

無線 LAN 使用時のトランスポート層における 複数コネクション集約に関する評価

宮 崎 悦 子^{†1} 小 口 正 人^{†1}

近年、様々なモバイルネットワークアプリケーションの実行に必要なスループットはますます増加している。それを受けて IEEE 802.11, Bluetooth や CDMA などの様々な無線技術が開発されており、WAN, LAN ないし PAN 上のデータ通信に広く利用されている。しかし各々の帯域幅は未だに乏しいものが多く、また広帯域を確保している無線通信は使用可能なシーンが限定されているのが現状である。さらに現存する多くの技術は、同時に一つの無線技術を使用することに限られている。そこで提案されているのが適用範囲にある無線技術の帯域幅を複数同時に使用することにより大きいスループットを得ようとする BAG (Bandwidth Aggregation) である。しかし BAG を行う際に TCP の性質による性能の低下がみられることが分かっている。この問題を解決するためのネットワーク層の手法として EDPF や PET-BMP などが提案されているが、それらの手法も問題点の完全の解決には至っていない。そこで本研究では既存研究で明らかになっている問題点を解決し、よりすぐれた実装を行うことを目標として、モデルウェアを用いた複数コネクションの制御手法を提案し、無線シミュレーションによる実験を行った。

The Evaluation of Aggregateing Multiple Connection over Transport Layer using wireless LAN

ETSUKO MIYAZAKI^{†1} and MASATO OGUCHI^{†1}

Recently, higher throughput is required for executing various mobile network applications. Accordingly, variety of wireless technologies such as IEEE 802.11, Bluetooth, and CDMA are developed and widely used for data communications in WAN, LAN and PAN environments. However, each bandwidth is still low, and as for a wireless communication with wide bandwidth, the scene that can be used is limited. In addition, most research in this domain has been confined to single interface used at any time to meet the connectivity requirements of the mobile client. Bandwidth Aggregation (BAG) is proposed that obtains large throughput using the bandwidth of a multiple wireless connections simultaneously. However, when BAG is evaluated, the lower performance by the

characteristic of TCP is observed. Thus EDPF and PET-BMP are proposed as a technique at the network layer to solve this problem. However, the problem can not be solved completely even with these methods. The goal of this research is solve the problem clarified by an existing research is solved, and it is evaluated by a wireless simulation.

1. はじめに

モバイルインターネットの発展により、様々な無線通信技術が開発されている。例えば無線 LAN に用いる IEEE 802.11, 近距離無線通信の Bluetooth や携帯電話での通信に利用される CDMA (Code Division Multiple Access) などである。

これまでの研究により、一つのインタフェースから別のインタフェースへスムーズなハンドオフを行うことについては議論されているが¹⁾、同時に複数の無線技術を用いて通信することについては未だに実用化されていない。様々な無線技術が開発されている一方で、有線の場合と比べるとネットワーク各々の帯域幅は未だに乏しく、広帯域の無線は利用可能なシーンが限定されているのが現状である。同時に複数の無線技術の適用範囲がオーバーラップしたとき、ユーザが1つのインタフェースだけでなく複数のインタフェースを選択して利用できればこれらの制限から解放され、より充実したモバイルサービスを受けることが可能となる。

そこで提案されているのが適用範囲にある無線技術の帯域幅を複数同時に使用することにより大きいスループットを得ようとする BAG (Bandwidth Aggregation) である。また、複数の無線技術を同時に使用することで、帯域幅集約の他に可動性のサポート、信頼性の向上やリソース共有などのメリットを享受することができる。

2. 研究背景

同時に複数のインタフェースを用いて無線通信を行った場合、複数の性質が異なる無線インタフェースのパケットが混ざった状態に到着すると TCP からパケットが抜けてしまったように見え、不必要な輻輳ウィンドウ制御をおこなってしまう。これは BAG により得ることを期待されたスループットを下げてしまう原因となる。

^{†1} お茶の水女子大学
Ochanomizu University

そこで、TCP の性質を考慮し、見積もられた配達時間を元にパケットをスケジューリングするアルゴリズムとして EDPF (Earliest Delivery Path First) が提案されている²⁾。EDPF を用いることで、パケットロスがない状態であるとはほぼ理想通りの BAG の性能を発揮したスループットを得ることができるが、パケットロスが起こった際には一つの接続のスループットの低下が他の全ての接続のスループットも下げってしまう。

この問題を解決するために提案されている方式が PET (Packet-Pair based Earliest-Delivery-Path-First algorithm for TCP applications) と BMP (Buffer Management Policy)³⁾ である。PET は各々の経路でのパケットの配達時間を見積もり、最速で届けるための経路をスケジューリングすることでパケットの再要求を最小化する。BMP は PET で完全に防ぎきれなかった TCP からのパケットの再要求を隠し、順番がばらばらなパケットをネットワーク層で緩和して順番どおりに TCP に渡す役割を果たす。PET と BMP を用いた場合、EDPF と比べるとパケットロスが起こった際の複数接続のスループットの低下を防ぐことができているが、パケットロスが集中するとロスが分離できずに複数の接続のスループットが下がってしまうことが問題となっている。既存研究では、この問題に対して他の仕組みで無線のロスを減らすことができれば期待した性能を発揮することができるとしているが、現実には無線のロスを減らすことは大変困難である。そこで本研究では明らかになっている問題点の検証を行い、よりすぐれた実装を考案することで帯域幅集約の性能を上げるための手法を検討する。

3. 研究概要

本研究では既存研究で行われた実験をシミュレーションで実行することで明らかになっている問題点の解決方法を見つけ、パラメータの変更、新しいモデルの考案やアルゴリズムの改良などにより帯域幅集約の性能を上げることを目標とする

3.1 ベースとなる提案手法

既存研究ではネットワーク層におけるアプローチを用いていかに帯域幅集約の性能を上げるかを議論している。これによりトランスポート層へは接続がひとつであるように見せることができ、帯域を集約していることをアプリケーションが意識することなく通信可能であることがメリットとなっている。しかし同一の TCP プロトコルスタックを利用している限り、ある接続に起こった問題の影響が、他の接続に波及することを避けるのは困難である。

そこで本研究では経路ごとに独立した TCP の接続を持つ、図 1 に示される手法を提案する。TCP レベルにおける複数の接続をミドルウェアで統合することで、帯域を集約していることをアプリケーションが意識することなく通信可能であるという従来手法のメリットを残すことができる。さらにパケットロスが起こった場合に複数経路のスループットが下がってしまう原因となっている輻輳ウィンドウを経路ごとに持たせることで、提示されている問題点を回避する。

この手法は、既存の TCP を作り変えて、複数の接続をトランスポート層で 1 本に束ねるアプローチでも実現可能である。しかしミドルウェアで統合することを考えれば既存の TCP をそのまま利用して目的を達成することが可能であると考えられる。

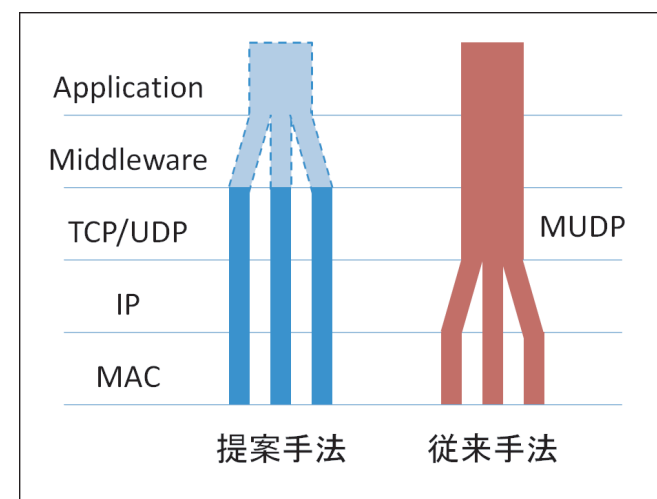


図 1 提案手法

3.2 QualNet

実験環境として無線ネットワークのシミュレーションソフトウェアである QualNet を使用する。QualNet は米国 Scalable Network Technologies 社⁴⁾ が販売しており高速かつ大規模な通信ネットワークシステムのシミュレーションが可能である。また物理層からアプリ

ケーション層までの通信プロトコルモデルをソースコードで提供しており、カスタマイズにより独自プロトコルを実装しシミュレーションすることができる。

QualNet のライセンスは Microvision 社の FLEXnet を用いたフローティングライセンス形式を用いており、サーバ用とクライアント用のライセンスファイルをそれぞれの端末に入れておき、クライアント端末がシミュレーションを実行するたびにサーバ端末のライセンスファイルへアクセスする。この仕組みを用いることでたとえば1つのライセンスしか持っていない場合でも、時間が重なっていなければ多数のマシンでシミュレーションを動かすことが可能である。

3.3 評価シナリオ

本実験では図2で示されるシナリオを実行した。node1, node2 は ifa0, ifa1, ifa2 の3つの無線インタフェースを持っており、伝送速度 11Mbps の IEEE 802.11b の無線規格で接続されている。node1 から node2 へ 10 秒間、512bytes のファイルを送信し続けるアプリケーションを動かした。

3.4 実験条件 1

node1, node2 間の通信で ifa0 のみを使用して通信した場合のスループットを観察した。

3.5 実験条件 2

node1, node2 間の通信で ifa0, ifa1, ifa2 の3つの無線インタフェースを使用して通信した場合のスループットを観察した。

4. 実験結果

実験条件1で1つの経路のみを使用した通信のスループットを UDP と、実験条件2で3つの経路を同時に使用した通信のスループットを MUUDP としグラフにしたものを図3へ示した。

3つの経路を同時に使った方が1つの経路のみを使用するより3倍スループットが良い値となった。

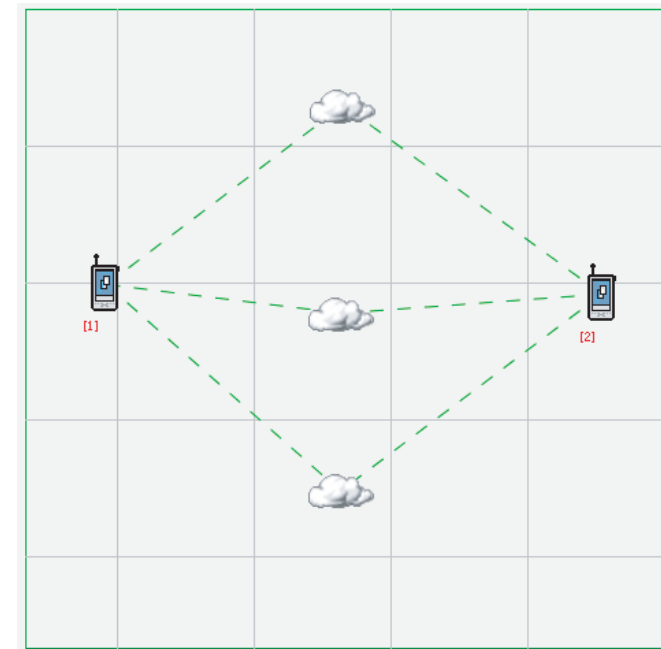


図2 評価シナリオ

5. まとめ

本研究では複数の無線インタフェースを同時に用いた通信に関する評価を行うためにシミュレーションによる実験を行った。

複数経路を用いて通信を行うと、1つの経路のみを用いて通信した場合と比較すると同時に使用した経路の本数倍だけスループットが高くなることを確認した。

6. 今後の課題

複数経路を用いて通信を行うと、1つの経路のみを用いて通信した場合と比較すると同時に使用した経路の本数倍だけスループットが高くなるということをトランスポート層のプロ

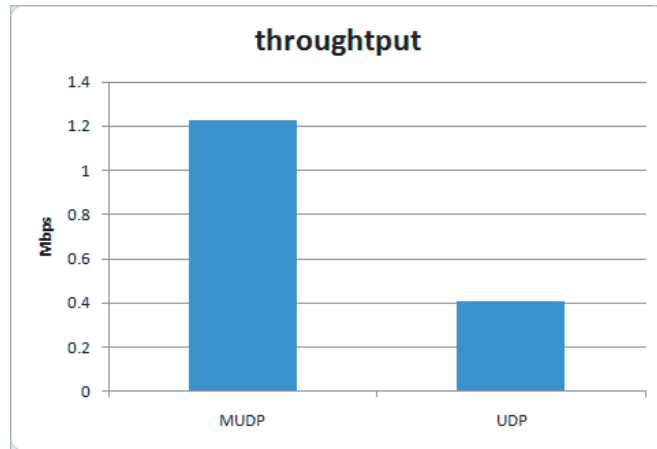


図 3 実験結果

トコルが UDP の場合で確認したので、今後は TCP を用いた場合でも同様の成果を発揮できるか確認する。

また、既存研究で指摘をされているような無線環境下での TCP のふるまいを回避したアルゴリズムを考案・実装する。

参 考 文 献

- 1) M. Stemm and R. Katz :“ Vertical handoffs in wireless overlay networks, "Mobile Networks and Applications Vol.3, No4, pp.335-350, Jan. 1998.
- 2) K. Chebrolu and B. Raman :“ Bandwidth Aggregation for Real-Time Applications in Heterogeneous Wireless Networks, " IEEE Transactions on Mobile Computing, Vol.5, No4, pp.388-403, April 2006.
- 3) K. Chebrolu, B. Raman, and R.R. Rao :“ A Network Layer Approach to Enable TCP over Multiple Interfaces,” J. Wireless Networks (WINET), Vol.11, No5, pp.637-650, Sept. 2005.
- 4) Scalable Network Technologies :
<http://www.scalable-networks.com/>