

複数無線インタフェース使用時の ミドルウェアにおけるバッファ量の評価

宮崎 悦子[†] 小口 正人[†]

Evaluation of Buffer Size for Middleware using Multiple Interface in Wireless Communication

Etsuko MIYAZAKI[†] and Masato OGUCHI[†]

あらまし 無線通信で利用できる帯域は有線での通信と比較してまだ乏しいものが多いのが現状である。そこで使用可能な無線通信を複数同時に使用する帯域幅集約が提案されているがトランスポート層より下の制御で集約を実現しようとするときパケットロスが起こった際に TCP の輻輳制御により必要以上に通信の品質が下がってしまうのが問題である。そこで本研究では帯域幅集約ミドルウェアを提案し、その設計に必要な受信側のバッファサイズについての評価を行った。

キーワード 複数帯域, ミドルウェア, 帯域幅集約, バッファサイズ, IEEE 802.11

1. はじめに

モバイルネットワークの発展により、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 IEEE802.11、近距離無線通信の Bluetooth や無線 MAN の WiMAX などである。しかしこれらの無線技術は広帯域なものであっても、Ethernet などの有線での通信と比較するとネットワークそれぞれの帯域幅が未だに乏しいものが多い。また、比較的広帯域を確保している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。そこで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つの無線技術だけでなく複数の無線技術を選択して利用できれば帯域の制限が緩和され、より充実したモバイルサービスを受けることが可能になると考えられる。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でより大きいスループットを得ようとする帯域幅集約 (Bandwidth Aggregation) である。

これまでの研究により、1つのインタフェースから別のインタフェースへの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。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受けることができると考えられる。

2. 研究背景

2.1 各層における帯域幅集約手法

帯域幅集約は様々な層において実現することが考えられるが、それぞれに利点と欠点が存在する。

データリンク層でのアプローチ [2] では、効率よく帯域を集約することができ、それより上位の層が集約していることを意識せずに済むのが最大の利点である。しかし、同じデータリンク層のプロトコルを使用しているネットワークの範囲内ではしか使用できず、すべてのノードに専用のハードウェアを導入しなければならないという欠点を持つ。

ネットワーク層での制御では、手法によって専用機器も必要なく帯域幅集約を実現することができる [5]。上位層に現在広く使用されている TCP・UDP に対して透過的に振る舞うことができるのが利点であるが、トランスポート層のプロトコルとして TCP を用いて集約を行った場合、本来の順序とは異なる順序で到

[†] お茶の水女子大学
Ochanomizu University

着するパケットが多くなる可能性が高い。それにより TCP は不必要な再転送を要求し、全ての接続のスループットを下げってしまうことが欠点となっている [6].

トランスポート層で集約を行うと輻輳制御や再転送制御を経路ごとに持たせることができる [3]. これにより各経路に応じた効率的なパケットの分配や再転送を行うことができるので効率のよい通信を実現可能であるが、エンド間でのオペレーティングシステム内の実装が不可欠である。

アプリケーション層での実装では 1 つのエンドノードのみへ集約の制御を加えることで実現できる [4]. 現行のオペレーティングシステム内を書き換える必要も無いということも大きな利点である。しかし様々な種類のアプリケーションが存在する中で、それぞれに適した集約の実装を行うのは大変困難である。また経路の数だけトランスポート層プロトコルの接続を張ってもどの接続への程度トラフィックを分散させるかを見積もることが重要となる。

2.2 ネットワーク層における帯域幅集約

性質の異なる無線インタフェースを複数同時に用いて通信を行った場合、異なる経路からのパケットは送信された順序以外でパケットが到着する可能性がある。パケット受信側の TCP は、次に期待していたパケットより後のパケットを先に受信してしまうと、パケットが抜けたと認識してしまう。それにより TCP は本来不要であるはずの再転送を要求し、期待した集約結果を得ることができない。

そこで提案されているのが EDPF(Earliest Delivery Path First) [5] である。EDPF は送信側からパケットを送信する際、経路が分岐するノードにおいて実装される。分岐ノードは分岐する先の経路の帯域幅や遅延、使用率を元に、パケットがどの経路を通れば最短で受信側ノードへ届くかを計算する。パケットは常に最短で届くと見積もられた経路へ配信されるので受信側 TCP は常に送信側で送信された順にパケットを受け取ることができる。この手法により、パケットロスの無い環境では期待通りの帯域幅集約の性能を発揮できることが確認されている。

2.3 ネットワーク層における問題解決のアプローチ

無線環境で通信を行った場合、有線の場合と比較してパケットロスが多く起こることが知られている。ネットワーク層以下で帯域幅集約を行う際にパケットロスが起こると、TCP は特定の経路のパケットロスであ

ることを認識できず輻輳制御を行ってしまう。その結果ウィンドウサイズが下がるのでスループットを下げることとなり、期待した帯域幅集約の性能を得ることができない。

この問題を解決するために提案されているネットワーク層の帯域幅集約方式が PET (Packet-Pair based Earliest-Delivery-Path-First algorithm for TCP applications) と BMP (Buffer Management Policy) である [6]. PET は EDPF で行っていた経路の見積もりを、より厳格に、動的に行う。BMP はパケット受信側のノードへ実装され、受信したパケットを並び替えが必要なものか、それともパケットロスなのかを判断する役割を果たす。次に到着する筈のパケットのシーケンス番号より大きいシーケンス番号のパケットがすべての経路から到着すれば、それは確かにパケットロスであることを BMP が TCP へ報告する。そう出ない場合には、順番どおりに並び替えたパケットを TCP へ渡す役割を果たす。

PET と BMP を実装した場合、EDPF を用いた場合と比較すると、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。しかし、パケットロスが集中した場合、複数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、ほか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、現実には無線通信でのパケットロスを減らすことは大変困難である。

3. 提案手法

3.1 概要

先に述べたようにトランスポート層以下やアプリケーションにおいて帯域幅集約を行う際には様々な制約や問題が存在する。そこで本研究ではトランスポート層とアプリケーション層の間にミドルウェアとして帯域を集約する層を新たに挿入することでより効率の良い帯域幅集約を行う手法を提案する。ネットワーク層での帯域幅集約と提案手法を比較したネットワーク各層の概念図を図 1 に示した。

提案手法では経路ごとに独立した TCP の接続を持ち、それら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

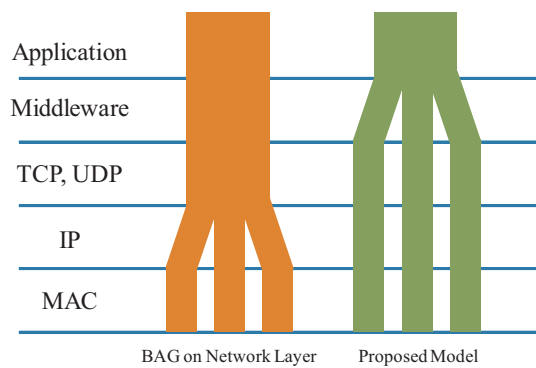


図1 提案手法

いうトランスポート層以下での制御で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、既存研究で指摘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、複数のコネクション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。

この方式を用いることにより、パケットロスが起こる環境でネットワーク層における帯域幅集約を行った場合の問題点は解決できるということを既存研究で確認した。[7]

3.2 構造

送信側の提案手法ミドルウェアは接続可能な経路すべてへそれぞれTCPコネクションを張る。アプリケーションからデータを受け取ると、配信順の番号をデータへ付与し、配信可能なコネクションへデータを送信していく。受信側のミドルウェアはそれぞれのTCPコネクションから受信されるデータを、配信時に付加された番号順に並び替えてからアプリケーションへ渡す役割をはたす。

このとき受信側のミドルウェアへは配信順にパケットが到着しない可能性がある。次に期待した番号を持ったパケットが到着するまで到着したデータを保存するために、受信側のミドルウェアはバッファを持つ必要がある。このバッファの大きさは、どの環境でどの程度必要になるかを見積もることは、ミドルウェア設計の上で重要な項目のひとつである。

4. シミュレーションソフトウェアでの受信側バッファサイズの評価

本実験では帯域幅集約を行う際の明らかになっている問題の解決のために新しいモデルを考案することにより、帯域幅集約の性能を上げることを目標とする。実験で使用したのは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である。提案手法のモデルを設計するために必要な受信側ミドルウェアのバッファサイズを調べるために、様々な状況での実験を行なった。

4.1 全体的に低帯域幅の回線におけるシナリオ

図2に示されるような2本の無線経路を利用できるモバイル端末（ノード2）がノード1から送信されたデータを2本の経路を同時に使用して受信するシナリオを実行した。これをシナリオ1と呼ぶ。

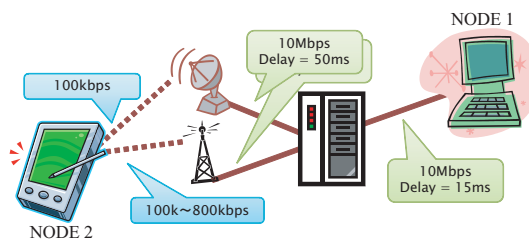


図2 シナリオ1（全体的に低帯域幅の回線を持つ環境）

有線部分の帯域幅は10Mbpsで遅延は図のとおりであり、無線部分の帯域幅は片方を100kbpsに固定し、もう片方を100~800kbpsまで100kbpsずつ変化させ、二つの無線の幅の比が1:1から1:8になるように設定して実験した。使用したトランスポート層のプロトコルはTCP new Renoでパラメータの値は表1のように設定した。

表1 TCP パラメータ

MSS	1460Bytes
Send buffer	65535Bytes
Receive buffer	65535Bytes

4.2 全体的に高帯域幅の回線におけるシナリオ

図3に示されるような2本の無線経路を利用できるモバイル端末（ノード2）がノード1から送信されたデータを2本の経路を同時に使用して受信するシナリオを実行した。これをシナリオ2と呼ぶ。

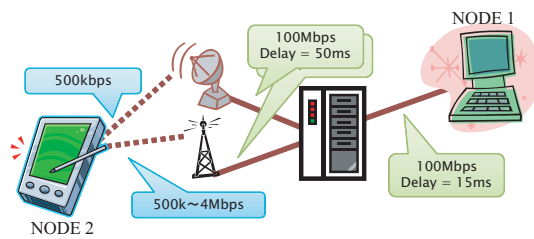


図3 シナリオ2（全体的に高帯域幅の回線を持つ環境）

有線部分の帯域幅は100Mbpsで遅延は図のとおりであり、無線部分の帯域幅は片方を500kbpsに固定し、もう片方を500k~4Mbpsまで500kbpsずつ変化させ、二つの無線の幅の比が1:1から1:8になるように設定して実験をした。使用したトランスポート層のプロトコルはTCP new Renoでパラメータの値はシナリオ1と同様に設定した。

4.3 様々な状況での実験

さらに図4と図5に示されるような2本の無線経路を利用できるモバイル端末（ノード2）がノード1から送信されたデータを2本の経路を同時に使用して受信するシナリオを実行した。

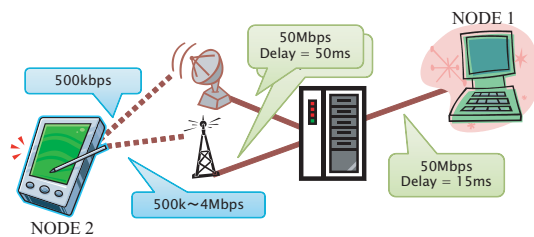


図4 シナリオ3（無線部分の帯域幅のみ広い環境）

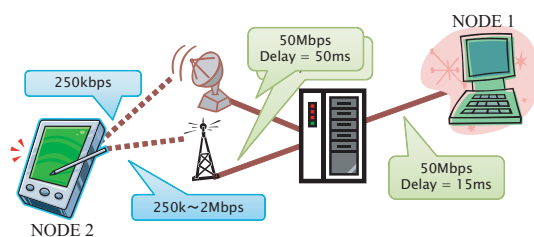


図5 シナリオ4（シナリオ4と比較して無線部分が狭い帯域幅を持つ環境）

シナリオ1やシナリオ2と同様に無線部分の帯域幅は片方を一定の値に固定し、もう片方を一定の値ずつ

変化させ、二つの無線の幅の比が1:1から1:8になるように設定して実験をした。使用したトランスポート層のプロトコルはTCP new Renoでパラメータの値はシナリオ1と同様に設定した。

4.4 定常状態と定常状態になるまでの時間

シナリオ1で無線の帯域幅が100kbpsと300kbpsでの実行時のそれぞれの経路のスループットと受信側ミドルウェアのバッファの大きさを図6へ表した。通信が開始されしばらく経つと、2本の無線のスループットはそれぞれ約100kbpsと300kbps程度の速度で通信しており、効率の良い通信を行えていることがわかる。受信側ミドルウェアのバッファのサイズは初めは小さいものが徐々に大きくなり、一定の大きさに安定することが分かった。このバッファのサイズが安定した状態を定常状態と呼び、定常状態の際のバッファの大きさと、定常状態になるまでの時間、2つの値に注目した。

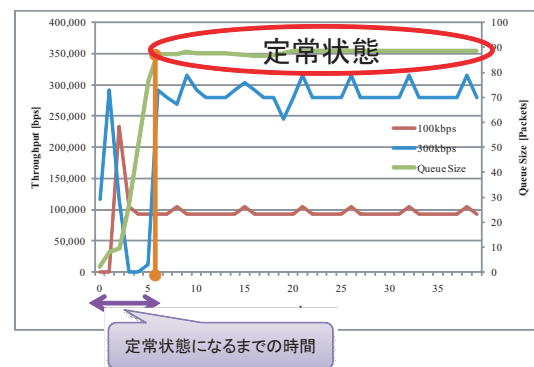


図6 定常状態と定常状態になるまでの時間

4.5 帯域幅の比とバッファサイズの関係

シナリオ1で2つの無線の幅を変化させた際の定常状態のバッファサイズがいくらか示したグラフを図7へ表した。

無線の幅が同じ時にはゼロであった定常状態のキューの大きさが片方の無線の幅を広くするとそれに比例して大きくなっていくことが分かった。

シナリオ2で2つの無線の幅を変化させた際の定常状態のバッファサイズがいくらか示したグラフを図8へ表した。

シナリオ1のときと同様に片方の無線の幅が大きくなるにしたがってそれに比例してキューの大きさも大きくなっていることがわかった。さらに、シナリオ1

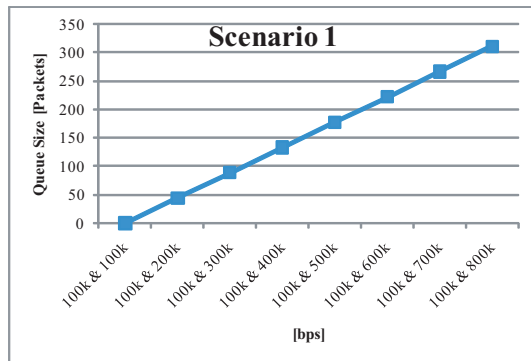


図 7 シナリオ 1 での帯域幅とバッファサイズの関係

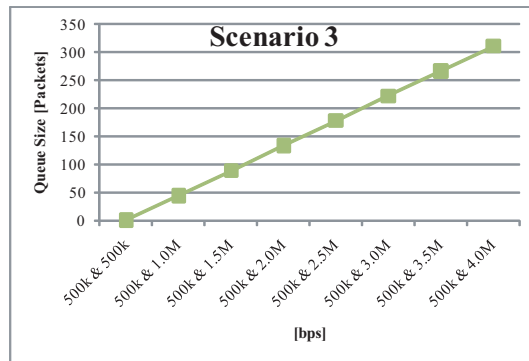


図 9 シナリオ 3 での帯域幅とバッファサイズの関係

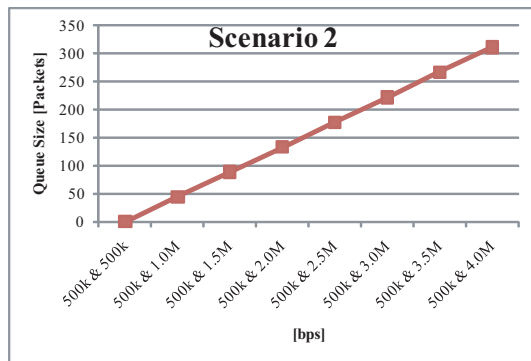


図 8 シナリオ 2 での帯域幅とバッファサイズの関係

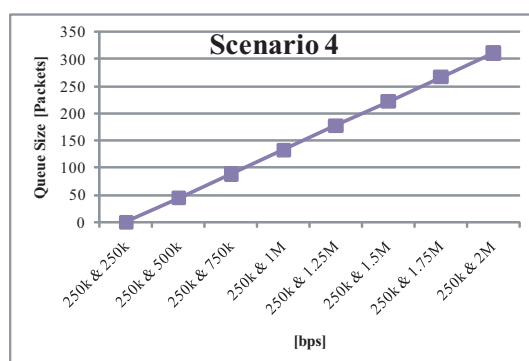


図 10 シナリオ 4 での帯域幅とバッファサイズの関係

とシナリオ 2 では有線部分や無線部分の速度、有線と無線の速度の比が違うにも関わらず、2つの無線部分の速度の比が同じだと、キューにたまるパケットの数も完全に同じであることが分かった。

さらにシナリオ 3 とシナリオ 4 での 2つの無線の幅を変化させた際の定常状態のバッファサイズがいくらになるか示したグラフをそれぞれ図 9 と図 10 へ表した。

4.6 帯域幅の比と定常状態になるまでの時間の関係

すべてのシナリオにおける 2つの経路の幅の比を変化させた場合の定常状態になるまでの時間を調べたグラフを図 11 へ示した。

全体的に速度の遅い経路を使うシナリオ 1 と全体的に速度の速い経路を使うシナリオ 2 を比較すると、全体を通してシナリオ 2 の方が定常状態になるまでの時間が短いことが分かった。また、2本の経路の幅の比

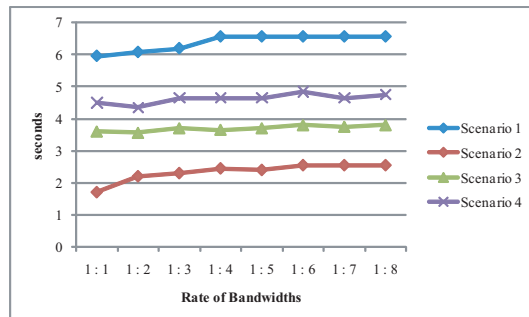


図 11 帯域幅の比と定常状態になるまでの時間の関係

が変化しても、すべてのシナリオにおいてそれぞれ定常状態になるまでの時間はあまり変化しないことが分かった。

5. 結論と今後の課題

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ネットワークシミュレータを用いて実験を行った。既存研究で提案されて

いる、ネットワーク層で帯域幅集約を行う手法についての検証を行い、パケットロスが発生するとすべてのインタフェースのスループットが下がってしまうという問題点を解決するためにミドルウェアにおける帯域幅集約を提案した。既存研究と同様に、それぞれの経路を使って通信を行ったスループットを束ねた値と同程度の性能を発揮することを確認した。

受信側のミドルウェアへ到着したデータは順番どおり並び替えてからアプリケーションへ渡される必要があるため、ある程度のバッファを持つことが想定される。そのバッファがどのような環境でどれだけ必要になるかを推測するために2本の無線経路を持つモバイルノードの経路の幅を変化させた際のバッファの大きさを調べた。2本の幅の比が大きくなると、それに比例して必要なキューの大きさも大きくなることが分かった。

今後は、結論により明らかになった、経路の幅の比に応じて受信側バッファに必要な大きさのキューを用意する機能をミドルウェアに持たせ、また、より効率のよい通信を行うために送信側のミドルウェアにおいてパケットの配信順を制御する機能を検討する。本実験では同時に使用可能な無線インタフェースが2本であると仮定したが3本以上ある場合のミドルウェアについても検討する。さらにパケットロスが多数起こる環境や、下位層のプロトコルが異なる場合、有線や無線部分の帯域幅が様々に変化した場合のシナリオを実行し、より適応性の高い帯域幅集約を実現する。

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] IEEE P802.3ad Link Aggregation Task Force : <http://grouper.ieee.org/groups/802/3/ad/>
- [3] Junwen Lai Ming Zhang and et al : "A transportlayer approach for improving end-to-end performance and robustness using redundant paths." *USENIX 2004 Annual Technical Conference*, pages 99-112, 2004.
- [4] H. Nozawa, N. Honda, K. Sakakibara, J. Nakazawa, and H. Tokuda: ARMS: "Application-level Concurrent Multipath Utilization on Reliable Communication" *Internet Conference* 2008, Oct. 2008.
- [5] 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.
- [6] 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.

- [7] E. Miyazaki, O. Altintas, and M. Oguchi : "複数無線インタフェース使用時のミドルウェアにおける接続集約に関する検討" *DICOMO* 2010, July 2010.
- [8] Scalable Network Technologies : <http://www.scalable-networks.com/>