

無線マルチホップネットワークにおける プロキシ型TCPのウィンドウサイズ設定の性能評価

山本 祐太[†] 伊藤 暢彦[†] 山本 幹^{††}

[†] 関西大学大学院工学研究科 〒564-8680 大阪府吹田市山手町 3-3-35

^{††} 関西大学システム理工学部 〒564-8680 大阪府吹田市山手町 3-3-35

E-mail: [†]sa8m238@edu.kansai-u.ac.jp, ^{††}yama-m@ipcku.kansai-u.ac.jp

あらまし 無線マルチホップネットワークでTCPを用いる場合の問題点として、伝送段数が増加するとスループットが低下するという問題がある。この問題を解決する手法としてプロキシ型TCPが提案されている。しかし、すでに著者らの研究によってプロキシ型TCPを使用しても、TCPによるセグメントのバースト送信が原因で、期待されたほどスループットが向上しないことがわかっている。本稿では、セグメントのバースト送信を回避するためにTCPが管理する広告ウィンドウサイズを小さな値に設定する。すなわち、プロキシホスト、宛先ホストが上流ホストに対して広告するウィンドウサイズを調整する。シミュレーションによる評価から、プロキシホストが送信ホストに対して広告するウィンドウサイズを小さな値に保つことが、セグメントのバースト送信を抑制し、スループット向上に効果的であることを示す。

キーワード 無線マルチホップネットワーク, TCP, プロキシ型TCP, ウィンドウサイズ

Performance Evaluation of Window Size in Proxy-based TCP for Multi-hop Wireless Networks

Yuta YAMAMOTO[†], Nobuhiko ITOH[†], and Miki YAMAMOTO^{††}

[†] Graduate School of Engineering, Kansai University 3-3-35 Yamate-cho, Suita, Osaka, 564-8680 Japan

^{††} Faculty of Engineering Science, Kansai University 3-3-35 Yamate-cho, Suita, Osaka, 564-8680 Japan

E-mail: [†]sa8m238@edu.kansai-u.ac.jp, ^{††}yama-m@ipcku.kansai-u.ac.jp

Abstract In multihop wireless networks, a TCP session can obtain very low throughput with increase of the number of hops. One promising way to resolve this technical problem of TCP throughput degradation is proxy-based TCP. In proxy-based TCP, an end-to-end TCP session is divided by (an) intermediate proxy host(s), which leads to reduction of RTT of each divided session and improvement of TCP throughput. However, our current studies show that proxy-based TCP surprisingly does not bring performance improvement and obtain lower throughput than conventional end-to-end TCP. The reason for this degradation is bulk transmission of segments by sender and proxy host. In this paper, we introduce intentional flow control method which prevents bulk transmission of segments. In intentional flow control method, proxy host and destination host intentionally returns given advertised window size. Performance evaluation results show that keeping a small advertised window size that proxy host returns effectively decreases bulk transmission of segments.

Key words Wireless multihop networks, TCP, Proxy-based TCP, Window size

1. ま え が き

無線マルチホップネットワークでTCPを用いた場合、送信ホスト-宛先ホスト間のホップ数が増加するにしたがって、TCPスループットが低下するという問題がある^[1]。これはMACプ

ロトコル IEEE802.11DCF の RTS/CTS/ACK フレーム交換による遅延などによりRTTが増加することが原因で発生する。この問題を解決する手法としてTCPコネクション上にプロキシを配置する方式が提案されている^[2]。プロキシは上流側(送信ホスト, 上流プロキシ)に対しては宛先ホストとして動作し、

下流側(宛先ホスト, 下流プロキシ)に対しては送信ホストとして動作する。このため, 送信ホスト-宛先ホスト間の接続をプロキシで分断した短い接続に分割することで, 多段伝送によるスループットの低下を軽減することが期待される。

しかし, すでに著者らの研究^[3]により, プロキシホストのバッファ容量が有限であるという現実的環境においては, プロキシ型 TCP の性能は向上するどころか, かえってスループット特性が劣化することが明らかになっている。この理由はセグメントのバースト的な送信に起因する。送信ホストとプロキシホストがセグメントをウィンドウベースでバースト的に送信することで, プロキシホスト周辺で無線チャネルの干渉が発生し, 前段セッション(送信ホスト-プロキシホスト間), 後段セッション(プロキシホスト-宛先ホスト間)の通信がそれぞれ停止する。すなわち, この問題は, 本来の TCP 接続をプロキシを用いて分割していることで前段セッションと後段セッションがそれぞれ独立に TCP の制御機構を動作させ, 前段, 後段セッションが非同期的にセグメントのバースト送信を行うことで発生する。

このセグメントのバースト送信を回避する方法として, TCP が管理しているウィンドウサイズを小さい値に設定するという手法がある^[4]。本来, TCP の送信レートは, 送信ホスト自身が管理している輻輳ウィンドウサイズと, 宛先ホストから受け取る広告ウィンドウサイズに基づいて, ウィンドウベースで決定している。そのため, 宛先ホストが送信ホストに対して広告するウィンドウサイズ(本来は宛先ホストの空きバッファサイズが広告される)を意図的に小さい値に設定することで, 必然的に TCP が開くことのできるウィンドウサイズ(輻輳ウィンドウサイズと広告ウィンドウサイズの最小値)を小さくすることができ, セグメントのバースト送信を抑制することができる。

本稿では, このセグメントのバースト送信を解決する手法のひとつである広告ウィンドウサイズの調整に着目し, これをプロキシ型 TCP に応用する。プロキシホストのバッファ容量が有限であるという現実的環境において, プロキシホスト, 宛先ホストが上流ホスト(送信ホスト, プロキシホスト)に広告するウィンドウサイズを変化させ, シミュレーションによる評価を行う。評価の結果, プロキシホストが送信ホストに対して広告するウィンドウサイズを小さな値に保つことによって, セグメントのバースト送信が抑制され, スループット向上に効果的であることを明らかにし, その要因を無線チャネルの競合の観点から考察する。

2. 現実的環境におけるプロキシ型 TCP

2.1 プロキシ型 TCP

伝送段数の増加に伴う RTT の増加を抑制するために, 送信ホスト-宛先ホスト間を複数の接続に分割する, プロキシ型 TCP が提案されている。図 1 にプロキシ型 TCP の動作を示す。

プロキシホストは送信ホストが宛先ホストに向けて送信したセグメントを宛先ホストに代わり受信し, ローカルな受信確認応答として LACK(Local-ACK) を送信ホストに返信する。こ

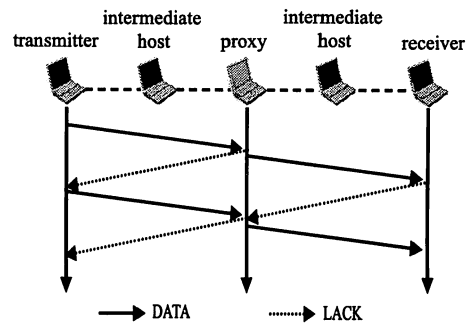


図 1 プロキシ型 TCP の動作

の際に, プロキシホストは宛先ホストと同様に自身のバッファ空き容量を広告ウィンドウとして LACK に付加し, 送信ホスト-プロキシホスト間のフロー制御を行う。プロキシホストは受信したセグメントをいったんトランスポート層バッファに蓄積する。

プロキシホストは, 送信ホストとは別に独自管理している輻輳ウィンドウと後段のプロキシホスト(もしくは宛先ホスト)からの広告ウィンドウをもとに, 次段セッションへのセグメント送出を制御している。具体的には, 後段のプロキシホスト(もしくは宛先ホスト)から到着する LACK により, このセッションの輻輳ウィンドウと広告ウィンドウを更新し, 送出可能であればトランスポート層バッファのセグメント(ウィンドウ制御により送出可能な先頭セグメント)を次段へと送出する。なお, プロキシホストが自らのトランスポート層バッファに蓄積しているセグメントは, 送信ホストのバッファと同様に該当する LACK が到着するまで破棄することはできない。このプロキシホストに設けられるトランスポート層バッファは, 前段の分割セッションと後段の分割セッションにより共有されているものとする。

このような動作により, プロキシ型 TCP ではエンドツーエンドセッションを, 送信ホスト, プロキシホスト, 宛先ホストで構成される複数の分割セッションに分ける。このことにより, 各分割セッションの RTT は小さく抑えてそのセッションのスループットをエンドツーエンド TCP の場合のスループットより高く維持し, 全体として送信ホスト-宛先ホスト間のスループット向上が期待できる。

2.2 有限バッファ環境におけるプロキシ型 TCP

しかし, すでに著者らの研究^[3]により, 多段伝送によるスループットの低下を軽減することが期待されるプロキシ型 TCP は, プロキシのバッファ容量が有限であるという現実的環境においては, 性能が向上するどころか, かえって性能が劣化することが明らかになっている。図 2 にプロキシ型 TCP のある区間でのウィンドウサイズの推移を示す。図中の $cwnd(source)$, $cwnd(proxy)$, $awnd(source)$ はそれぞれ, 送信ホストの輻輳ウィンドウサイズ, プロキシホストの輻輳ウィンドウサイズ, 送信ホストがプロキシホストから受け取った広告ウィンドウサ

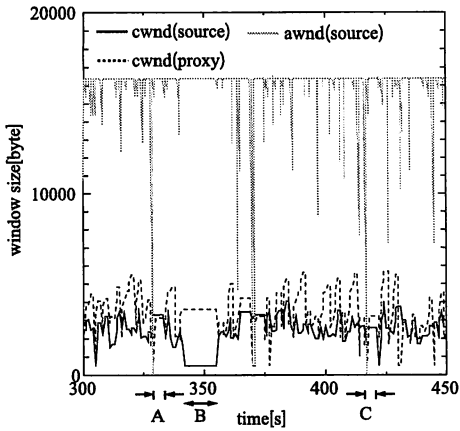


図2 プロキシ型 TCP のウィンドウ推移

イズを表している。文献 [3] ではプロキシのバッファ容量が有限であるという現実的環境においてプロキシ型 TCP のスループットが劣化する原因を、プロキシホスト周辺の無線チャネル干渉によるものとし、スループットが低下する 2 つの現象を説明している。

1 つはプロキシホスト周辺の無線チャネル干渉により、後段セッションがバースト的にフレーム送出できた場合に生じる現象である (図 2 における B の区間)。この場合、後段セッションが連続して無線チャネルをキャプチャし、その間前段セッションがフレームを送出できない。このため、前段セッションにおいてタイムアウトが連続して発生し、長期間前段セッションからの送出が停止する。前段セッションからプロキシホストへのセグメント到着が一時的に停止することから、結果的に後段セッションのフレーム送出も停止する。図 2 においては、区間 B において前段セッションの輻輳ウィンドウサイズが 1 となり、タイムアウトによる送信が停止していることを示している。また、プロキシ側の輻輳ウィンドウが変化していないことから、プロキシからのセグメント送出も停止していることが分かる。2 つめは同干渉により前段セッションがバースト的にフレーム送出できた場合に生じる現象である (図 2 における A, C の区間)。この場合、後段セッションからの送出が停止しプロキシホスト内のバッファが一杯になり、広告ウィンドウサイズ 0 の LACK が返送されて、前段セッションの送出が停止する。前段セッションからのセグメント送出が一時的に停止することで、プロキシへのセグメント到着が停止し、結果的にある期間後段セッションも同時に停止するという現象である。図 2 の区間 A, C では、前段セッションの広告ウィンドウが 0 にまで低下したのち、前段セッションならびに後段セッションの双方において一時的に輻輳ウィンドウの変化が停止しており、両セッションにおいてセグメント送出が一時的に停止している様子がみれる。

これら 2 つの現象はいずれも、送信ホストとプロキシホストが TCP の制御機構であるウィンドウベースのフロー制御に基づいてセグメントをバースト的に送信することに起因する。送信ホスト、プロキシホストがそれぞれ独立にセグメントをバースト

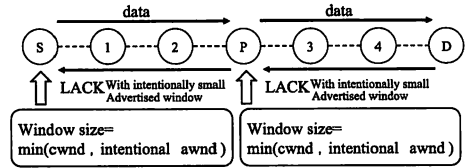


図3 強制フロー制御

ト送信することにより、前段、後段セッションのどちらかが無線チャネルをキャプチャし、前段セッションと後段セッションの通信がうまく流れず通信停止となる時間区間 (図 2 における A, B, C の区間) が発生し、低スループットとなる。

3. 強制フロー制御

前章で述べたように、プロキシ型 TCP は、TCP のウィンドウベースのフロー制御によって発生するセグメントのバースト送信が原因で性能が劣化する。セグメントのバースト送信による性能劣化は、プロキシ型 TCP だけでなく、もちろん従来の TCP でも同様に発生する。このセグメントのバースト送信を抑制する方法として、TCP が管理しているウィンドウサイズを小さな値に設定するという手法がある^[4]。TCP ではセグメント送信を行う際は、送信側自身が管理している輻輳ウィンドウと相手ホスト (宛先ホスト) から受け取る広告ウィンドウの最小値から送信レートを決定している。また TCP は送信レートをウィンドウベースで決定しているため、宛先ホストが送信ホストに対して広告するウィンドウサイズを意図的に小さい値に設定することで、必然的に送信レートを低く抑えることができ、セグメントのバースト送信を抑制することができる。

本稿ではセグメントのバースト送信を回避するために上記で説明した手法をプロキシ型 TCP に応用する。すなわちプロキシホスト、宛先ホストが上流ホスト (送信ホスト、プロキシホスト) に広告するウィンドウサイズを調整する。この動作を図 3 に示す。これにより、フロー制御に意図的な制限を加えることができ、セグメントのバースト送信を抑制することが可能となる。プロキシ型 TCP においても、TCP の場合と同様に、この手法を用いて広告ウィンドウサイズを小さな値に設定することでセグメントのバースト送信を抑制可能であることが予想されるが、本稿では広告ウィンドウサイズとプロキシ型 TCP の性能特性に着目して調査を行うために、プロキシホスト、宛先ホストが上流ホスト (送信ホスト、プロキシホスト) に広告するウィンドウサイズを、比較的小さい値 (2MSS など) だけではなく、1~32MSS の広範囲でそれぞれ変化させて性能評価を行う。

4. 性能評価

4.1 シミュレーションモデル

図 4 にシミュレーションモデルを示す。S は送信ホスト、P はプロキシホスト、D は宛先ホストを表している。本稿では、プロキシ型 TCP を用いた場合の基本特性を調査するため図 4 のような単一タンデムモデルを使用した。文献 [2] において 3~5 ホップ毎にプロキシを配置することが推奨されているため、

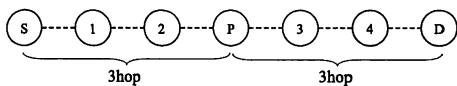


図4 シミュレーションモデル

S-P, P-D 間をそれぞれ 3 ホップとし、全体を 6 ホップのモデルとした。

シミュレーション条件を表 1 に示す。プロキシホストではトランスポート層における送受信バッファが共有バッファとして管理されていると仮定し、その容量を 32 セグメントとした。トラフィック発生モデルは常に送信データが存在する Gleedy モデルとし、シミュレータとして Qualnet3.9.5 を用いた。各ホストの無線帯域は 2Mbps とし、MAC プロトコルには IEEE802.11DCF を使用した。またルーティングプロトコルには AODV を用いた。TCP の MSS は 512byte とした。

TCP version	Reno
MSS	512 [byte]
Routing protocol	AODV ^[5]
MAC protocol	IEEE802.11DCF
Wireless band	2[Mbps]

本稿では、広告ウィンドウサイズとプロキシ型 TCP の性能特性に着目して評価を行うことを目的としている。そこで、本シミュレーションでは無線マルチホップネットワークにおけるプロキシ型 TCP のスループット特性を、プロキシホスト、宛先ホストが上流ホスト (送信ホスト、プロキシホスト) に対して広告するウィンドウサイズを変化させて評価する。広告ウィンドウサイズの調整はプロキシホスト、宛先ホストが上流ホストに広告するウィンドウサイズの上限值を変化させることで行う。例えば、広告ウィンドウサイズを 2(MSS) に設定した場合、TCP のバッファの空きが 2MSS よりも大きければ広告ウィンドウサイズは 2MSS となる。

また本稿では無線チャネルの競合の評価指標として次式 (1) で表される C を用いる。

$$C = \frac{\text{Number of RTS Sent}}{\text{Number of Frame Sent}} \quad (1)$$

Number of RTS Sent は RTS フレームの送出回数を表しており、Number of Frame Sent はデータフレームの送出回数を表している。無線チャネル内でフレーム競合が頻発すると、無線チャネル予約のために RTS フレームの送出が増加し、C が増加する。すなわち、C 値は 1 以上であり、C 値が増加するにしたがって、無線チャネルのフレーム競合が頻発しているといえる。

4.2 スループット評価

図 5 にプロキシホスト、宛先ホストが上流ホスト (送信ホスト、プロキシホスト) に対して返信する広告ウィンドウサイズに対するスループット特性を示す。Source AW と Proxy AW はそれぞれプロキシホスト、宛先ホストが上流ホスト (送信ホ

スト、プロキシホスト) に広告するウィンドウサイズ (以下、前者を前段 AW、後者を後段 AW と呼ぶ) を表している。図 5 より、広告ウィンドウサイズの変化に対してスループットが大きく変動していることがわかる。また、スループット特性は送信ホストが受け取る広告ウィンドウサイズ (前段 AW) が 2MSS のときに高スループットとなっている。しかし、一般的には TCP の広告ウィンドウサイズは 32MSS より大きい。すなわち、通常の TCP と比較してかなり小さい広告ウィンドウサイズに設定することで高スループットとなっていることがわかる。

図 5 のスループット特性について、さらに詳しく調査するために、図 6 に宛先ホストがプロキシホストに広告するウィンドウサイズ (後段 AW) を 2MSS に設定したときのスループット特性を示す。この図は図 5 における Proxy AW を 2MSS に設定した場合の断面図を表している。この図から Source AW が 2MSS のときスループットが最大となっており、Source AW が増加するにしたがってスループットが低下していることがわかる。この理由には無線チャネルの競合が関係していると考えられる。広告ウィンドウサイズが小さい値のときは、バースト送信が抑えられ、無線チャネルの競合が減少し高スループットとなると考えられる。この理由に対する詳しい考察は次章で説明する。また図 6 では Source AW が 1MSS の場合は 2MSS の場合と比較して低スループットとなっているが、これは広告ウィンドウサイズが 1MSS の場合だと、ネットワーク容量に対して負荷が小さいことが考えられる。

つぎに、後段 AW とスループット特性の関係について詳しく調査するために、図 7 に前段 AW を 2MSS に設定したときのスループット特性を示す。この図は図 5 における Source AW を 2MSS に設定した場合の断面図を表している。図 7 より、Proxy AW が 2MSS から 32MSS の範囲では図 6 と比較してスループットはあまり変化しておらず、高スループットを維持していることがわかる。Proxy AW が 1MSS のときにスループットが低下しているのは、ネットワーク容量に対して負荷が小さいことに加えて、後段セッションの方が広告ウィンドウサイズが小さいことでプロキシホストにセグメントが滞留し、セグメントがうまく流れていないためだと考えられる。

図 5 から 7 を合わせて考察することでスループット特性に

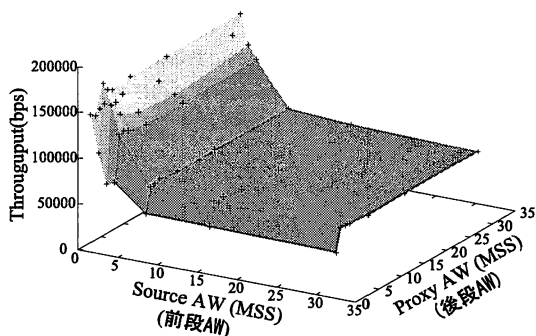


図 5 スループット特性

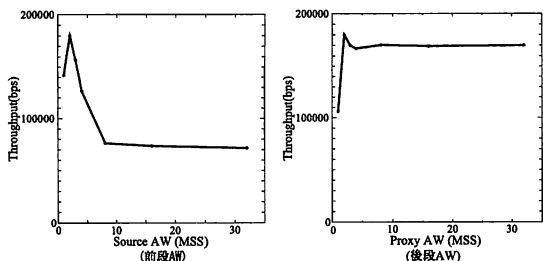


図6 スループット特性
(Proxy AW=2)

図7 スループット特性
(Source AW=2)

は以下のような特徴があることがわかる。スループットは前段 AW と後段 AW を 2MSS のような小さな値に設定することで向上することがわかる。またスループット特性は前段 AW の影響を強く受けており、後段 AW の変化に対してはほとんど変化していないことがわかる。この理由として、前段 AW を小さい値に設定することでセグメントのバースト送信を効果的に抑制されていることが考えられるが、これについては次節で詳しく考察する。

4.3 無線チャネルの競合

前節では、広告ウィンドウサイズに対するスループット特性を示したが、このスループット特性は無線チャネル競合と関係が深いと考えられる。そこで本節では、無線チャネル競合の観点からスループット特性を考察する。

スループット特性と無線チャネルの競合の関係について調査するために、図8に各ホストにおける無線チャネルの競合について示す。縦軸は前章の式(1)で導入したC、横軸は各ホストを表している。Host1, Host4, Host7はそれぞれ送信ホスト、プロキシホスト、宛先ホストを表している。なお、縦軸のC値は、TCPによるデータフローと、それとは逆方向に流れるACKフローを考慮した値である。そのため、Host7(宛先ホスト)においてもACKフローがあるため、C値は存在する。また図8では、(前段AW, 後段AW)=(1,1),(2,2),(2,32),(32,2),(32,32)の5つのパターンにおける無線チャネルの競合について示す。例えば、(2,32)の場合は、前段AWを2MSS、後段AWを32MSSに設定した場合を表している。

図8より、前段AWと後段AWが増加するにしたがってC値が増加していることから、無線チャネルの競合が頻発していることがわかる。この理由は前段AWと後段AWが増加することで、送信ホストとプロキシホストでセグメントがバースト送信されるためだと考えられる。つまり、広告ウィンドウサイズの増加によりネットワーク内に滞留するセグメントが増加するにあわせて、このバースト送信により無線チャネル上で競合する頻度が増加する。無線チャネル内でフレーム競合が発生すると、再度チャネル予約を行うためRTS/CTSフレーム交換を行うことによる遅延や、バックオフによる遅延などによりRTTが増加し、スループットが低下する。このことは図8の(32,32)の場合に、送信ホスト(Host1)とプロキシホスト(Host4)でC

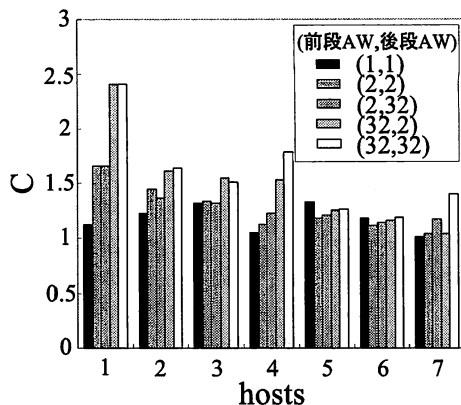


図8 無線チャネルの競合

値が比較的大きいことからわかる。

つぎに、スループット低下の要因である送信ホストによるセグメントのバースト送信について考察するために、図8のHost1(送信ホスト)での無線チャネル競合に注目する。送信ホストでの無線チャネル競合(C値)は、(2,2),(2,32)の場合ではほとんど同じ値となっており、また(32,2),(32,32)場合でもほぼ同じ値となっている。このことから、無線チャネルの競合は、当然ではあるが、後段AWとは関係なく、前段AWの影響が大きいことがわかる。また、前段AWを小さい値に設定する(2,2),(2,32)の場合は、前段セッションにおけるC値のみならず後段セッションにおいてもC値が比較的小さくなっており、前段AWを大きい値に設定する(32,2),(32,32)の場合では両者においてC値が大きい。これは、無線チャネル競合を抑制するには、前段AWを小さい値に設定すれば十分であり、後段AWについては影響がほとんどないことを示している。例えば、図8より、(2,32)の場合では、前段AWを小さい値に設定することによって、後段AWを大きい値に設定しても、効果的に無線チャネル競合が減少している。逆に、(32,2)の場合のように、後段AWを小さい値に設定しても、前段AWを大きい値に設定すると、無線チャネル競合の抑制には効果がないことがわかる。

また図8より、前段AWと後段AWが1MSSのときはCの値は1~1.5の範囲にあり、他の場合と比較してあまり無線チャネルの競合が発生していないことがわかる。しかし、図5から7のスループット特性では、前段AWと後段AWを1MSSに設定した場合に低スループットとなっていることがわかる。この結果からは、上で示した無線チャネル競合の増加によるスループット低下のメカニズムと矛盾しているように思われるが、これは以下のように考察できる。すなわち、(1,1)の場合ではネットワーク容量に対して広告ウィンドウサイズが小さすぎるためだと考えられる。前段AWと後段AWが1MSSの場合では、コネクションがネットワーク容量を全て使うことができず、無線チャネルの競合はあまり発生していないにもかかわらず低スループットとなっていると思われる。このことから、広告ウィンドウサイズは小さい値に設定することでスループット向上が期待できるが、広告ウィンドウサイズを小さな値に設定するこ

とがスループット特性に対して必ずしも良いのではなく、トポロジーのネットワーク容量を考慮する必要があることがわかる。

5. ま と め

本稿では、無線マルチホップネットワークでの TCP スループットを改善する手法として期待される、プロキシ型 TCP に対し、プロキシホストと宛先ホストが上流ホストに対して広告するウィンドウサイズの上限值を変化させ、シミュレーションによる評価を行った。その結果、送信ホストが受け取る広告ウィンドウサイズが増加すると無線チャネルの競合が増加し、スループットが低下することを明らかにした。またプロキシ型 TCP が低スループットとなる原因であるセグメントのバースト送信を解消するには、プロキシホストが送信ホストに広告するウィンドウサイズを小さな値に設定することが効果的であることを明らかにした。今後は、スループット特性とウィンドウサイズの関係についてさらに詳しい検討を行うため、コネクション内にプロキシホストを複数個配置した場合の評価や、複数セッションが存在する場合におけるスループット特性や、セッション間の公平性などの観点から評価を行う予定である。

文 献

- [1] J. Li, C. Blake, D. S. J. D. Couto, H. Lee and R. Morris, "Capacity of Ad Hoc Wireless Networks," *ACM MOBICOM '01*, pp.61-69, July 2001.
- [2] S. Kopparty, S. V. Krishnamurthy, M. Faloutsos and S. K. Tripathi, "Split TCP for Mobile Ad Hoc Networks," in *Proc. Global Telecommunications Conference, IEEE GLOBECOM '02*, Nov. 2002.
- [3] 伊藤 暢彦, 今西 友樹, 山本 幹, "アドホックネットワークにおけるプロキシ型 TCP の性能評価," 信学技報 *NS2007-103*, 2007.
- [4] Z. Fu, P. Zerfos, H. Luo, S. Lu, L. Zhang, and M. Gerla, "The Impact of Multihop Wireless Channel on TCP Throughput and Loss," *IEEE INFOCOM 03*, 2003.
- [5] C. E. Perkins, E. M. B. Royer and S. R. Das, "Ad Hoc On Demand Distance Vector Routing," *draft-ietf-manet-aodv-13.txt*, 2003.
- [6] 山本 幹, 松田 崇弘, "無線 TCP の研究動向 (特別講演)," 信学技報, *MoMuC2003-58*, Nov. 2003.
- [7] J. Stepanet, M.Y. Sanadidi and M. Gerla, "Efficient Buffer Management for TCP Service," *IEEE GLOBECOM '05*, 2005.
- [8] S. Xu and T. Saadawi, "Does the IEEE802.11 MAC Protocol Work Well in Multihop Wireless Ad Hoc Networks?," *IEEE Communications Magazine*, vol. 39, no. 6, pp. 130-137, June. 2001.