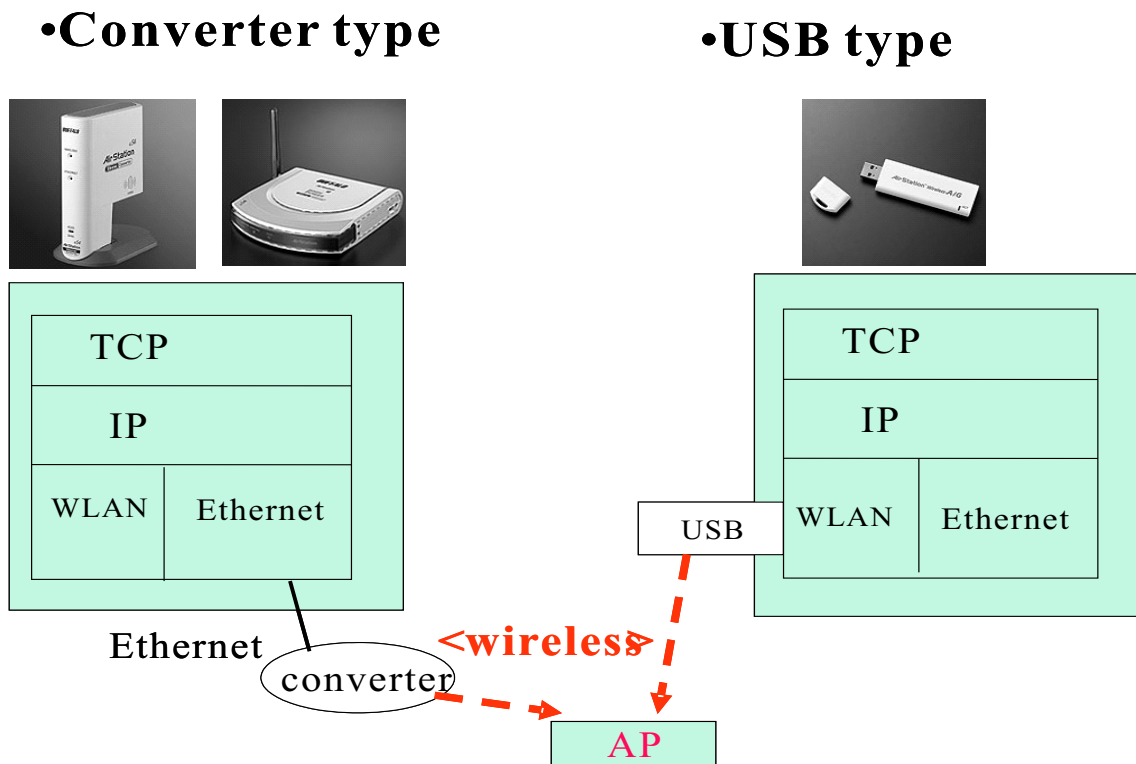


図 2.6: 無線子機と MAC 層の違い

合も発生すると異常状態に遷移して輻輳ウィンドウは急激に減少し、その後また単調増加に戻る。

2.2.3 TCP-AV

TCP-AV はストリーミング通信の品質向上を目指して提案された。目標帯域を確保するための制御を行っている。既存の TCP の輻輳制御方式を拡張し、パケットのバーストロス耐性を向上する再送制御方式を採用している。具体的には、再送クライアントの受信バッファを枯渇させない輻輳ウィンドウ制御、一時的な輻輳発生時の通信速度減少を抑える再送制御である。帯域確保 TCP の累積パケット量を再生クライアントにおける受信バッファ量の推定とみなし、目標送信速度をコンテンツのエンコーディングレートとみなすことができる。そのため、この輻輳ウィンドウ制御は受信バッファ量を一定に保ち、枯渇させない制御となり、結果としてストリーミング再生の停止を防ぐことができる。

また、本方式では、SACK (Selective ACK) で通知される紛失セグメントから、一時的な輻輳か、重度な輻輳かを判断する。輻輳の程度は、再送が完了していない紛失セグメント量が閾値以下であ

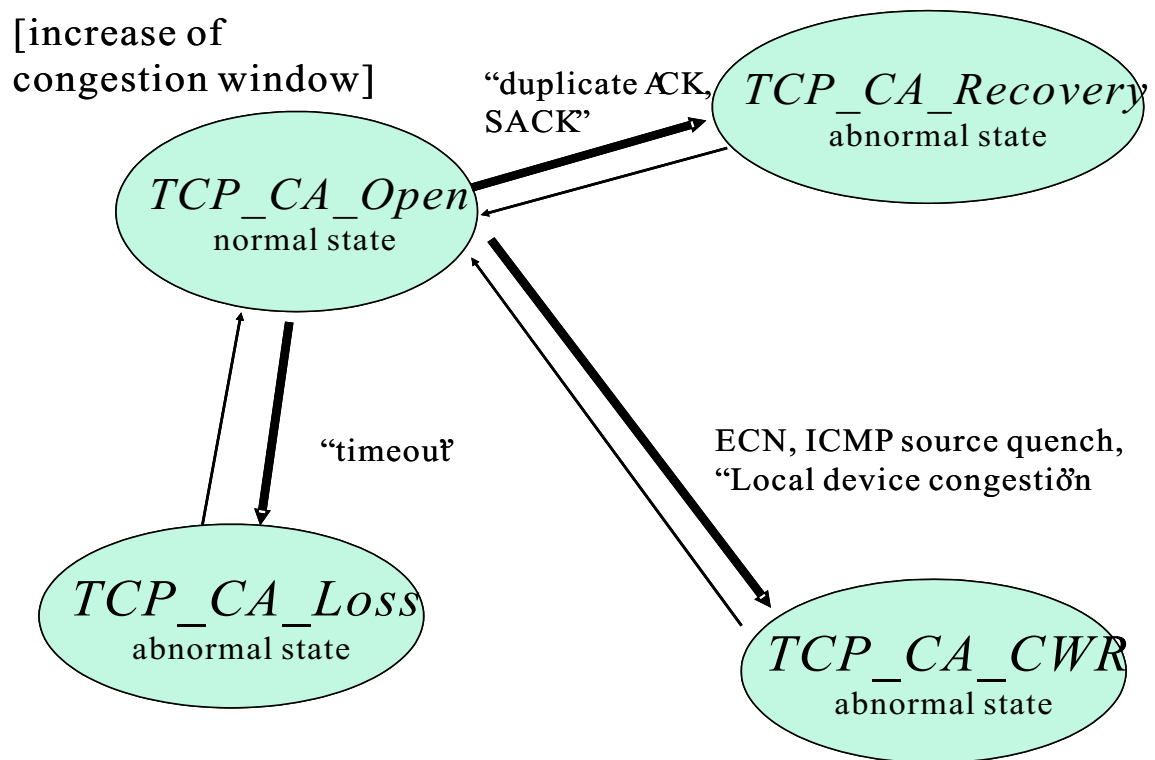


図 2.7: 輻輳ウィンドウの状態変化

るかで判断する。一時的な輻輳であると判断された場合には、積極的に再送を行い、輻輳ウィンドウを減少をさせない。一方、重度の輻輳である場合は、これを回避するため輻輳ウィンドウを減少させる処理を行う。このような制御により、一時的な輻輳の場合はスループットを維持できるため、ストリーミング品質を安定させることができる。また、重度な輻輳である場合には、スループットを抑える制御を行うために、輻輳崩壊を回避できる。

第3章 公平性に関する実機評価

無線 LAN 環境における QoS 実現に必要な要素技術を評価するために、無線 LAN 環境における各プロトコルの公平性について調べる。本実験環境は、既存研究で不公平が示されている環境と同様に、複数の無線 LAN 端末から有線の 1 台の端末にデータを送る環境を想定している。この環境は一般的に用いられるデータ配信などとは逆の向きである。

本実験環境においては、OS として WindowsXP と LinuxOS (カーネルバージョン 2.4 および 2.6) を、アクセスポイントとして、BUFFALO WZR-AMPG300NH[9] を使用した。また、遅延やパケットロスなどのパラメータの変更のために Dummynet (遅延装置) を用いた。また、本研究ではすべて無線 LAN として、IEEE802.11g[10] を使用した。これは、カードバスタイプ 802.11b を用いた際の検証がすでに行われているためであり、また、現在この規格の無線 LAN が広く使用されているためである。

また、以下の実験のために、EB (Effective Bandwidth) ,fair-share を定義する。EB は実際の最大の帯域幅のことであり、実際に実験環境においてフロー 1 本あたりのスループットを測定すると最大で約 25Mbps となったため、この値を本研究の EB とする。fair-share とは EB を送信端末の台数で割った値であり、実効的に使える最大帯域を全送信端末で均等に分けたときの 1 台あたりの帯域幅であるとする。

3.1 同機種無線 LAN 環境における TCP フローの公平性

3.1.1 実験概要

複数の無線 LAN 送信端末から有線側の受信端末にデータを送る環境においてシミュレーション、カードバスタイプの 802.11b 無線 LAN を用いた実機において不公平が起こることがわかっている。しかし、イーサネットコンバータタイプのみを用いた環境においては不公平が起こるかどうか明らかではない。さらに、TCP-AV を使用する際などには、イーサネットコンバータタイプの無線子機

を用いるのが便利である。そこで、イーサネットコンバータタイプのみを用いる環境を構築した。このとき送信端末の台数を3～7台と変えて実験を行った。スループットの測定に Iperf[11] を用いた。フローの投入順序に影響されず、また安定した測定を可能にするために測定時間は120秒とした。この測定を10回行い、それぞれ前述のような不公平が観測されるか調べる。また、WindowsとLinuxで Iperf のデフォルトのバッファサイズが違うのでバッファサイズは64kバイトに統一した。本実験では、パラメータは片道遅延2.5ミリ秒とした。実験環境は図3.22に示した。ただし、図は送信端末の台数が5台の場合を表している。

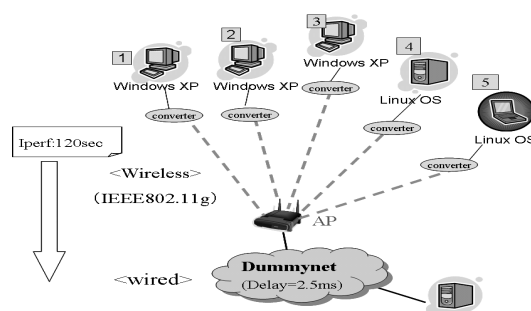


図 3.1: 同機種無線 LAN による実験環境

3.1.2 実験結果と考察

送信端末の台数が3台～7台の場合のそれぞれのスループットを図3.2～図3.6に示す。送信端末が3台～4台の場合は、ほぼ公平に帯域を分け合っていた。しかし、送信端末を5台～7台にした際には、10回の測定、ほぼすべてにおいて不公平に陥る端末が現れた。

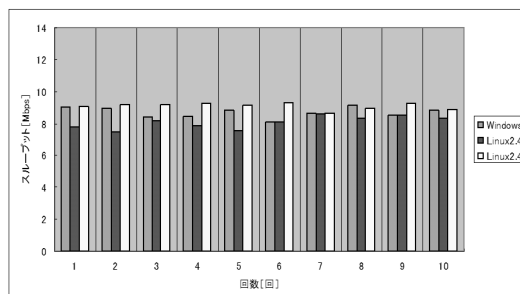


図 3.2: TCP フローが公平な場合（送信端末数 3）

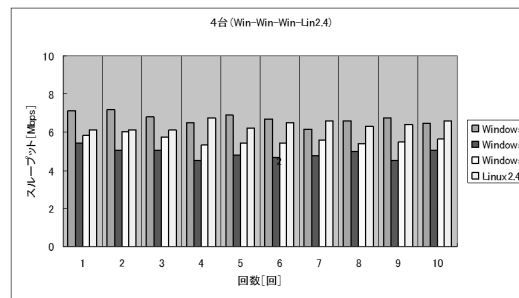


図 3.3: TCP フローが公平な場合 (送信端末数 4)

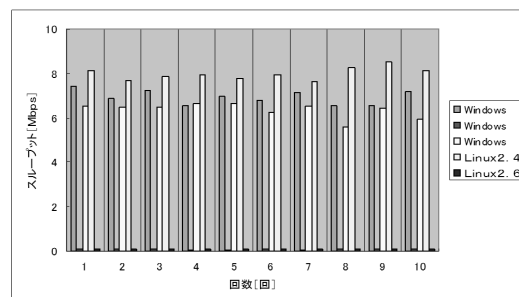


図 3.4: TCP フローが不公平な場合 (送信端末数 5)

これはイーサネットコンバータ無線子機のみを使用しているため送信能力が同じなので通常は MAC 層の制御などにより、同様にロスして平等になると考えられるが、端末の台数が増えるとアクセスポイントのバッファあふれと MAC 層における送信権制御、および、トランスポート層における輻輳ウィンドウ制御が組合わさった原因から不公平が観測されたと考えられる。

シミュレーション環境における結果においては、毎回異なる端末が不公平になるといわれているが、本実験では一度の測定中は同じ端末が不公平になる傾向が見られた。なので、イーサネットコンバータタイプ独特の制御の問題などにより、本実験環境では実機特有の不公平が起これと考えられる。そこで、実機の端末において、不公平になりやすい端末があるかなどを検証していく。

3.2 不公平の傾向についての検証

シミュレーションにおいては全ての端末を平等に扱うため、確率的な不公平が起これ。しかしながら、実機を用いた環境においては条件が完全に平等にはならないため、実機特有の不公平が起これ可能性がある。これはつまりハードウェアや OS に不公平になりやすいものがあるのかというこ

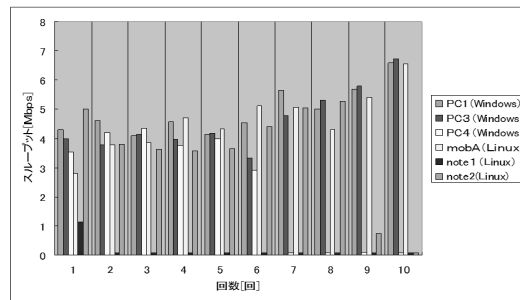


図 3.5: TCP フローが不公平な場合 (送信端末数 6)

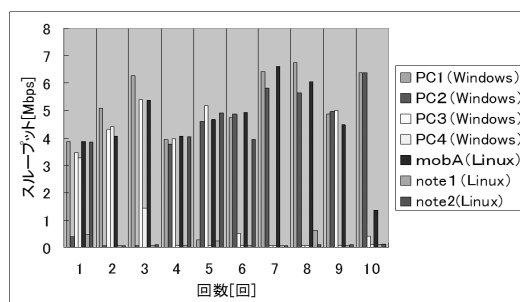


図 3.6: TCP フローが不公平な場合 (送信端末数 7)

とである。これらの傾向があるのか又はないのかということを調べるため、以下の実験を行った。

3.2.1 基礎実験 (前提条件)

まずはじめに、端末などの故障や電波の強度がこの不公平に関わっていないことを示す。

(1) 端末

端末が故障していないことを示すため、使用する無線 LAN クライアントを固定して図 3.7 のように Iperf を 120 秒間用いて一台ずつスループットを測定した。

結果は図 3.8 のようになった。どの端末もスループットは 20Mbps ~ 25Mbps 前後であり、IEEE802.11g 無線 LAN の所定の性能であると言える。よって、端末の故障によるスループットの不公平は考えにくい。また、WindowsXP の端末と LinuxOS の端末間で多少のスループットの違いが見られるが、本研究で問題とする不公平ほどではないといえる。

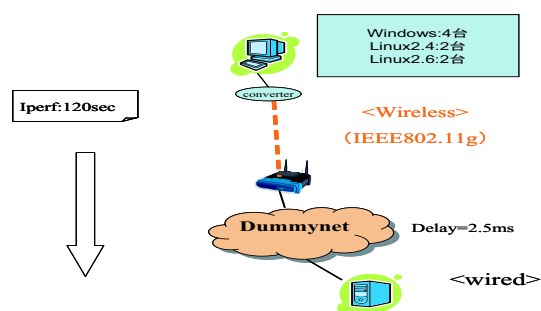


図 3.7: 端末の故障がないことを確かめる実験環境

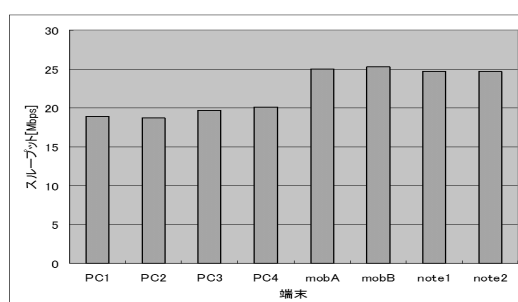


図 3.8: 端末の故障がないことを確かめる実験

(2) 無線 LAN クライアント

本実験で使用する無線 LAN クライアントであるイーサネットコンバータタイプ無線 LAN の故障によって不公平が起きていないことを示すために、端末を固定した。Iperf を用いて 120 秒間スループットを測定した。実験環境は図 3.9 の通りである。端末の OS は WindowsXP, Linux2.4, Linux2.6 とした。

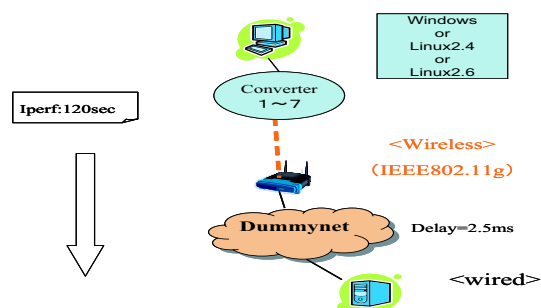


図 3.9: 無線 LAN クライアントの故障がないことを確かめる実験環境

結果は図 3.10 のようになった。これにより、所定の性能がでているので、イーサネットコンバー

タタイプ無線 LAN の故障は考えにくい。また, やはり WindowsXP の端末と LinuxOS の端末間でスループットの違いが見られるが, 不公平ほどではないといえる。

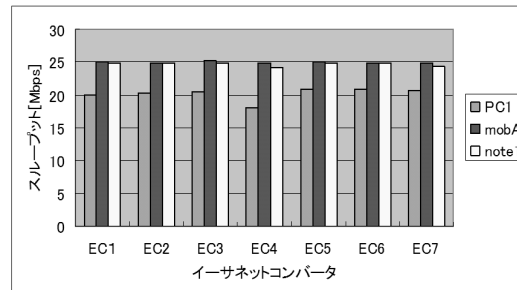


図 3.10: 無線 LAN クライアントの故障がないことを確かめる実験

(3) 無線 LAN クライアントの位置 (電波の強度)

完全に電波の強度を一致させることはできないが, 本実験環境においては無線 LAN クライアントを図 3.11 のように配置しているため, 電波による強度の違いは考えなくてよいものとする。

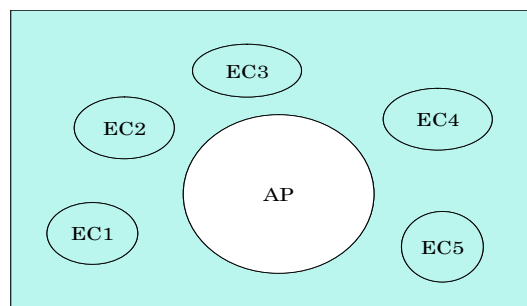


図 3.11: 無線 LAN クライアントの配置

3.2.2 マシンのスペックや OS

上記の実験により, 機器の故障や電波の強度の違いは考えなくてよいものとする。そこで, 次に端末のスペックや OS の違いを考える。2 台ですべての組み合わせでスループットの測定を行った。実験環境は図 3.12 の通りである。

結果を図 3.13 と図 3.14 に示した。縦軸はスループットを示している。横軸に 1 ~ 10 回の結果を並べて表示している。それぞれの回には, 端末ごとのその回のスループットを示している。これ

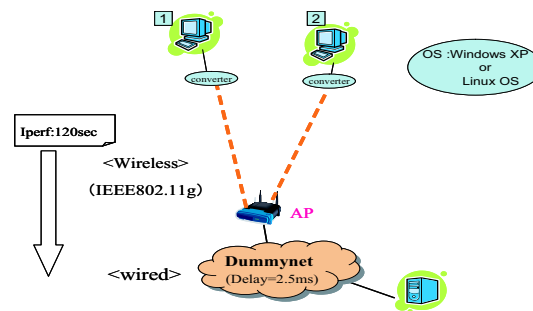


図 3.12: マシンのスペックや OS の違いに関する実験環境

らから、2 台のときはほぼ公平であることがわかる。WindowsXP と LinuxOS を比較すると、多少 LinuxOS のスループットが高いが、不公平ほどではない。また、3.14 の結果から同種の OS の端末同士でも多少のスループットの差が見られる。

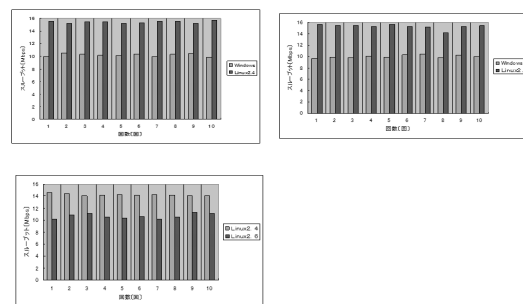


図 3.13: マシンのスペックや OS の違いに関する実験 1

3.2.3 端末の組み合わせによる傾向

実際に不公平が起こった 5 台の場合に実機では OS などの違いで落ちやすい傾向があるのかを調べる。そのために、端末の組み合わせを変えて実験を行った。

実験環境は図 3.22 と同様である。Iperf を用いて 120 秒間スループットの測定を行った。結果は、図 3.15 ~ 図 3.19 に示した。縦軸はスループットを示している。横軸に 1 ~ 10 回の結果を並べて表示している。それぞれの回には、端末ごとのその回のスループットを示している。

このように様々な組み合わせで行った。しかし、一度不公平になった端末が連続して不公平になることがわかったが、それ以外には多少偏りがあっても特に傾向がない。このことから、多少 OS や無線 LAN クライアントの位置、組み合わせなどによる差がみられても実験を重ねていくと不公

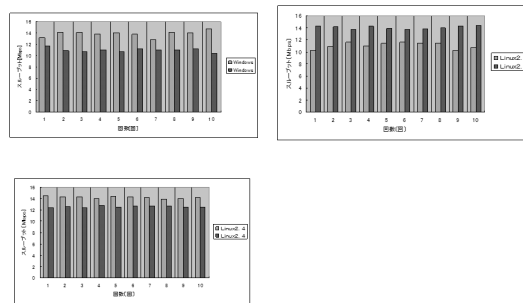


図 3.14: マシンのスペックや OS の違いに関する実験 2

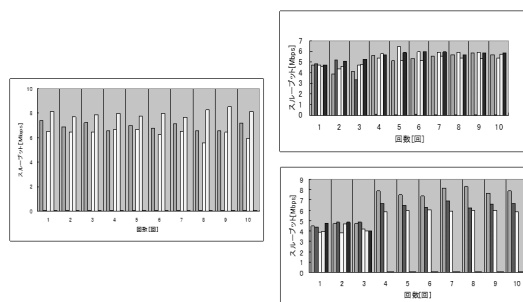


図 3.15: 5 台の組み合わせを変えた実験 1 (Windows 3 台, Linux2.4 1 台, Linux2.6 1 台)

平になる端末が変わることからシミュレーションの傾向に近づくことがわかる。つまり、実機を用いた環境においてもシミュレーション同様理論的に不公平になると考えられる。また、イーサネットコンバータタイプの無線 LAN を使うと、端末がどのような OS を使用するかに影響を受けにくいと考えられる。

3.3 原因の検証

実機における不公平がシミュレーションに近い傾向がみられた。そこで、シミュレーションで原因として考えられている AP のバッファあふれによる ACK のロスが本実験環境においても不公平の原因となりうるかどうか検証する。

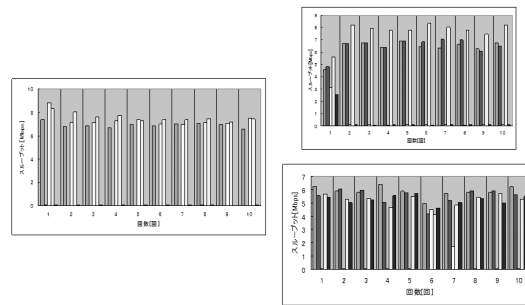


図 3.16: 5 台の組み合わせを変えた実験 2(Windows 2 台,Linu2.4 1 台,Linux2.6 1 台)

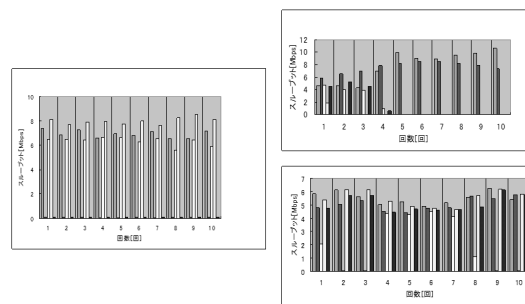


図 3.17: 5 台の組み合わせを変えた実験 3(Windows 3 台,Linux2.6 2 台)

3.3.1 AP のバッファサイズの検証

(1) 実験概要

まず, 無線 LAN AP のバッファサイズを求めた. 無線 LAN AP として, 市販されている AP の中でもごく一般的であり, 本実験環境で使用している BUFFALO WZR-AMPG300NH[9] を使用した. 図 3.20 のように 1 台の有線端末から番号を付加した UDP パケットを AP を経由して, 無線端末に送り出す [12]. これを受信した無線端末において, 受信回数と送信端末で付加された番号を比較する. 受信回数と番号が一致していれば, パケットがロスすることなく通信が行われているといえる. しかし, 番号が飛んでいる場合, その間のパケットはロスしていると考えられる. 図 3.20 のようなネットワーク構成の場合, パケットロスの原因は有線側の AP のバッファあふれであるといえる.

(2) 実験結果

AP のバッファサイズの測定を 10 回行った. その結果を基に AP のバッファサイズを推察したところ, 約 300 パケットとなり, シミュレーションで用いられる値と大きく変わらない結果となった.

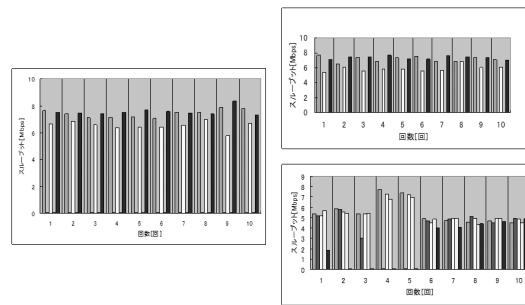


図 3.18: 5 台の組み合わせを変えた実験 4(Windows 4 台,Linux2.6 1 台)

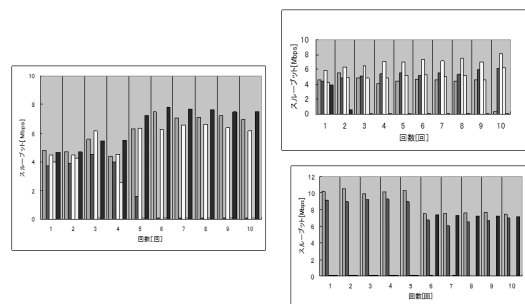


図 3.19: 5 台の組み合わせを変えた実験 5(Windows 4 台,Linux2.4 1 台)

実機の測定結果がシミュレーションの値と近くなったことから、実機においてもシミュレーション同様 AP のバッファにおける TCP-ACK のあふれが不公平の原因のひとつである可能性が高いと推測される。

3.3.2 UDP を用いた環境

次にトランスポートプロトコルに UDP を用いる実験を行う。ここで、公平性を示すことができれば、ACK のロスが TCP フローの不公平の原因であることの根拠の 1 つとなると考えられる。Iperf を用いて、UDP フローを流し、スループットを測定した。その際、フローの投入順序に影響されず、また安定した測定を可能にするために測定時間は 120 秒とした。UDP は 1470byte データグラムとした。また、Windows と Linux で Iperf のデフォルトのバッファサイズが違うので UDP バッファサイズは 64k バイトに統一した。また、帯域幅は有線のリンクと同じ 100Mbps とした。パラメータは片道遅延 2.5 ミリ秒とし、10 回の測定の平均スループットを結果として用いる。

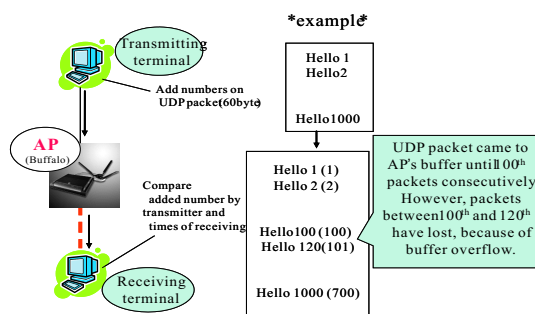


図 3.20: AP のバッファサイズ検証実験

(1) 基本性能

まず図 3.21 のようにフロー本あたりのスループットを調べた。本実験では Windows の端末と Linux の端末の両方を使用しているので、両方のスループットを測定した。

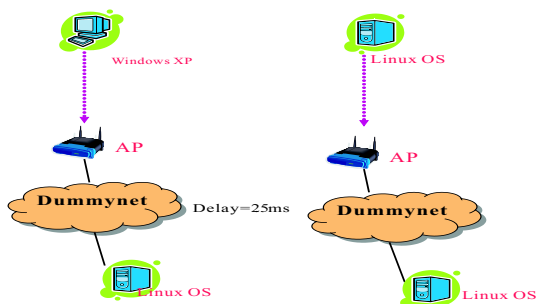


図 3.21: UDP を用いた際の基本性能

その結果が表 3.1 である。この表の左側は Linux の端末の、右側は Windows の端末の値を示している。表中の値は端末から送り出すデータのスループットであり、() 内の値は受信端末が受信することができたデータのスループットである。縦の項目は図の送信端末と AP の間の点線部分を何で通信しているかを示している。wired は、イーサネットケーブルで接続していることを、EC はイーサネットコンバータタイプ無線 LAN で通信していることを、USB は USB タイプ無線 LAN で接続していることを示している。Linux 端末では USB タイプの IEEE802.11g 無線 LAN を動作させるドライバとして使用できるものがなかったため、実験を行っていない。Linux 端末の有線とイーサネットコンバータの送り出すスループットがほぼ同じであることから、イーサネットコンバータタイプの無線 LAN は端末からは有線と認識されていることがわかる。また、このときの Linux と Windows の受信側のスループットが送信側のスループットと一致していないことから、

表 3.1: フロー本あたりのスループット

	Linux	Windows
wired	93Mbps(93Mbps)	97Mbps(90Mbps)
EC	92Mbps(30Mbps)	34Mbps(33Mbps)
USB		27Mbps(27Mbps)

たくさん送り出すことができてもコンバータの先の無線の部分で送信権を得て AP に届けることができず、全てのパケットを受信端末まで届けることができていると考えられる。また、USB タイプは端末が無線と認識して、送り出しているので送信権を得られる程度のパケットを送り出しているためスループットは低いロスほとんど見られない。

(2) 同機種無線 LAN 環境

文献 [2] ではカードバスタイプの無線子機において TCP フロー間でスループットの不公平が起こることが示されている。さらに、イーサネットコンバータタイプのみを用いた環境においては不公平が起こることを前節で示した。また、前述の実験ではイーサネットコンバータタイプの無線 LAN で高いスループットが示された。そこで、イーサネットコンバータタイプのみを用いる環境を構築した。Iperf を用いて 120 秒間スループットの測定を行った。これを 10 回行い、平均を取った。このとき送信端末の台数を 2～5 台と変えて実験を行った。実験環境を図 3.22 に示す。ただし、図は送信端末の台数が 5 台の場合を表している。

実験結果を図 3.23 に示す。横軸が端末番号、縦軸がスループットを示している。また、この図は端末の台数が 5 台のときの結果であり、10 回の測定とも同様の傾向であったので、ここで示すスループットはそれぞれの受信端末において測定したスループットの平均である。また、送信端末が 2 台～4 台のときも同様の結果であった。

この結果から UDP フローを用いた場合、スループットはほぼ公平であることがわかった。UDP はフロー制御を行わず、ACK を使用しない。そのため、UDP と TCP の違いを考慮すると、TCP フローの不公平には ACK が関わっていることが考えられる。

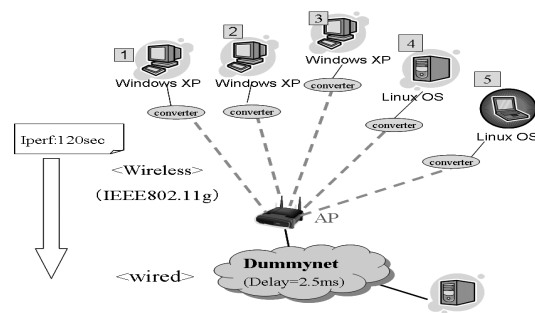


図 3.22: 同機種無線 LAN 環境 (UDP)

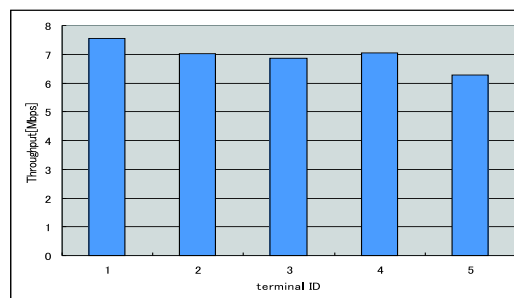


図 3.23: 同機種無線 LAN 環境のスループット (UDP)

3.3.3 一般的な環境における TCP を用いた環境

次にトランスポート層プロトコルに TCP を用いる実験を行う。この際、一般的にデータ配信などで使用され、本実験で想定している環境とは逆向きである。有線側の送信端末から無線側の複数の受信端末にデータを送る実験を行い、本環境での基本性能を確認する。スループットの測定には Iperf を 120 秒間用いた。Windows と Linux で Iperf のデフォルトのバッファサイズが違うのでバッファサイズは 64k バイトに統一した。パラメータは片道遅延 2.5 ミリ秒とした。

まず図 3.24 のように一般的な有線側の送信端末から無線側の複数の受信端末にデータを送る実験を行う。無線子機としてイーサネットコンバータを用いた。端末の台数は 2~5 台とした。図は受信端末の台数が 5 台の場合を示している。端末番号 1~4 に Windows, 端末番号 5 に Linux をそれぞれ OS として使用した。

結果を図 3.25 に示す。10 回の測定とも同様の傾向のため平均のスループットを示した。このときスループットはほぼ公平となった。CSMA/CA の仕組み通り、AP からそれぞれの端末に平等にデータを送り、平等に ACK を返すことができていると考えられる。データは最大 100Mbps で送

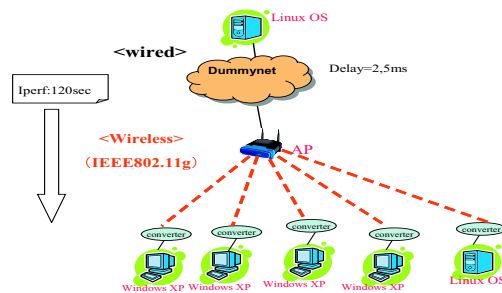


図 3.24: 一般的な無線環境における TCP フローを流す実験

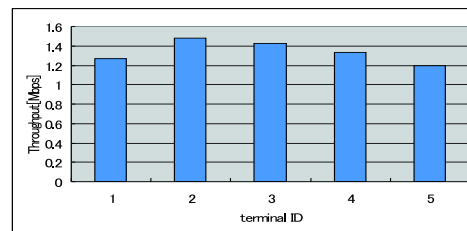


図 3.25: 一般的な無線環境における TCP フローのスループット

信され、ダミーネット上での遅延の影響や、AP の有線側バッファのデータパケットのあふれは考えられるが、無線側から有線側の向きでの ACK ロスはあまり起こらないため公平な結果になったと考えられる。なので、AP のバッファあふれが本実験環境における不公平に関係していると考えられる。

3.4 異機種無線 LAN 混在環境における TCP フローの公平性

次に、実際の環境に近づけるために異なる種類の無線子機の混在環境において実験を行った。また、このような無線 LAN クライアントが異なるものが混在している環境はシミュレーションでは考慮しにくく、実機特有の特徴が見られる可能性がある。

3.4.1 基礎実験

異機種環境での実験を行う前に基礎実験として、USB タイプ無線 LAN のみを用いて実験を行った。実験環境は図 3.26 のように USB タイプ無線 LAN を使用した 3 台の送信端末から、1 台の有

線の受信端末にデータを送信する。その際,UDP 1 M 指定,10M 指定,100M 指定と TCP のそれぞれのスループットを Iperf を用いて測定した。

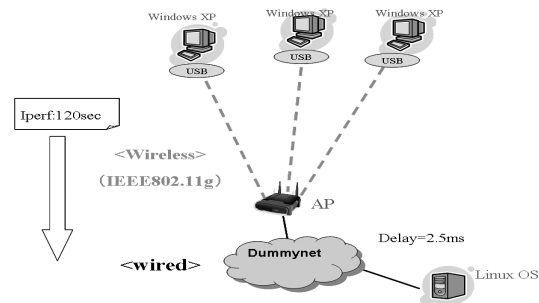


図 3.26: USB タイプ無線 LAN のみを用いた実験環境

結果は図 3.27 のようになった。UDP1M 指定および 10M 指定のときは指定した通りに通信を行うことができていた。しかし,UDP100M 指定および TCP のときは,その合計値が通常のフロー一本あたりの帯域に近い値であった。また,このときのスループットは公平であった。

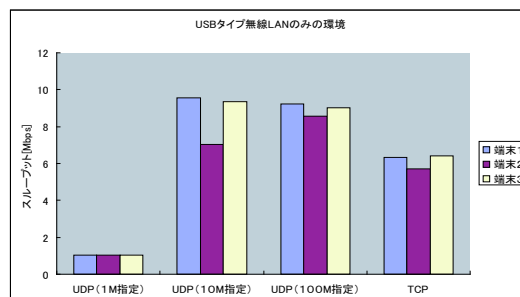


図 3.27: USB タイプ無線 LAN のみを用いた環境のスループット

3.4.2 UDP

異機種無線 LAN 端末混在環境における UDP の公平性の違いを検証する実験環境を図 3.28 に示す。また,少ない端末台数においてはスループットの不公平が起こらない可能性があるため,7 台の送信端末とした。OS として,端末番号 1, 8 に Linux2.4, 端末番号 6 と 7 に Linux2.6, 端末番号 2 ~ 5 に windows XP を使用した。無線子機として,端末番号 1,2,6,7 にイーサネットコンバータタイプを,端末番号 3 ~ 5 に USB タイプを使用した。7 台の無線送信端末から AP, ダミーネットを経由して 1 台の有線端末にフローを流したときのスループットを測定する。USB タイプの無線子機と

イーサネットコンバータタイプの無線子機の混在環境の場合, MAC 層における違いから UDP を用いてもスループットの公平性の違いが現れる可能性がある。

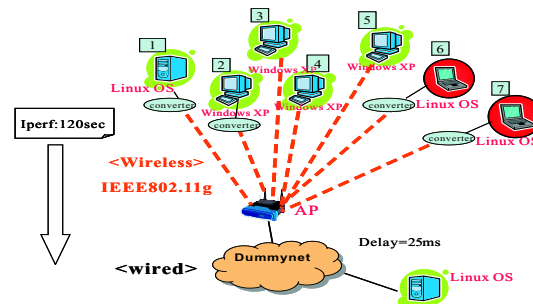


図 3.28: 異機種無線 LAN 端末混在環境を用いた実験環境 (UDP)

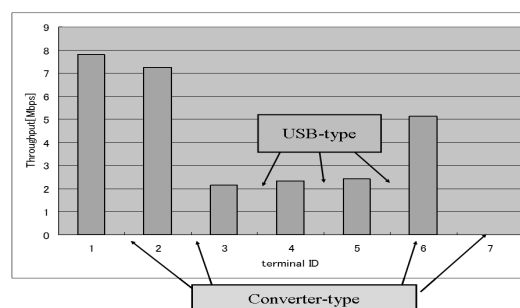


図 3.29: 異機種無線 LAN 端末混在環境のスループット (UDP)

このときのスループットを図 3.29 に示す。横軸が端末番号, 縦軸がスループットであり, ここで示すスループットは受信端末において測定したスループットである。

これを見ると, USB タイプは常に fair-share ($EB \div 7 \text{ 台} = \text{約 } 3\text{Mbps}$) 程度のスループットが出ているのに対し, イーサネットコンバータタイプはスループットがこれより高い値となっている。このとき, 基本性能でフロー本あたりのときと同様, 有線のリンクいっぱいにデータを送り出して大量にロスをしているが, 送出している量が多いため, USB タイプより高いスループットとなったものであり, 文献 [2] や [3] で述べられているような不公平の問題による性能の違いではないと考えられる。また, 端末番号 7 の端末は何らかの理由により, 送信したデータを受信端末に届けることができなかった。これらは, TCP フローのスループットの不公平の原因とは異なると考えられ, UDP フローを用いた場合は送り出す部分のリンクの太さによる不公平が起こることが示された。

3.4.3 TCP フロー

次に,UDP の時と同様に実際の環境に近づくために異なる種類の無線子機の混在環境において実験を行った. 異機種無線 LAN 端末混在環境における TCP の公平性の違いを検証する実験環境を図 3.30 に示す. 前述の実験同様 7 台の送信端末とした. OS として, 端末番号 1, 8 に Linux2.4, 端末番号 6 と 7 に Linux2.6, 端末番号 2~5 に windows XP を使用した. 無線子機として, 端末番号 1,2,6,7 にイーサネットコンバータタイプを, 端末番号 3~5 に USB タイプを使用した. 7 台の無線送信端末から AP, ネットワークエミュレータ(ダミーネット)を経由して 1 台の有線端末にフローを流したときのスループットを測定する. USB タイプの無線子機とイーサネットコンバータタイプの無線子機の混在環境の場合, MAC 層における制御の違いからスループットの公平性の違いが現れる可能性がある. はじめに片道遅延 2.5 ミリ秒の環境とし, 次に片道遅延 20 ミリ秒, パケットロス 0.5%の環境とした. 一般的にパケットロスがあるほうが公平性が増すといわれている [13].

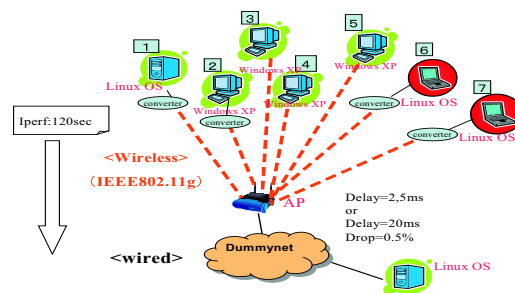


図 3.30: 異機種無線 LAN 端末混在環境において TCP フローを流す実験

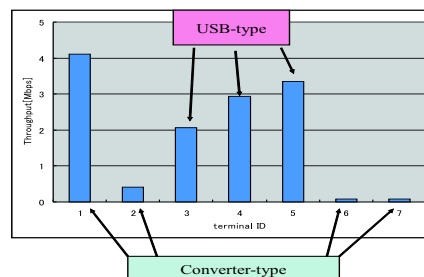


図 3.31: 異機種無線 LAN 端末混在環境における公平性（遅延 2.5 ミリ秒）

はじめに遅延が 2.5 ミリ秒の際のスループットの公平性の違いを図 3.31 に示す. これを見ると, スループットの不公平が見られる. さらに, USB タイプ間, イーサネットコンバータ間とわけて考え

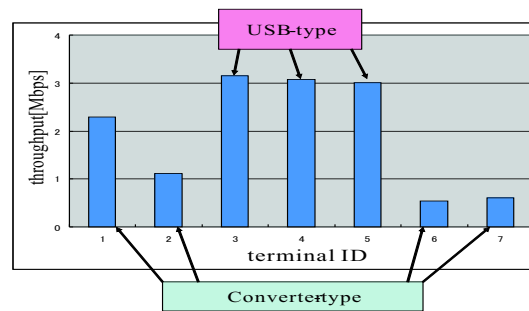


図 3.32: 異機種無線 LAN 端末混在環境における公平性 (遅延 20 ミリ秒, パケットロス 0.5%)

てみても, スループットの不公平が見られる. USB タイプは常に fair-share ($EB \div 7$ 台 = 約 3Mbps) 前後のスループットが出ているのに対し, イーサネットコンバータタイプはスループットが高い場合には fair-share 程度, 低い場合には 1 Mbps 以下となっている.

次に, 遅延 20 ミリ秒, パケットロス率 0.5% という別のパラメータの実験環境におけるスループットの公平性の違いを図 3.32 に示す. これを見ると, 先ほどと同様の傾向が見られるが, 不公平性は緩和している.

この 2 つの実験においてイーサネットコンバータタイプのスループットは高い場合には fair-share 程度, 低い場合には 1 Mbps 以下の 2 つのパターンが見られる. しかし, 本実験結果からだけでは, この違いが MAC 層におけるどのような制御の違いからきているのかは明らかではない.

そこで輻輳ウィンドウの動きとエラーの種類を観測することがきる TCP モニタ [14] というツールとパケットの情報を見ることができる tcpdump コマンドを使ってこの原因を推察する. これらを用いて, イーサネットコンバータタイプでスループットが高い場合 (2.26Mbps fair-share 程度) とスループットが低い場合 (504Kbps) の 2 つの場合を比較する. このときの実験環境を図 3.33 に示す.

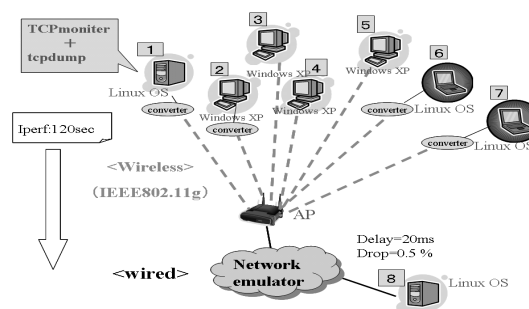


図 3.33: イーサネットコンバータの動作を解析する実験環境

TCP モニタを用いたときの結果を図 3.34 に示した。図中のグラフの実線は輻輳ウィンドウの変化を示しており、破線はその時刻にエラーが起こったことを示している。TCP モニタは3種類の異常状態を見分けることができるが、今回はすべて重複 ACK または SACK 受信によるエラーが観測されている。これはパケットロスが起こっていることを示す。

次に tcpdump コマンドから、データを送信した時間と ACK を受信した時間を取り出して作成した図を図 3.35 に示した。横軸に時間を取り、図中の上段の点がデータを送信した時間を、図中の下段の点が ACK を受信した時間を、それぞれ1パケットにつき1つの点で表した。これにより、データ送信と ACK 受信のタイミングや頻度を明らかにした。なお、図 3.34、図 3.35 とも、上部をスループットが高い場合、下部をスループットが低い場合とした。

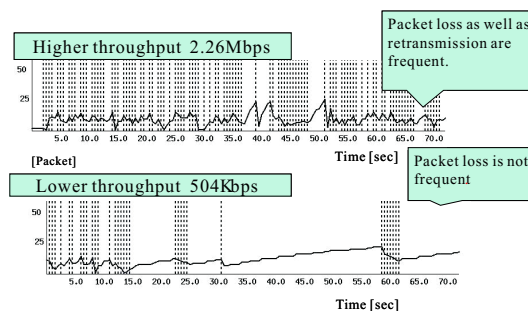


図 3.34: TCP モニタの実行結果

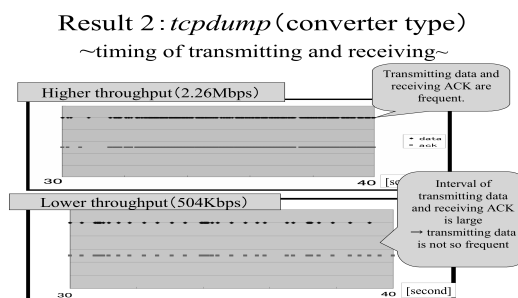


図 3.35: tcpdump の実行結果

この2つのグラフを見ると、イーサネットコンバータタイプのスループットが高い場合には頻繁にデータを送信し、パケットがロスしても積極的に再送を行っており、ウィンドウが増減している。しかし、スループットが低い場合にはゆっくりとしかウィンドウが上がらず、データの送信頻度が低いままである。これは前述の ACK ロスによる影響を受けていると考えられる。

ただし、スループットが低い場合はスループットが低く抑えられているために、スループットが高いときのようにパケットロスを頻発していない。このことから、ACK が返ってこないなどでコンバータの無線部分でパケットが送り出せなくなったときに、コンバータのバッファがあふれるのを防ぐためにパケット送信を緩めるよう、コンバータが OS 側に指示している可能性がある。これにより送信量が減り、全体的なスループットの低下を招いていると考えられる。イーサネットコンバータタイプ無線子機を用いた通信は不安定で、周囲の状況の些細な変化により ACK ロスが起るタイミングが変わるなどして、その結果スループットに大きな影響を受けやすいと考えられるため、ある端末がどちらのパターンになるかは推察が難しい。

また、送信側端末では、コンバータは直接は認識されておらず、従って、コンバータと送信端末間でのローカルな制御情報のやりとりは起こらないはずである。しかしながら、制御情報のやりとりが無く、前記仮定が正しいとすると、送信端末は、ウィンドウがある限りフルスピードでパケットを送信し、その結果、コンバータで多くのパケット破棄が発生するはずである。しかしながら、実験ではそのようなパケット破棄が起らず、またウィンドウも小さくなっていないという現象が観測されているからである。本予測の検証は今後の課題である。

第4章 TCP-AVの無線環境における評価

4.1 TCP-AVの無線LAN環境における実装

TCP-AVを無線LAN環境において実装するために図4.1のように、TCP中継器（TCP-Proxy）を端末とイーサネットコンバータの間に設置し、端末から出たTCPフローをTCP-AVに載せかえる．そうすることで、全体としてTCP-AVのフローとする．この実装のメリットとしては、送信端末、受信端末のTCPの実装を変更しなくてよい点である．

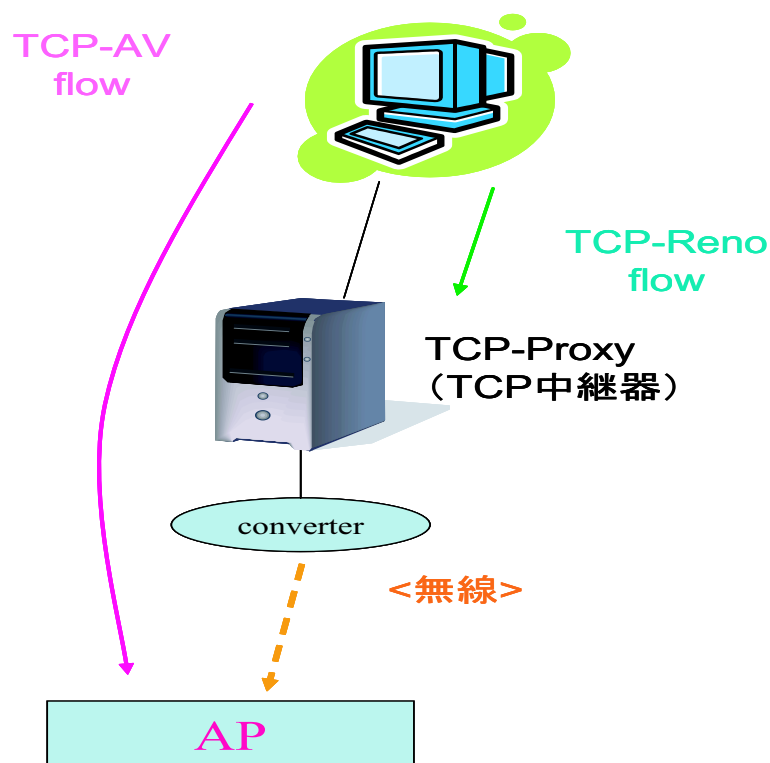


図 4.1: TCP-AVの無線LAN環境における実装

4.2 端末端末が3台の場合

4.2.1 実験概要

3台の無線LAN送信端末を用いてTCP-AVのAVRを調べる実験環境を図4.2に示す。ここで,Flow1はTCP-AVのフローとするが,送信端末のTCPの実装は変更せず,TCP中継器(TCP-Proxy)において,TCPをTCP-AVに載せかえる。Flow2および,Flow3は既存のTCP-Renoを用いて通信を行う。このとき遅延を10ミリ秒と設定した。

ここで,評価指標として,NRT(Normalized Required Throughput),AVR(Achievement Ratio)を定義する。NRTは指定した帯域がEBの何割にあたるかとする。つまり,理論値ではなく,実際の帯域の正味何割を指定したかを表す。AVRは,指定した帯域を確保できた時間帯が全実行時間のうちのどれくらいあるかという比率を示す。今回の実験では120秒の測定を5秒ごとに分割した。ただし,投入順序による差が見られる最初の5秒間は除き,全部で23区間とする。23区間のうち何区間目標帯域を達成できていたかをカウントして,10回測定した平均を取った。

期待される振舞としては,TCP-AVの積極的な帯域確保制御により,fair-share(NRTの32%:8Mbps)以上の帯域を確保できると考えられる。

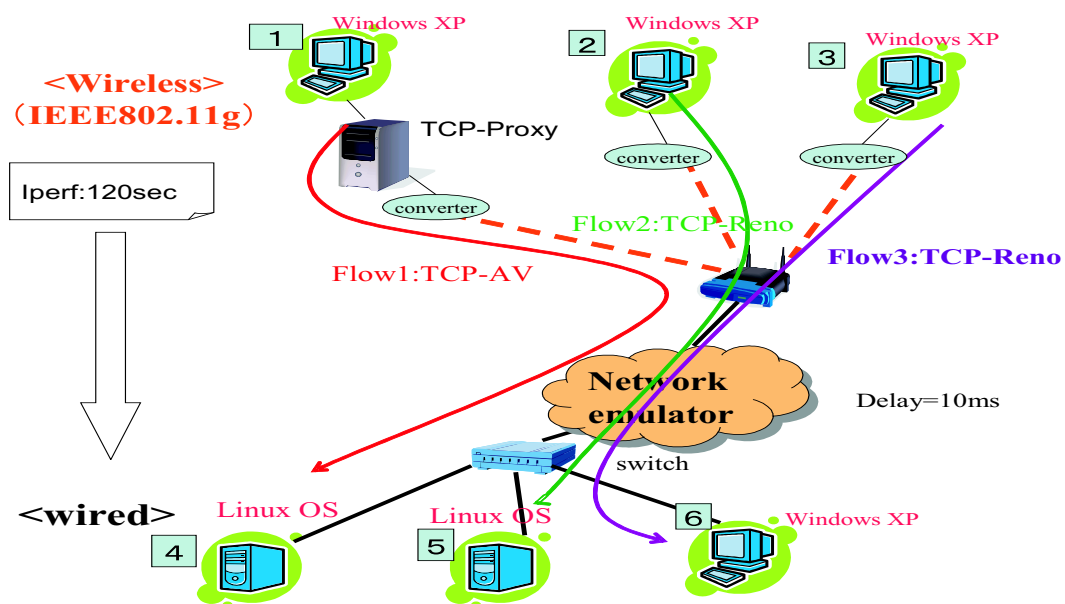


図 4.2: AVR を評価する実験環境

4.2.2 実験結果および考察

結果を図 4.3 の “3flows” に示した。横軸は NRT を、縦軸は AVR を示している。この NRT は、 $NRT100\% = 25\text{Mbps}$ とし、TCP-AV の帯域を NRT の何%を目標として設定したかを表す。ここで、“3flows” とは、図 4.2 のように、TCP-AV フロー 1 本・TCP-Reno フロー 2 本の計 3 本のフローで実験を行ったことを表す。このとき、全実行時間の 9 割程度の時間目標帯域を確保できるのは、NRT の 40%程度までを設定した場合であった。

次に TCP-Reno の端末からのフローを 1 台につき 2 本に増やし、TCP-AV フロー 1 本・TCP-Reno フロー 4 本の計 5 本のフローとした。この結果は図 4.3 中に “5flows” として示した。このとき、全実行時間の 9 割程度の時間、目標帯域を確保できるのは、NRT の 30%程度までを設定した場合となった。これらの結果から、fair-share より 10%増程度であれば、ほぼ全実行時間中、帯域を確保できることがわかった。しかし、この程度の帯域確保にとどまるのは無線の CSMA/CA の送信権の平等性の影響が考えられる。そのため、フロー数よりも端末の台数による影響の方が大きいと考えられる。

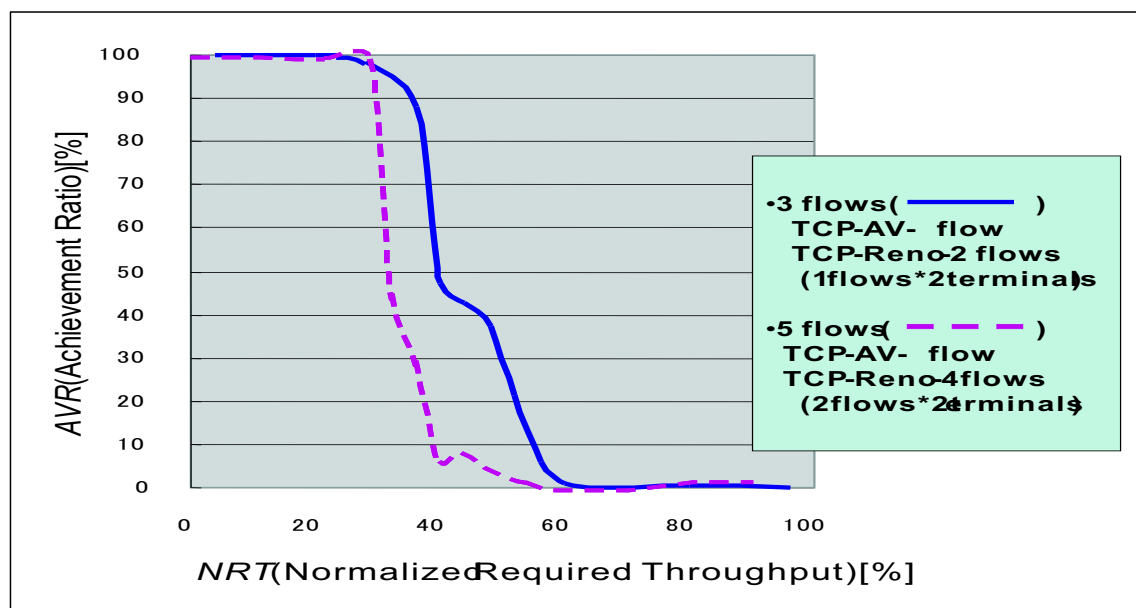


図 4.3: AVR による TCP-AV の評価 1 (送信端末数 3)

4.3 送信端末が5台の場合

4.3.1 実験概要

次に, 送信端末が増加した場合に同様の評価を行った. 実験環境を図 4.4 に示す. 前章で公平性を検証した際に不公平が生じていた 5 台の送信端末における環境とする.

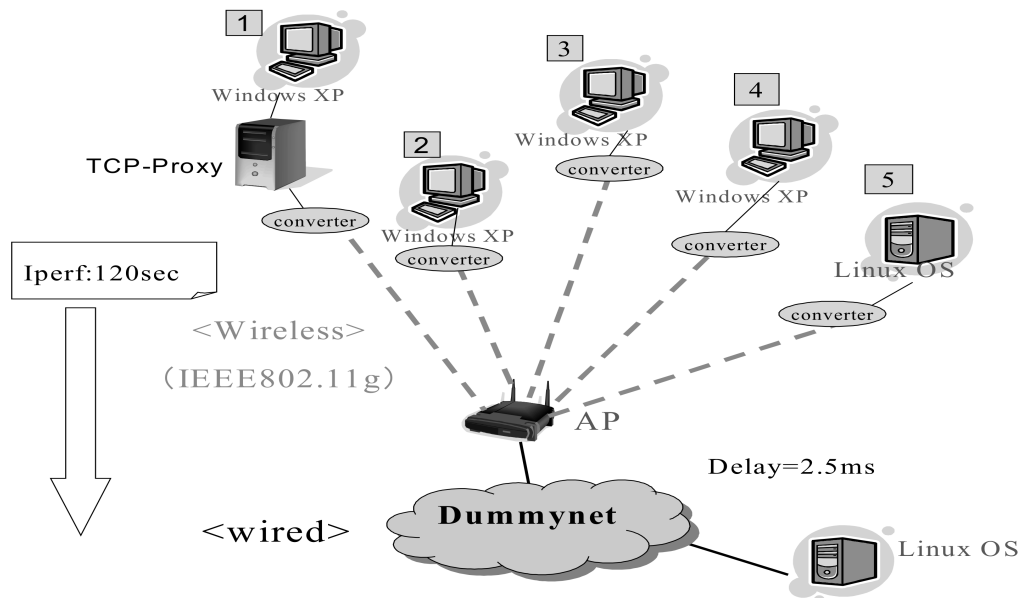


図 4.4: 送信端末が増加した場合の AVR の評価

4.3.2 実験結果および考察

本実験における fair-share は NRT の 20%である 5Mbps である. 結果を図 4.5 に示す. 設定帯域を fair-share 程度とした場合までは AVR は 100%である. つまり, 全実行時間目標帯域以上で通信できていることになる. fair-share を超えて設定帯域を NRT の 25%とすると, AVR は 70%となる. このときは目標帯域以上で通信できている時間が全体の 7 割ということになる. さらに NRT の 40%を設定すると AVR は 1 割以下となる.

これらのことから, TCP-AV の積極的な帯域確保制御により, fair-share を超えた帯域であっても指定した帯域をある程度確保することができた. これにより, 不安定な無線 LAN 環境においても TCP-AV を用いれば端末が不公平に陥らずに, 要求帯域の品質保証が可能であることがわかった. また, 端末台数がこれより少ない場合および 6 台の場合も同様の傾向であった.

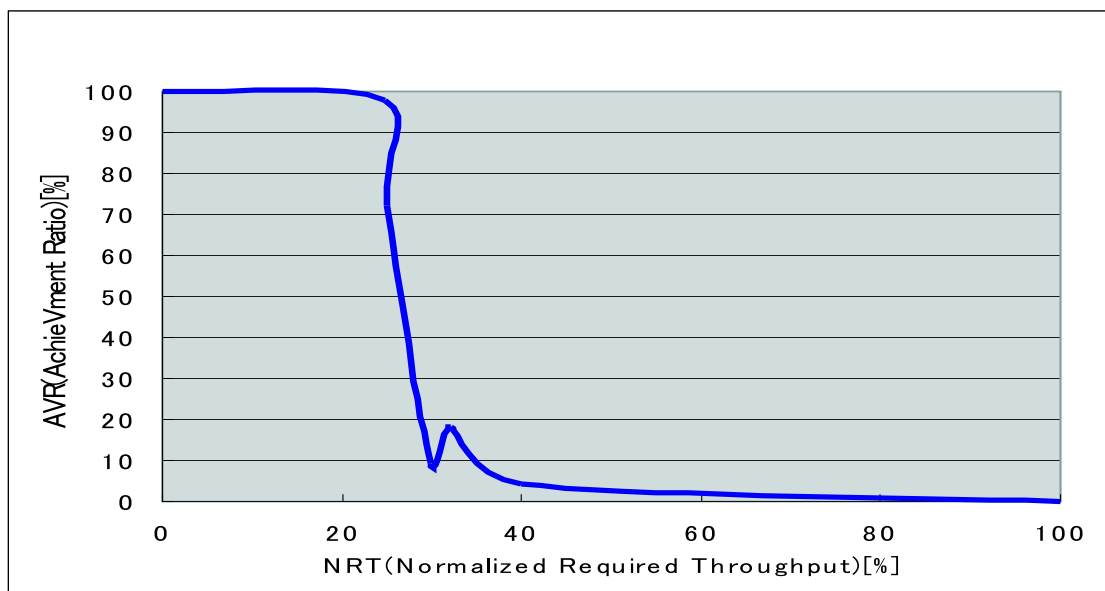


図 4.5: AVR による TCP-AV の評価 2 (送信端末台数 5)

第5章 まとめと今後の課題

5.1 まとめ

本研究では、無線環境における QoS 実現のために実機を用いて実験を行った。無線 LAN 環境におけるスループットの公平性の検証として、イーサネットコンバータタイプ無線 LAN(802.11g) を用いた際の不公平を示した。

その際にシミュレーションとは異なり、同じ端末が連続して不公平になるという傾向が見られた。しかし、端末の組み合わせを変えて実験を行い、不公平になりやすい端末があるのか検証した結果、特に傾向がみられなかったため、シミュレーション同様確率的に不公平が起こればと考えられる。

次に、AP のバッファサイズの検証と UDP フローを流す実験を行うことでシミュレーション同様 AP のバッファあふれによる TCP-ACK の破棄が不公平の原因のひとつとなりえることを示した。

このようにイーサネットコンバータタイプのみの環境においては不公平が起これうということを観測した。そこで、MAC 層が異なるイーサネットコンバータタイプと USB タイプの無線子機の混在環境において、UDP を用いた場合、イーサネットコンバータタイプ子機群と USB タイプ子機群を比較するとタイプ別の不公平が起これていた。しかし、それぞれのタイプ中での各子機のスループットはほぼ公平となった。一方 7 台の無線送信端末から有線の端末に TCP を用いて送信するという環境では、イーサネットコンバータタイプ間のみでスループットの不公平が起これることが明らかになった。そこで TCP モニタツールと tcpdump コマンドを用いて、イーサネットコンバータタイプの挙動について考察を行った。また、このときパケットロスにより、不公平が緩和されることがわかった。

最後に、有線環境において QoS 保証を可能にした TCP-AV を、無線環境において AVR という新たな基準を用いて評価を行った。その結果、TCP-AV は fair-share を超えた帯域（今回の実験では、実効帯域の 30%以上の帯域）を確保することができた。これより、TCP-AV は、実機環境でも、

目標帯域に近い通信を長時間行うことができ、不安定な無線環境においても要求された帯域の QoS を確保することができたといえる。

5.2 今後の課題

不公平性の問題に関しては、イーサネットコンバータタイプ無線 LAN の挙動の詳しい分析を、電波の受け具合や Windows 端末の無線 LAN ドライバによる制御なども考慮にいれて考えたい。また、さらに台数を増やした場合の公平性の变化なども検証したい。

QoS 保証に関しては、実際にストリーミングを流して、TCP-AV の有用性を視覚的に確認したい。また、他の QoS プロトコルを使用したときの挙動、TCP-AV を他の無線子機を使用したときの振舞や高遅延や高負荷をはじめとする様々な無線環境における TCP-AV の評価をしていきたい。

これらのことから TCP-AV の改良など無線環境におけるより良い品質保証の提案を進めていきたい。

謝辞

指導教官である、お茶の水女子大学人間文化創成科学研究科の小口正人教授には、本研究を進めるにあたり、終始有益な御指導ならびに御教示を賜りました。NEC システムプラットフォーム研究所へのインターンと生しての派遣とその際に適切な助言や貴重な時間を割いてミーティングへの参加などをしていただき、心から感謝いたします。研究以外でも就職活動等についてアドバイスをいただき、大変お世話になりました。研究室を移動してからの2年間を、内容の濃い充実したものにしたと思います。深く感謝致します。

副査をお引き受けくださいましたお茶の水女子大学人間文化創成科学研究科の渡辺 千恵美先生に深く感謝いたします。着手発表や中間発表など定期的に進捗具合を聞いていただき、本研究に関する貴重なコメントをたくさんいただきました。

お茶の水女子大学人間文化創成科学研究科 小林 一郎助教授には、卒業研究の際にお世話になりました。ここに記してお礼申し上げます。

NEC システムプラットフォーム研究所の皆様は私をインターン生として快く受け入れてくださいました。また、研究以外の面でも、勉強になることがたくさんあり、貴重な経験をさせていただいたことに心より感謝いたします。さらに、NEC システムプラットフォーム研究所 村瀬 勉さん、平野 由美さんには、私の論文の共著を快くお引き受けくださり、また研究のことに何事からもわからなかった私に根気よく丁寧にアドバイスをしてくださいました。大変お忙しい中であつたのにもかかわらず、いろいろとご面倒をみていただき、本当にありがとうございました。また、本研究に TCP-AV プロトコルに関する情報提供と適切なアドバイスを頂いた NEC システムプラットフォーム研究所の藤田範人さんと浜崇之さんに感謝致します。

そして、博士前期課程から研究室を移動してきた私を暖かく迎えてくれた小口研究室の先輩方、後輩の皆様、そして同期の皆様。皆様のおかげで、3年間の研究生活を大変楽しく有意義に送ることができましたことを、心より感謝いたします。本当にありがとうございました。

最後に、私をここまで育ててくれ、支えてくれた家族や、その他ご支援くださいました皆様に

感謝致します．

参考文献

- [1] H.Shimonishi, et al., "Congestion Control Enhancements for Streaming Media, "IEICE Trans. on Comm.,Vol.E89 B, No.9,pp.2280-2291 Sep. 2006.
- [2] D.Leith, et al., "Using the 802.11e EDCF to Achieve TCP Upload Fairness over WLAN ling," IEEE,2005
- [3] S.Pilosof,R.Ramjee,Y.Shavitt,P.Sinha, "Understanding TCP fairness over Wireless LAN " INFOCOM 2003, 1-3 April 2003.
- [4] Anthony C.H. NG, David Malone, Douglas J.Leith " Experimental Evaluation of TCP Performance and Fairness in an 802.11e Test-bed, "ACM SIGCOMM 2005,pp.17-22 Aug.2005
- [5] BUFFALO AirStation 54Mbps 802.11 abg (WLI-UC-AG):
<http://buffalo.jp/products/catalog/network/wli-uc-ag/>
- [6] BUFFALO 2.4GH 54Mbps wireless LAN Ethernet conversion media converter (WLI3-TX1-G54) :
<http://buffalo.jp/products/catalog/item/w/wli3-tx1-g54/>
- [7] BUFFALO AirStation High Power wireless LAN Ethernet converter 54Mbps (WLI-TX4-G54HP) :
<http://buffalo.jp/products/catalog/item/w/wli-tx4-g54hp/>
- [8] Linux Kerner ver.2.4 implementation: <http://www.linux.org/>
- [9] BUFFALO wireless LAN BB router AirStation NFINITI GIGA 300Mbps 11n conformity (WZR-AMPG300NH) :
<http://buffalo.jp/products/catalog/network/wzr-ampg30nh/>

- [10] IEEE802.11 wireless LAN medium access control (MAC) and physical layer (PHY) specification, 1999
- [11] Iperf: <http://dast.nlanr.net/Projects/Iperf/>
- [12] Perl program of inspect the size of buffer: <http://x68000.q-e-d.net/~68user/net/udp-2.html>
- [13] Yumi Hirano, Tutomu Murase: "Evaluation of Packet Loss Effect on Throughput Unfairness between TCP Upflows over IEEE802.11 Wireless LAN," APSITT 2008, 22-24 April 2008
- [14] 山口 実靖, 小口 正人, 喜連川 優: "iSCSI 解析システムの構築と高遅延環境におけるシーケンシャルアクセスの性能向上に関する考察", 通信学会論文誌, Vol. J87-D-I, No. 2, pp. 216–231, 2004 年 2 月

発表文献

1. 新井絵美，平野由美，村瀬勉，小口正人：
 “ 無線 LAN における帯域公平性の検討および QoS 保証 TCP の動作解析， ”
 電子情報通信学会第 19 回データ工学ワークショップ (DEWS2008)，2008 年 3 月
2. 新井絵美，平野由美，村瀬勉，小口正人：
 “ 無線通信環境における QoS 保証 TCP の動作に関する一検討， ”
 第 70 回情報処理学会全国大会，2008 年 3 月
3. Emi Arai，Yumi Hirano，Tutomu Murase，Masato Oguchi：
 “ Empirical Stdy on a Session Layer Resorce Control over Resorce Control over 802.11 DCF
 Wireless LAN， ”
 Asia-Pacific Symposium on Information and Telecommunication Technologies (APSITT2008)，
 April 2008
4. 新井絵美，平野由美，村瀬勉，小口正人：
 “ 無線 LAN 環境におけるトランスポート層プロトコルの相違による帯域公平性の検討， ”
 マルチメディア、分散、協調とモバイル (DICOMO2008) シンポジウム，2008 年 7 月
5. 新井絵美，平野由美，村瀬勉，小口正人：
 “ 無線 LAN 環境における QoS 実現のための検討， ”
 並列分散工学ミニワークショップ，2008 年 9 月
6. 新井絵美，平野由美，村瀬勉，小口正人：
 “ 無線 LAN 環境における実機特有の帯域公平性についての検討と QoS 保証 TCP の性能評

価，”

データ工学と情報マネジメントに関するフォーラム（DEIM2009），2009年3月