

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

新井 絵美[†] 平野 由美^{††}
村瀬 勉^{††} 小口 正人[†]

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, これらの問題について実機を用いた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. 本論文では特に, トランスポート層プロトコルの相違が帯域公平性に及ぼす影響について評価する.

A Study on Fairness of Bandwidth with difference of transport layer protocol on a Wireless Environment

EMI ARAI,[†] YUMI HIRANO,^{††} TUTOMU MURASE^{††}
and MASATO OGUCHI[†]

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 these problems 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. Especially in this paper, we have evaluated how the difference of transport layer protocol affects fairness of bandwidth.

1. はじめに

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-AV¹⁾ 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

[†] お茶の水女子大学

Ochanomizu University

^{††} NEC システムプラットフォーム研究所

System Platforms Research Laboratories NEC Corporation

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。

本論文では特に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。この無線 LAN の公平性の問題を考えるためにいくつかのトランスポート層プロトコルを用いて実験を行う。

無線 LAN の公平性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の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。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。

以下、第 2 章にて無線環境における TCP フローのスループットの公平性について議論し、第 3 章でトランスポートプロトコルについて紹介し、第 4 章で UDP、TCP の評価を行い、第 5 章で TCP-AV の評価を行い、第 6 章で本稿のまとめを行う。

2. 無線環境における公平性

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 1 に示す。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。

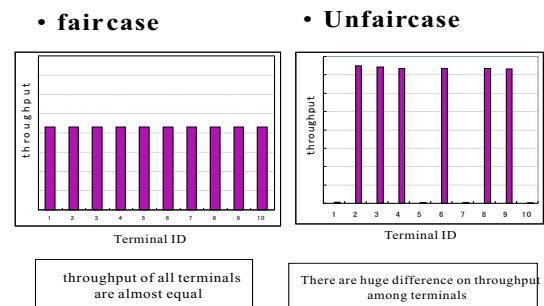


図 1 公平および不公平なスループット

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 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。

2.1.1 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。

2.1.2 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。以下でその問題を議論する。

2.2 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。

2.2.1 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く、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。

2.2.2 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 参照)。このとき、データの送信は高々 54Mbps 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。

2.3 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

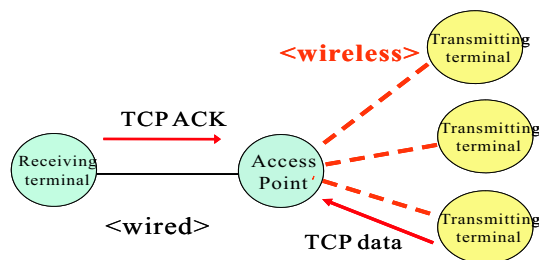


図 2 シミュレーションにおける TCP フローの不公平

では、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 3 に示す。

本稿では、USB タイプの無線子機として、BUFFALO WLI-UC-AG⁴⁾ 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-G54⁵⁾ と BUFFALO WLI-TX4-G54HP⁶⁾ を使用した。

3. トランスポート層プロトコル

本論文では無線 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。

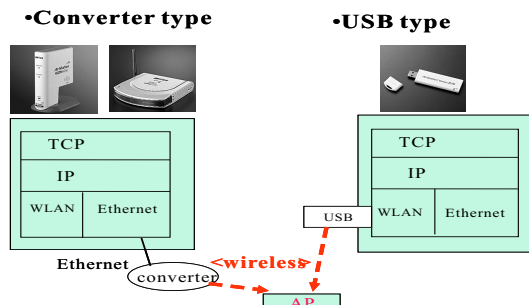


図 3 無線子機と MAC 層の違い

3.1 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。

3.2 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⁷⁾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移移し、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 4 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。

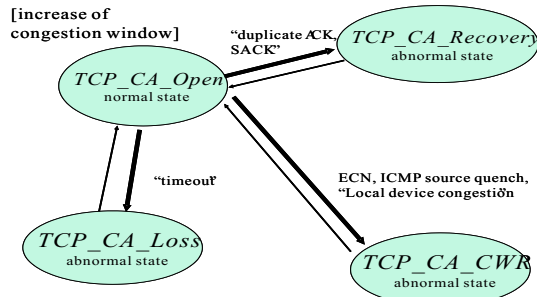


図 4 輻輳ウィンドウの状態変化

3.3 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。

4. 無線環境における各プロトコルごとの評価

本節では無線 LAN 環境における各プロトコルの公平性について調べる。そのためにまず、AP のバッファサイズを調べ、シミュレーション環境における不公平

の原因が実機においても適用できるか検証した。次に、それが ACK のロスであることを確認するために UDP を用いて実験を行った。さらに、TCP を用いて同様の実験を行い、スループットの公平性がどのような場合に問題となるか調べた。

また、本実験のために、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。また、本研究ではすべて無線 LAN として、IEEE802.11g⁸⁾ を使用した。

4.1 AP のバッファサイズの検証

4.1.1 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な、BUFFALO WZR-AMPG300NH⁹⁾ を使用した。図 5 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¹⁰⁾。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 5 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。

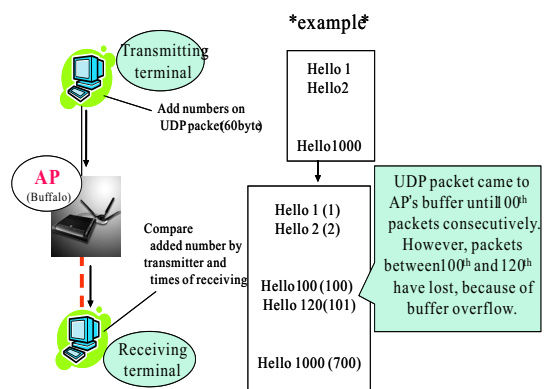


図 5 AP のバッファサイズ検証実験

4.1.2 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。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。

4.2 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。スループットの測定に Iperf⁽¹⁾ を用いて、UDP フローを流す。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。以下全ての実験で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。

4.2.1 基本性能

まず図 6 のように一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ため、両方のスループットを測定した。

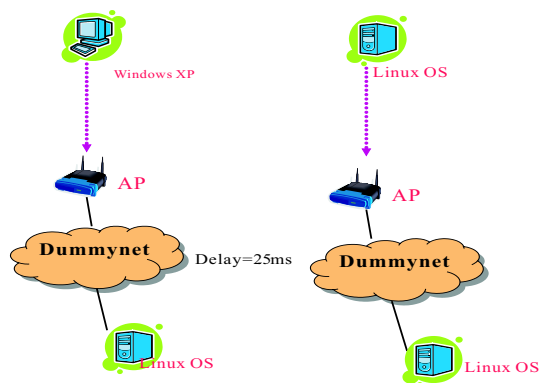


図 6 基本性能

表 1 フロー一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)

その結果が表 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、たくさん送り出すことができて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ないと考えられる。また、USB タイプは端末が無線と認識して、送り出しているため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。

4.2.2 同機種無線 LAN 環境

文献⁽²⁾ ではカード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。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。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 7 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 8 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、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。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

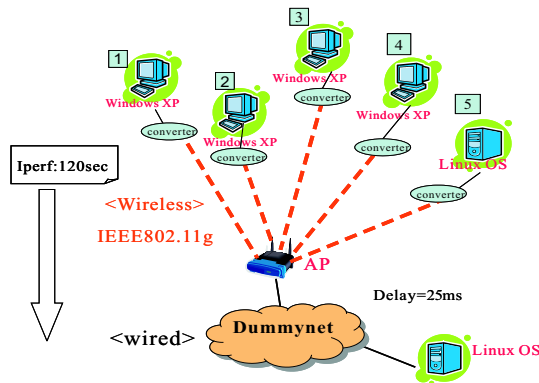


図 7 同機種無線 LAN 環境

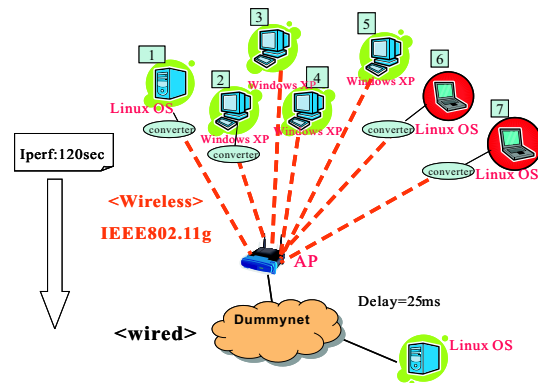


図 9 異機種無線 LAN 端末混在環境を用いた実験環境

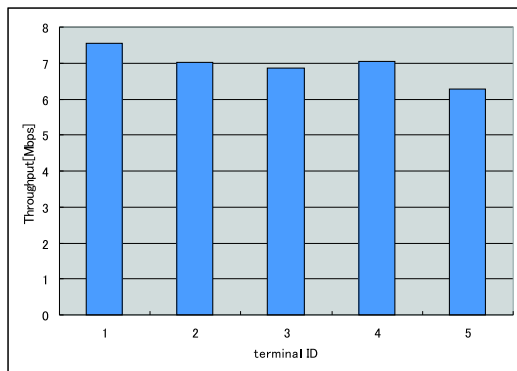


図 8 同機種無線 LAN 環境のスループット

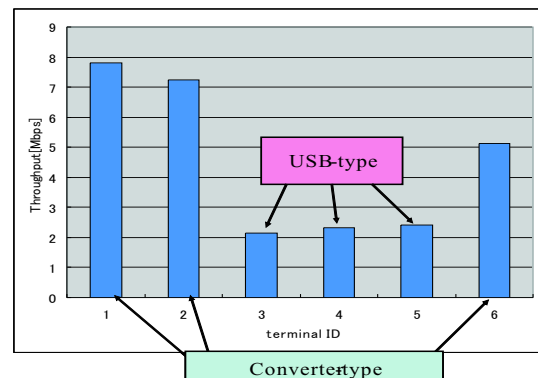


図 10 異機種無線 LAN 端末混在環境のスループット

ACK が関わっていることが考えられる。

4.2.3 異機種無線 LAN 端末混在環境

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。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 9 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 層における基本製の違いから 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。

このときのスループットを図 10 に示す。先ほどと同様、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 ÷ 7 台=約 3Mbps) 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、基本性能でフロー一本あたりときと同様、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。

4.3 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。まず、一般的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。さらに複数の無線 LAN 端末から有線の端末に TCP フローを流すことで不公平が示されているので、実際にイーサネットコンバータ無線 LAN のみを用いる環境、イーサネットコンバータ無線 LAN と USB 無線 LAN の混在環境で実験をおこない、公平性を調べる。スループットの測定には UDP のときと同様、Iperf120 秒間用いた。UDP と同様、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 ミリ秒または、片道遅延 20 ミリ秒、パケットロス 0.5%とした。

4.3.1 基本性能

まず図 11 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った。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,5 に Linux をそれぞれ OS として使用した。

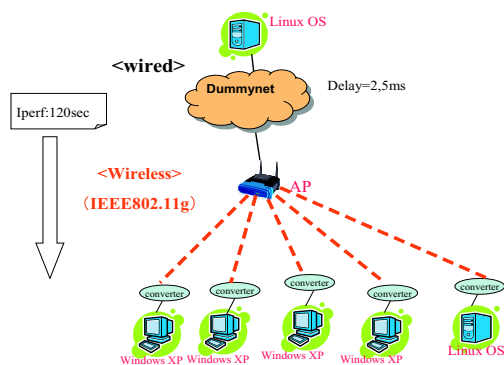


図 11 一般的な無線環境における TCP フローを流す実験

結果を図 12 に示す。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。

4.3.2 同機種無線 LAN 環境における TCP フローの公平性

次に、無線側の複数の送信端末から有線側の受信端末にデータを送り、不公平の問題を調べた。前述のよ

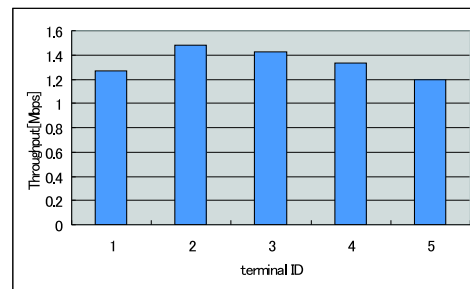


図 12 一般的な TCP フローを流す実験のスループット

うに、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 2～5 台と変えて実験を行った。実験環境は 4.2.2 の図 7 と同様である。ただし、図は送信端末の台数が 5 台の場合を表している。

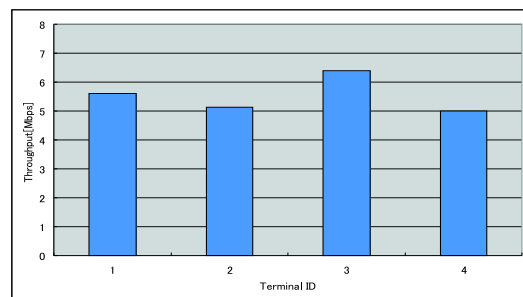


図 13 TCP フローが公平な場合（送信端末数 4）

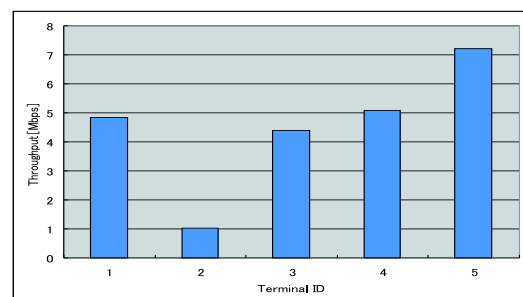


図 14 TCP フローが不公平な場合（送信端末数 5）

送信端末が4台のときと5台のときのスループットを図13と図14に示す。送信端末が2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台にしたときに1台不公平に陥る端末が現れた(図14のTerminal ID2)。すなわちイーサネットコンバータを用いた無線LAN通信では送信端末が5台のときに不公平が観測された。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は送り出す力が同じなので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APのバッファでのACK破棄が顕著となり、このような不公平が起きていると考えられる。

4.3.3 異機種無線LAN 端末混在環境

次に、UDPの時と同様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。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15に示す。前述の実験より端末台数を増やして、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¹²⁾。

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16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÷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1Mbps以下となっている。

次に、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17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

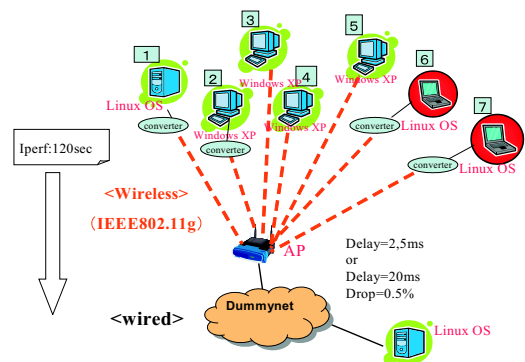


図15 異機種無線LAN 端末混在環境においてTCPフローを流す実験

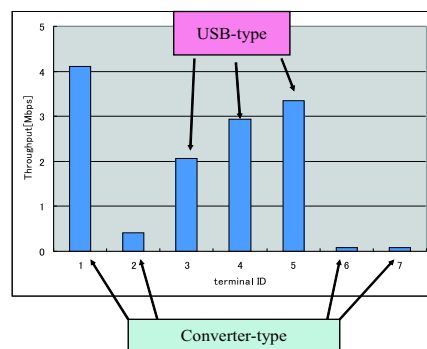


図16 異機種無線LAN 端末混在環境における公平性(遅延2.5ミリ秒)

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1Mbps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起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できるTCPモニタ¹³⁾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18に示す。

TCPモニタを用いたときの結果を図19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き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

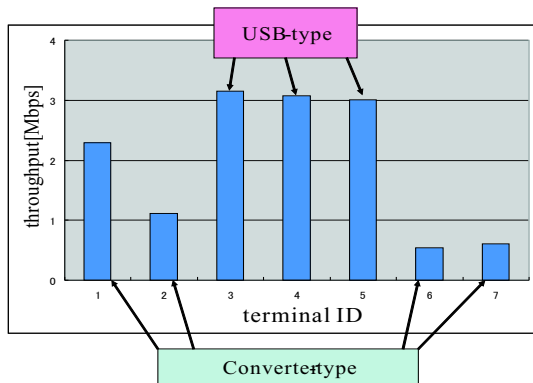


図 17 異機種無線 LAN 端末混在環境における公平性 (遅延 20 ミリ秒, パケットロス 0.5%)

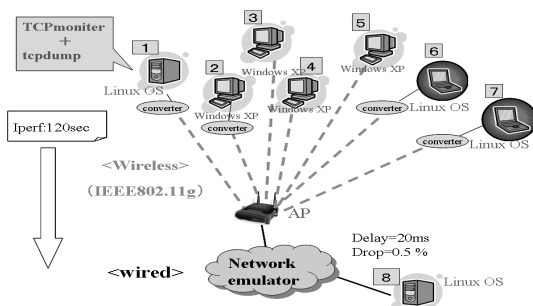


図 18 イーサネットコンバータの動作を解析する実験環境

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きていることを示す。

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 20 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 19、図 20 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。

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

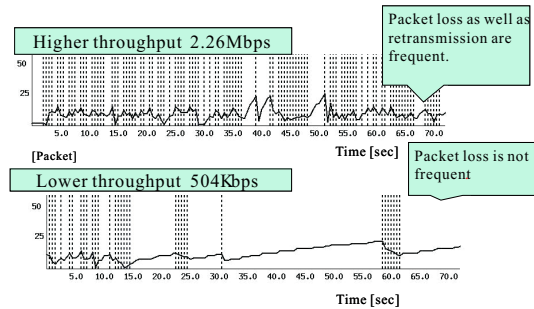


図 19 TCP モニタの実行結果

Result 2 : tcpdump (converter type) ~timing of transmitting and receiving~

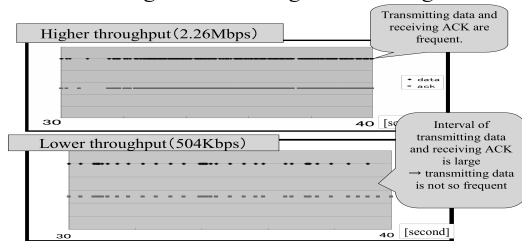


図 20 tcpdump の実行結果

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。つまり、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こらず、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。

5. TCP-AV の評価

5.1 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 21 に示す。ここで、Flow1 は TCP-AV のフローとするが、送信端末の TCP の実装は変更しない。その代わり 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、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。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。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。

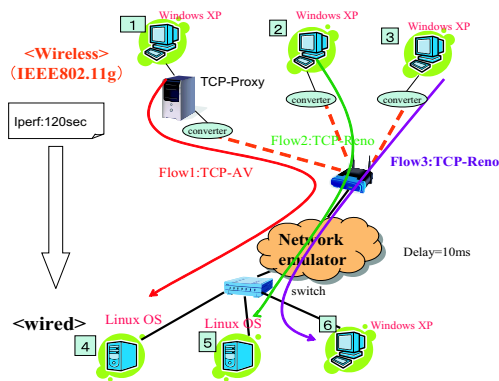


図 21 AVR を評価する実験

5.2 実験結果および考察

結果を図 22 の“3flows”に示した。横軸の 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 21 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 22 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。

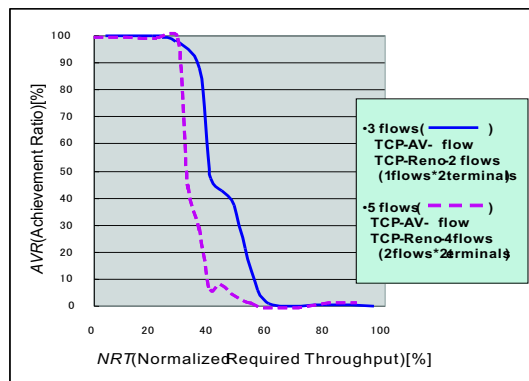


図 22 AVR による TCP-AV の評価 1 (送信端末数 3)

5.3 送信端末が増加した場合の実験概要

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 23 に示す。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。

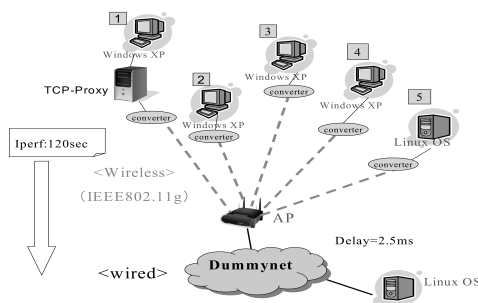


図 23 送信端末が増加した場合の AVR の評価

5.4 実験結果に関する考察

本実験における fair-share は NRT の 20%である 5Mbps である。結果を図 24 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。fair-share を超えて設定帯域を NRT の 25%とすると、AVR は 70%となる。さらに NRT の 40%を設定すると AVR は 1 割以下となる。

無線 LAN の送信権の平等性により、fair-share を大幅に超えて帯域を確保することはできなかったが、不安定な無線かつイーサネットコンバータを用いた環境においても TCP-AV を用いた端末が不公平な端末と

ならず, fair-share より 5%増ぐらいまでであれば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。

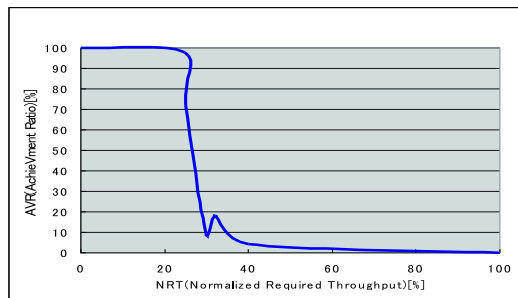


図 24 AVR による TCP-AV の評価 2 (端末台数 5)

6. まとめと今後の課題

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。具体的には, 2 種類のトランスポート層プロトコルを用いて実験を行い, いくつかの無線 LAN 環境におけるスループットの公平性について考察した。

まず 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。

次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公平と不公平の両方が起こることを観測した。UDP を用いた環境, および有線端末から複数の無線端末に TCP を用いて送信するという環境においては, ACK のロスが起こらないことから, 公平であることを示し, 複数の無線端末から有線端末に TCP を用いて送信するという環境においては端末の台数が増えたときに不公平が観測された。

さらに, 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っていた。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ある程度の QoS 保証を導入できたといえる。

今後は, さらに台数を増やした環境における公平性の検証, イーサネットコンバータタイプの挙動の詳しい分析, 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, 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評価を通して, 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

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氏と浜崇之氏に感謝致します。

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) 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
- 3) S.Pilosof, R.Ramjee, Y.Shavitt, P.Sinha, "Understanding TCP fairness over Wireless LAN " INFOCOM 2003, 1-3 April 2003.
- 4) BUFFALO AirStation 54Mbps 802.11 abg (WLI-UC-AG):
<http://buffalo.jp/products/catalog/network/wli-uc-ag/>
- 5) BUFFALO 2.4GH 54Mbps wireless LAN Ethernet conversion media converter (WLI3-TX1-G54) :
<http://buffalo.jp/products/catalog/item/w/wli3-tx1-g54/>
- 6) BUFFALO AirStation High Power wireless LAN Ethernet converter 54Mbps (WLI-TX4-G54HP) :
<http://buffalo.jp/products/catalog/item/w/wli-tx4-g54hp/>

- tx4-g54hp/
- 7) Linux Kerner ver.2.4 implementation:
<http://www.linux.org/>
 - 8) IEEE802.11 wireless LAN medium access control (MAC)and physical layer(PHY) specification,1999
 - 9) BUFFALO wireless LAN BB router AirStation NFINITI GIGA 300Mbps 11n conformity (WZR-AMPG300NH) :
<http://buffalo.jp/products/catalog/network/wzr-ampg30nh/>
 - 10) Perl program of inspect the size of buffer:
<http://x68000.q-e-d.net/68user/net/udp-2.html>
 - 11) Iperf: <http://dast.nlanr.net/Projects/Iperf/>
 - 12) 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
 - 13) 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
-