

マルチパスにおける TCP Spoofing の検証

柳生理子†
Riko Yagiu齋藤正史†
Masashi Saito

1. 背景

近年、様々な無線および有線の回線上で TCP を用いた高速・高信頼のサービスが増加すると共に、通信インフラの整備・高速化が進んでいる。しかし、こうした回線の中には高速の通信網である一方で、伝送遅延やエラー発生率が大きく、Transmission Control Protocol(TCP)上のアプリケーションを用いた通信において、TCP の輻輳制御機構の過剰な制御により、速度性能の劣化を引き起こすという短所を持つデータリンクがある。たとえば、光通信回線を用いた高速・大容量の回線を経由し、ファイルを送信する場合に、通常の FTP を用いると、速度性能はバンド幅の数パーセント未満にまで落ちてしまう。

この問題に対し、我々は、TCP の輻輳制御機構の緩和を促す TCP Spoofing (TCP-SPF) アルゴリズムの研究開発を行っている。このアルゴリズムは通信系路上に設置するゲートウェイ (GW) に適用することを前提とするものであり、ユーザ側のアプリケーションを変更することなく速度性能劣化の改善を可能とする。

本稿において、TCP-SPF 技術の応用として、遅延区間において、直列に配置する複数の GW を経由するマルチパス回線への TCP-SPF の適用についての提案と、評価実験を通して効果の検証を行う。

2. 高遅延回線における課題と TCP-SPF アルゴリズム

2.1 高遅延回線における高速・高信頼化に関する課題

近年、急速に普及しつつある光通信ケーブルは高速大容量の通信網である一方で、物理的伝送距離の長さゆえの伝送遅延がある。この遅延は TCP 上のアプリケーションを使用する際に、致命的な影響を及ぼす [1]。これは回線の遅延と、TCP の輻輳制御機構との間の、以下に示す二つの相互作用によるものである。

- 回線の伝送遅延の影響により、TCP 送信側においてセグメントが最終宛先に到達したかどうかを判別するまでに、長い時間を要し、セグメントの再送を促す
- TCP の輻輳制御機構は、「セグメントの再送の頻発は、回線の輻輳の信号である」との解釈に基づき、輻輳を緩和させるために送信量を抑制する

この遅延と TCP の輻輳制御との問題に対し、様々な技術の開発や提案が行われてきている。例えば、TCP を用いず UDP 上に構築された、遅延の影響を受けない高信頼のアプリケーションを用いるもの [2]、衛星通信向けの輻輳制御を緩和した TCP などが挙げられる。しかし、これらの方法は、PC(パーソナルコンピュータ)およびインターネットの普及により、容量の大きな PC 上の一般的なアプリケーションを用いた通信の速度性能向上に対しては有効ではない。

2.1 TCP-SPF アルゴリズム

上述の課題のために、我々は、通信系路上に Spoofing と呼ばれる特殊な機能を備えたルータを導入するアプローチ

を採り、この Spoofing 機能を実現する S/W (TCP Spoofing S/W) のアルゴリズムの研究開発を行ってきた[3][4]。図 1 は、TCP-SPF アルゴリズムを用いた仕組みと、その特徴および働きを示したものである。Spoofing とは、図 1 に示すように遅延区間の手前に設置されたゲートウェイ (SGW) において、パケットを転送すると共に、宛先 TCP において転送パケットを受信した場合に作成する送達確認応答

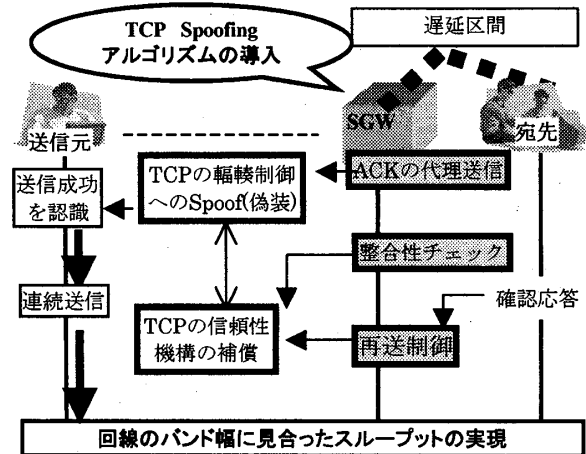


図1 TCP-SPF の概要

(ACK) と同等のもの (擬似 ACK) を作成し、送信側 TCP に対し、返送 (ACK の代理送信) する機能である。この ACK の代理送信機能により、送信元の TCP は宛先にまでセグメントが到達したと Spoof (偽装) され、連続的なデータ送信が可能となる。また SGW において TCP の端点間での通信結果の整合性チェックおよび再送制御を行い、擬似 ACK 送信による TCP 信頼性の保障を行っている。この手法は通信クライアントの S/W への変更が必要なく、ユーザから処理を隠蔽することから、一般的な TCP 上のアプリケーションを用いた大容量データに対する高速通信を可能とするものである。

3. マルチパスへの TCP-SPF の適用と効果

3.1 マルチパスへの TCP-SPF の適用

従来の TCP-SPF 技術は、図 1 に示すように、区間を、遅延のあるなしで 2 分割するものが主流であった。この手法は 1 本の専用回線のように特性が一定している場合は効果的であるが、端点間の経路の細かな特性や回線状況に応じた Spoofing の細かな制御がしづらという課題があった。これに対し、今回、回線をゲートウェイ (あるいはルータ) を挟んだ複数の区間 (セグメント) に分割し、それぞれのセグメントにおいて TCP-SPF を適用するアプローチを提案する。このアプローチは、各セグメントの輻輳状態、回線の容量、距離(遅延/RTT)などに応じて、それぞれの TCP Spoofing を適応するか否かを決定し ON/OFF を設定する仕組みを設けることにより、よりきめ細かな、TCP Spoofing の制御をおこない、回線の有効利用を目的とする。

図2は、マルチパスへのTCP-SPFの適用例である。図に示すように送信サーバから受信サーバに至る回線は、大容量で且つ高遅延の回線であり、ルート上に、ルータ1からルータnが設置され、それぞれにTCP-SPFが適用されている。図において、送信サーバから受信サーバに向けてTCP上のアプリケーションを用いてデータが配信されると、ルータ1は、受信サーバへ向けてデータパケットを転送すると共に、受信サーバが作成するものと同じ内容のTCPの擬似ACKを送信サーバへ返信する。このとき、ルータ1は受信サーバからACKを受け取るまで対応するデータを保持し、必要に応じて再送を行う。一方、ルータ1からデータを受け取ったルータ2は、ルータ1と同様に、受信サーバへ向けてデータパケットを転送すると共に、受信サーバが作成するものと同じ内容のTCPの擬似ACKをルータ1へ返信し、かつ受信サーバからACKを受け取るまで対応するデータを保持し、必要に応じて再送を行う。以後、

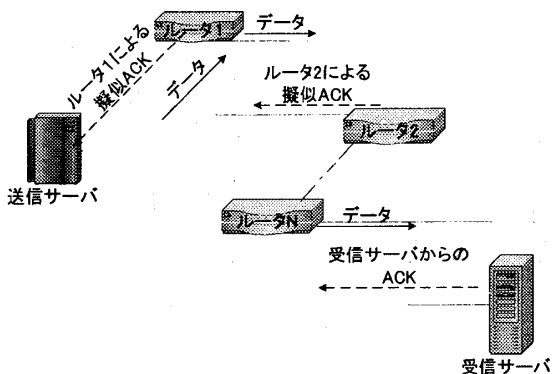


図2 マルチパスへのTCP-SPFの適用例

ルータNまで、遅延区間をN分割しTCP-SPFを適用し、TCPの速度性能の劣化の改善を行う。このN分割により、個々の遅延を短縮しかつ、各セグメントの輻輳状態、回線の容量、距離(遅延/RTT)などに応じて、それぞれのTCP Spoofingを適応するか否かを決定する仕組みを設ける。結果として、よりきめ細かな、TCP Spoofingの使用ができ、回線の有効利用が可能となる。

3.2 シミュレーションによる効果の検証

上述の遅延の分割とTCP-SPFアルゴリズムの適用の効果

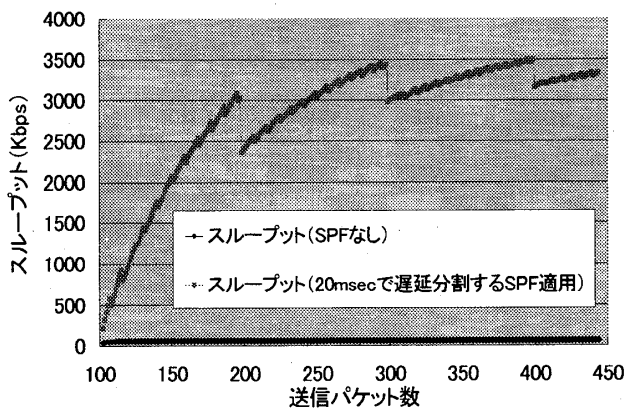


図3 シミュレーションによる性能比較(10M回線)

について、グアムを経由した日米(サンフランシスコ)間の光通信ケーブルをモデルとしたシミュレーションによる検証を行った。シミュレーションは、回線の帯域幅を10Gbps、全区間における総遅延を80msec、日-グアム間の遅延を20msec、グアム-米間の遅延を60msecに分割し、TCP-SPFを適用すると想定した。図3は、この検証の前に10Mの回線を想定した予備検証を行った結果である。図に示す通り、TCP-SPFを適用しない場合は、帯域の数パーセント未満のスループットであったものが、適用により30%以上の性能まで上がっていることが分かる。次に、同じ検証を実際の光通信回線と同等の10G回線を想定して行った。結果を図4に示す。図に示すとおり10G回線にて同じアルゴリズムを適用したところ、一度スループットは上がったものの、ピークを機に、徐々に下がり最終的には、効

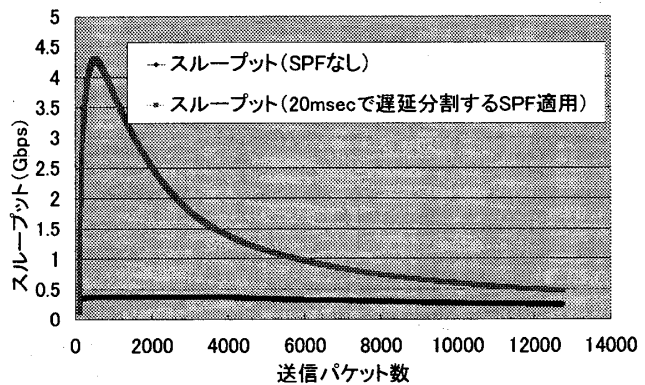


図4 シミュレーションによる性能比較(10G)

果がさほどない状況にまで落ちている。これは急激なスループットにより受信バッファのオーバーフローが起きており、バッファ容量を考慮したアルゴリズムの改良が必要であることを示す。

4. まとめと今後の課題

TCP-SPF技術の応用として、遅延区間において、直列に配置する複数のGWによる遅延の分割とTCP-SPFの適用についての提案と、評価実験を通して効果の検証を行った。検証の結果、提案する手法は10Mの回線では効果を示す一方で、さらに高速な回線においてはアルゴリズムの改良が必要なことが分かった。今後、アルゴリズムを改良し、遅延分割を含めたネットワーク設定のチューニングについても検討を行ってゆく予定である。

Reference

- [1] M.Allman:Enhancing TCP Over Satellite Channels using Standard Mechanisms, Request For Comments:2488(1999).
- [2] 柳生他：衛星利用共有型情報配信システムの設計と実装，情処学会マルチメディア通信と分散処理研究会 No.96,2000
- [3] 柳生他：高伝送遅延回線における速度性能劣化の改善，電子情報通信学会 2001年 総合大会, B-8-14
- [4] 柳生他：TCP輻輳制御緩和技術の開発と検証，情処学会マルチメディア通信と分散処理研究会, No.106,2001