

SCTP におけるプライマリパスの切替に関する研究

光家 弘城[†] 劉 鑫壯[†] 趙 悦[†] 森田 圭介[†]

概要 : 近年, ストリーミング配信などの実時間性を伴う通信が増加している. このような通信を行うアプリケーションは一定以上の通信品質の確保を必要としている. しかし, TCP や UDP のようなトランスポート層プロトコルは 1 対 1 の通信を前提として設計されているため, アプリケーションによる複数の接続の管理を行わない限り, ネットワークの状態の悪化や切断に対応することは難しい. これらの問題に対応するには, 端末が複数の IP アドレスを用いた通信を可能にするマルチホーム機能を持つトランスポート層プロトコルを利用する必要がある. これらの問題の対策としてマルチホーム機能を持つ SCTP に注目した. SCTP は複数の通信経路の中からデータ転送に用いるプライマリパスと呼ばれる通信経路を選出し, それ以外をオルタネイティブパスと呼ばれるバックアップ用とする. 通常の SCTP ではプライマリパスは通信経路の状態を考慮せずに決定され, 通信中の変更も行われない. 本稿ではオルタネイティブパスの状態を考慮し, アプリケーションの要求するスループットを満たした通信経路をプライマリパスとして選択する手法を提案する.

キーワード : SCTP, プライマリパス

A Study of Primary Path Handover in SCTP

HIROKI MITSUIE[†] XINZHUANG LIU[†]
YUE ZHAO[†] KEISUKE MORITA[†]

Abstract: In recent years, real time communication applications such as live streaming delivery is increasing. Those applications need communication quality high enough. However, since major transport layer protocols such as TCP and UDP are designed mainly for point-to-point communication, it is difficult for applications using such protocols to deal with the deterioration of communication quality unless the applications manage multiple connections by themselves. As a solution to this problem, SCTP provides multi-homing that enables communication using multiple IP addresses. It selects one path as primary path for data transfer and others as alternate paths for backup. However, in current SCTP, primary path is determined without considering the state of the communication path and no change is made during communication. This paper proposes a method that considers the state of alternate paths to select the primary path to give throughput requested by the application.

Keywords: SCTP, Primary Path

1. はじめに

近年, ストリーミング配信などの実時間性を伴う通信が増加している. このような通信を行うアプリケーションは一定以上の通信品質の確保を必要としている. また, ストリーミング配信のようなアプリケーションでは UDP のようなトランスポート層プロトコルを用いることでデータ転送量を固定化することで一定品質のスループットを得ようとする手法も存在するが, 輻輳制御を行わないために他の通信との公平性などを考慮することができない. また, TCP は競合するフローの数などによってスループットに大きな影響が発生する. そして, TCP や UDP のようなトランスポート層プロトコルは 1 対 1 の通信を前提として設計されているため, アプリケーションによる複数の接続の管理を行わない限り, ネットワークの状態の悪化や切断に対応することは難しい.

これらの問題に対応するには, 端末が複数の IP アドレスを用いた通信を可能にするマルチホーム機能を持つトラン

スポート層プロトコルを利用する必要がある. SCTP^[1] (Stream Control Transmission Protocol)は, マルチホーム機能を持つトランスポート層プロトコルであり, TCP のように信頼性の高い順序通りのデータ転送と輻輳制御を提供することが可能である. SCTP では通信をアソシエーションという単位で管理している. アソシエーションに含まれる 2 つ以上の IP アドレスを利用することで複数の通信経路を確保し, マルチホーム機能を実現している. マルチホーム環境で SCTP を利用する場合に, アソシエーション内で管理している IP アドレスの中からデータの転送に使用するプライマリパスと呼ばれる通信経路を選出し, それ以外の通信経路をオルタネイティブパスと呼ばれるバックアップ用の通信経路とする. SCTP におけるプライマリパスは, 明示的に指定しない限り, SCTP を実装している OS によって通信経路の状態やインタフェースを考慮せずに決定される. パケットロス率や RTT などが悪い状態の通信経路を使用することはスループットに大きな影響を与えるため, 複

[†] 広島工業大学
Hiroshima Institute of Technology

数の通信経路を有効に活用できない可能性がある。現在、SCTP においてオルタネティブパスを用いたデータ転送が可能なのはパケット再送とフェイルオーバーと呼ばれる一時的な通信経路の切り替えが行われる場合のみである。この問題を解決するためには、フェイルオーバー以外に、オルタネティブパスを使用する機能が必要である。本稿では、データ転送を行う通信経路が通信経路の状態を考慮せずに決定されることを防ぐために、プライマリパスの切替（ハンドオーバー）を行うことを提案する。提案手法ではハンドオーバーのための指標として用いる `cwnd` や `RTT` などの値を、輻輳制御を行った際にオルタネティブパスに対してデータ転送を行うことで計測する。その計測結果から計算されたスループットを基にプライマリパスを決定する。

以下、第2章にSCTPの概要と現在のオルタネティブパスの利用方法から考えられる問題点をまとめる。第3章では、第2章で挙げられた問題点を解決するための提案を行う。そして、第4章に提案手法利用した実験結果と考察をまとめ、第5章でまとめと今後の課題について述べる。

2. SCTP の概要

SCTP は TCP や UDP などの既存のトランスポート層プロトコルの持つ特徴に新たな機能を加えたトランスポート層プロトコルである。追加され機能で最も注目すべき点は複数の IP アドレスを扱うことができるマルチホーム機能である。SCTP は1つの通信をアソシエーションという単位で扱う。このアソシエーションを確立するための 4-way handshake によって通信に使用可能な IP アドレスのリストを交換することによってマルチホーム機能を実現している。

2.1 通信障害の検知

SCTP では、各通信経路が利用可能かどうかを判断するために HEARTBEAT パケットとそれに対する応答である HEARTBEAT-ACK パケットによって確認する。この HEARTBEAT パケットは、通信経路の到達性に問題が存在する可能性がある場合に送信される。SCTP がこの問題が存在する可能性があるかと判断するのは、`HB.interval` (idle 状態の通信経路に対して HEARTBEAT を周期的に送信する間隔) が経過した場合である。また、SCTP では通信経路の到達性が確認されている状態を ACTIVE、確認されていない状態を INACTIVE と呼ぶ。INACTIVE な状態の通信経路に対しても `HB.interval` ごとに HEARTBEAT パケットを送信し、到達性が確認できた場合は Error Counter の値を 0 にリセットする。

2.2 フェイルオーバー

現在、SCTP でオルタネティブパスが使用されるのは再送タイムアウト時のパケット再送とフェイルオーバーと呼ばれる一時的な通信経路の切り替えのみである。フェイルオーバーが起こると、データ転送を行う通信経路がオルタネティブパスに変更される。フェイルオーバーが行われる条

件となっているのは、P.M.R と呼ばれる Error Counter の閾値のみである。P.M.R の値は、5 が推奨されている。Error Counter は再送タイムアウトが発生した場合のみインクリメントされる。1回でも連続して再送タイムアウトが起こらなかった場合は、Error Counter は 0 にリセットされる。再送タイムアウトが連続で発生するごとに RTO の値は 2 倍にバックオフされる。推奨されている MIN RTO (RTO の最小値) の値は 1 秒、MAX RTO (RTO の最大値) の値は 60 秒である。そのため、推奨値を用いるとフェイルオーバーを行うためにかかる時間は 63 秒となる。この値ではアプリケーションやユーザからの要求に応えることは難しい。また、P.M.R の値を低く設定することでフェイルオーバーを早いタイミングで行うことができるが、すべての通信経路で Error Counter の値が P.M.R を超えるとすべての通信経路が到達不可能な状態にあるとみなされアソシエーション自体が終了してしまう可能性がある。このことから、プライマリパスの状態がオルタネティブパスと比較して悪い場合でも、フェイルオーバーを行うことができずに、プライマリパスを使用し続けてしまう。

またフェイルオーバーではプライマリパスのハンドオーバーが行われないために、周期的に Idle 状態（一定期間通信を行っていない状態）の通信経路に対して送信される HEARTBEAT パケットによって到達性が確認された場合にプライマリパスでの通信に復帰してしまうという問題も存在する。フェイルオーバーの流れを以下の図 1 にまとめる。

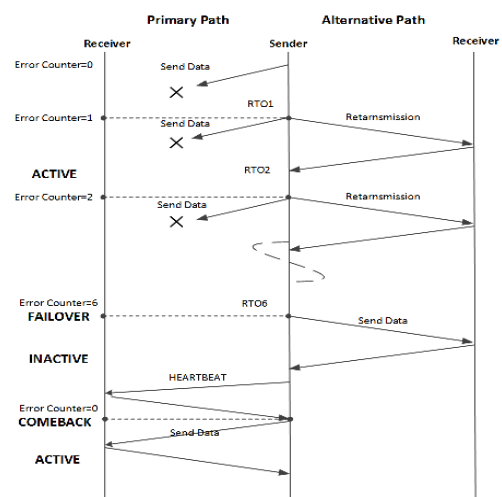


図 1 フェイルオーバーの流れ

2.3 SCTP における輻輳制御

SCTP におけるパケットの再送と輻輳制御は、基本的には TCP と同様であるが、マルチホーム環境で通信を行う場合に、異なる部分が存在する。SCTP では、再送タイムアウトした場合は、オルタネティブパスの 1 つを用いた再送を行う。再送タイムアウトの RTO の計算などは通信経路ごとに行われるが、計算方法は TCP と同様である。SCTP

ではデータの確認応答に SACK を採用している。SACK により、データ送信側は、パケットロスが起こったパケットを特定できる。また、輻輳制御を行うために管理されている $cwnd$ (送信ウィンドウサイズ)、 $ssthresh$ (スロースタート閾値) などの値は通信経路ごとに管理されるが、 $rwnd$ (受信ウィンドウサイズ) はアソシエーション全体に対して 1 つだけ管理する。

3. 提案手法

本稿での提案手法では、プライマリパスの切替によってアプリケーションが要求するスループットを満たす通信経路をプライマリパスとして選択することとオルタネティブパスのスループットがプライマリパスのスループットを上回っている場合にプライマリパスの切替を行うことを目標としている。

通常の SCTP が、オルタネティブパスを使用した通信を行うのは再送タイムアウト時のパケット再送とフェイルオーバーを行った場合のみである。そのためオルタネティブパスを利用できるのは、プライマリパスで継続的に高い頻度でパケットロスが発生する場合のみである。つまり、プライマリパスでデータ転送をほとんど行うことができない状態になるまでオルタネティブパスを使用することができない。提案手法では、再送タイムアウトによるスロースタートを除いた輻輳制御によって $cwnd$ の値が下げられた時点でプライマリパスにハンドオーバーをするかどうかのチェックポイントを設けることで、この問題の解決を行う。なお、フェイルオーバーのような再送タイムアウト時の処理は既存のまま維持する。

3.1 オルタネティブパスへのデータ転送

SCTP における輻輳制御は、基本的には TCP の New Reno 方式と同様のため、重複した SACK によってパケットロスが検知された場合に $cwnd$ を $1/2$ の値に設定し、高速再送と高速回復を行う。提案手法では、高速再送が終了した直後にオルタネティブパスを使用して、 $1/2$ になる前の $cwnd$ 分のデータ転送を行う。この時に送信されるデータはダミーのパケットを使用しない。そのため各パケットには TSN 番号が割り当てられており、SACK がどのパケットに対するものなのかを確認できるため RTT を求めることができる。この通信で得られたスループットを基に通信経路の状態を評価する。また、このオルタネティブパスへのデータ転送のことをチェックポイントと呼ぶ。スループットを得るために用いる計算式は以下の式(1)である。T はスループットを示す。

$$T = cwnd * (1000 / SRTT) * 8 \text{ [bps]} \quad (1)$$

プライマリパスのスループット計算時の $cwnd$ の値は $1/2$ になる前の値を用いる。

3.2 通信経路の評価指標

チェックポイントで行ったオルタネティブパスに対する通信から、得られる情報はパケットロスの有無と RTT である。 $cwnd$ はパケットロスが検出されると $1/2$ の値になるため、利用可能な帯域幅とパケットロスの有無を確認するための指標の 1 つとして用いることができる。RTT は通信経路の往復遅延時間を示している。また、RTT は輻輳によってキューイング遅延が増加するため、輻輳の度合いをスループットに反映させることができる。また、RTT は変化が激しいため、評価に用いるために平滑化を行う必要がある。RTT は平滑化を行うための式は以下の式(2)である。

$$SRTT = \alpha SRTT + (1 - \alpha) RTT \quad (2)$$

3.3 チェックポイントの設定条件

提案手法のチェックポイントの設定を常に行うとハンドオーバーを行うことができない通信経路に対してもデータ転送を何度も行ってしまうことになり、オルタネティブパスの他の通信に悪影響を与えてしまうことにつながるため、チェックポイントを設定することに対して条件を設定する必要がある。今回は、以下の 2 つの閾値をチェックポイントの設定条件に用いた。

- (1) 最後に計測されたオルタネティブパスのスループット
- (2) アプリケーションが要求するスループット

(1)の閾値をプライマリパスのスループットが下回った場合にチェックポイントの設定を行う。(1)の閾値は、チェックポイントの設定を頻繁に行いすぎること防ぐだけでなく、プライマリパスのスループットに大きな低下がない限りハンドオーバーによるプライマリパスの振動を抑える役割を持つ。

(2)の閾値を(1)の閾値が下回った場合に一定時間後にチェックポイントの設定を行う。(2)の条件でチェックポイントの設定を行った場合は、ハンドオーバーを行ったかどうかに関わらず一定時間経過後に再びチェックポイントを設定する。これは、アプリケーションの要求するスループットを得るためだけでなく、(1)の閾値が低過ぎる値に設定されてしまった場合にチェックポイントを設定できないためにオルタネティブパスの状態を確認できなくなることを防ぐ役割も持つ。

また、通信開始時のスロースタートの直後はスループットが安定しないため、スロースタート終了後から 10 秒間はチェックポイントの設定は行わない。そのため、最初のチェックポイントの設定は通信開始から約 10 秒後に行う。提案手法の流れについて図 2 にまとめる。

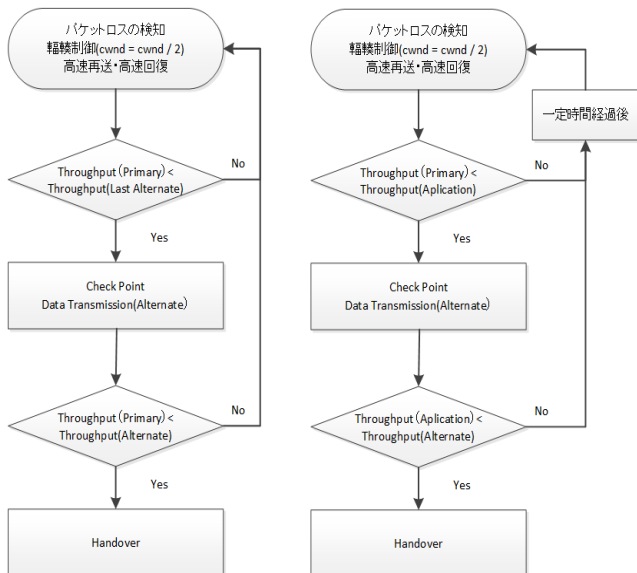


図 2 提案手法の流れ

3.4 関連研究

関連研究^{[2][3]}には、SCTP の HEARTBEAT 機能を使用したものがある。この類似研究での手法は、提案手法とは異なり、VoIP などの CBR(固定ビットレート)で行われる実時間通信での通信を対象とし、Unorderd mode (データの順序を保証しないモード)でシミュレーションしたもので輻輳制御やパケット再送は行われない。CBR はパケットロスが起きた場合でも、一定間隔でデータ通信を続ける通信方法である。HEARTBEAT 機能は、オルタネティブパスの通信経路が利用可能かどうかを確認するために、HB.interval ごとに HEARTBEAT パケットと HEARTBEAT-ACK パケットの送受信を行う機能のことである。デフォルトでは 30 秒である HB.interval を 1 秒に設定し、HEARTBEAT パケットをペアで送信することによって、RTT とボトルネックリンクの計測を行っている。VoIP などの実時間通信はアプリケーションごとに要求帯域や障害復旧時間が決まっているので、これらを考慮して一定間隔ごとにハンドオーバーを行うかどうかの評価を行っている。

4. 実験による評価

提案手法によってボトルネックリンクの性能が高い通信経路がプライマリパスに選択されることを確認するために実験を行う。アプリケーションが要求するスループットは 2Mbps に設定する。提案手法の SCTP を FreeBSD 10.0 に実装し、図 3 のトポロジを用いて実験を行う。伝搬遅延と帯域幅、キューサイズの制御は Dummynet を用いて行う。Dummynet は、遅延やパケットロス率、帯域幅などを制御できるツールである。今回の実験ではキューサイズを 50Kbyte に設定した。

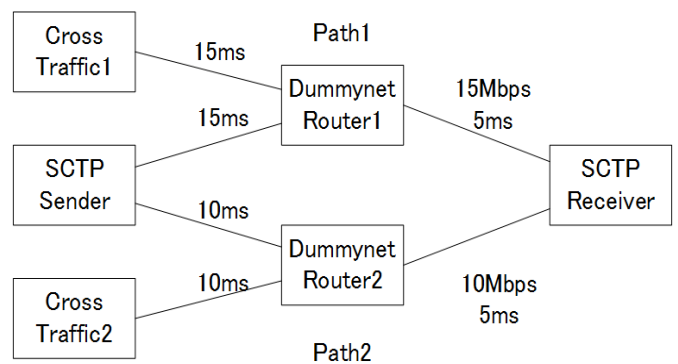


図 3 トポロジ

4.1 実験 1 ボトルネックリンクの安定した通信

実験 1 では輻輳が発生していないオルタネティブパスへのハンドオーバーを正常に行えているかどうかを明らかにする。図 3 のトポロジに対して表 1 のタイムテーブルに沿って Cross Traffic1 と Cross Traffic2 から Sctp Receiver に対してデータ転送を行う。また、通常の SCTP と提案手法の差を明らかにするために、各スループットの比較を行う。計測時間は 300 秒とする。なお、この実験で最初に選択されるプライマリパスは Path1 とする。図 4 に通常の SCTP での通信での通信の実験結果をまとめる。また、図 5 に提案手法のスループットを通信経路ごとにまとめる。そして、表 2 に既存手法と提案手法のスループットの比較をまとめる。

表 1 タイムテーブル (実験 1)

Time[s]	Path1	Path2
0~100	CBR 7Mbps	CBR 4Mbps
100~200	CBR 11Mbps	CBR 4Mbps
200~300	CBR 11Mbps	CBR 8Mbps

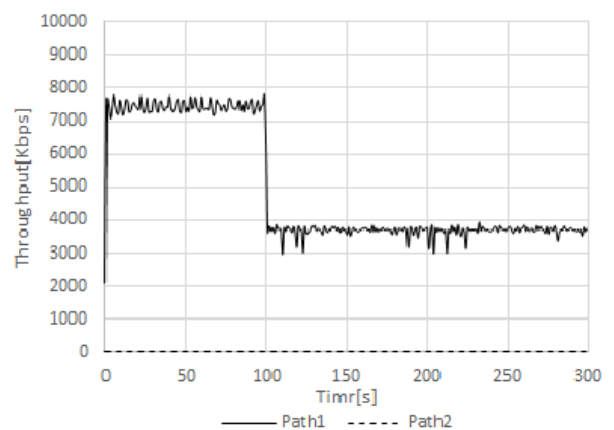


図 4 既存手法 (実験 1)

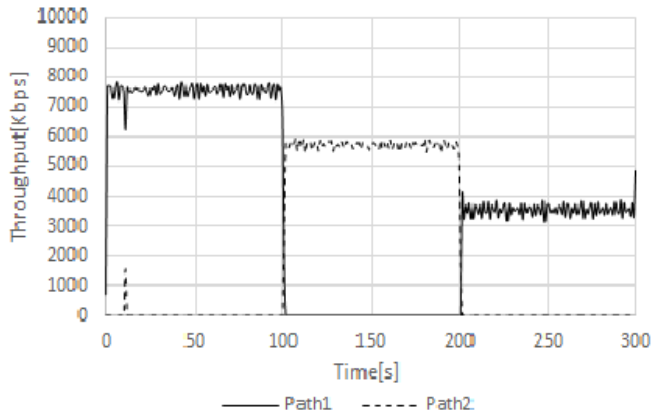


図 5 提案手法 (実験 1)

表 2 スループットの比較 (実験 1)

Time[s]	既存手法	提案手法
0~100	7.4Mbps	7.5Mbps
101~200	3.7Mbps	5.7Mbps
201~300	3.7Mbps	3.7Mbps

以下に実験 1 の提案手法の実験結果に対する考察を述べる。通常の SCTP では再送タイムアウトが連続して発生しなかったためフェイルオーバーを行うことができない。常に Path1 を使用した通信を行う。それに対して、提案手法を実装した SCTP では、プライマリパスの帯域幅が悪くなったタイミングでハンドオーバーを行うことで 100 秒から 200 秒の間に帯域幅が広い Path2 を使用した通信を行うことでスループットが向上している。最初のチェックポイントではオルタネイティブパスでパケットロスが発生し、ハンドオーバーは行われなかった。100 秒時点のチェックポイントで計測された Path1 のスループットは、約 3.6Mbps で Path2 のスループットは約 6Mbps であったためハンドオーバーを行った。200 秒時点のチェックポイントで計測された Path1 のスループットは約 2.6Mbps で Path2 のスループットは約 3Mbps であったためハンドオーバーを行った。これらの実験結果から、ボトルネックリンクが安定している輻輳が発生していないオルタネイティブパスへのハンドオーバーを正常に行えていることが明らかになった。

4.2 実験 2 ボトルネックリンクが不安定な通信

実験 2 では輻輳が発生しているオルタネイティブパスへ利用可能な帯域がプライマリパスより優れている場合に、ハンドオーバーを正常に行えているかどうかを明らかにする。実験 1 と同じく図 3 のトポロジに対して表 3 のタイムテーブルに沿って Cross Traffic1 と Cross Traffic2 から SCTP Receiver に対してデータ転送を行う。Cross Traffic の TCP の最大スループットは約 8Mbps になるように設定してい

る。また、通常の SCTP と提案手法の差を明らかにするために、各スループットの比較を行う。計測時間は 200 秒とする。なお、この実験で最初に選択されるプライマリパスは Path1 とする。図 6 に提案手法のスループットを通信経路ごとにまとめる。また、表 4 に既存手法と提案手法のスループットの比較をまとめる。表 4 の既存手法でのスループットに関しては、実機での実験を行ったためパケットロスが発生するタイミングなどを再現することができないので TCP を除いた利用可能帯域をフロー数で割った値を記述している。

表 3.タイムテーブル (実験 2)

Time[s]	Path1	Path2
0~50	TCP*1	TCP*1
50~100	TCP*1 + CBR 7Mbps	TCP*1
100~150	TCP*1 + CBR 7Mbps	TCP*2
150~200	TCP*2 + CBR 7Mbps	TCP*2

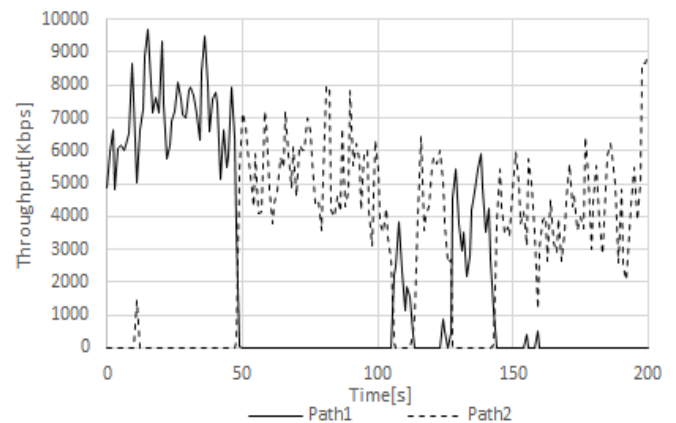


図 6 提案手法 (実験 2)

表 4 スループットの比較 (実験 2)

Time[s]	既存手法	提案手法
0~50	7.5Mbps	7Mbps
51~100	4Mbps	5.5Mbps
101~150	4Mbps	3.7Mbps
151~200	2.7Mbps	4.3Mbps

以下に実験 2 の提案手法の実験結果に対する考察を述べる。50 秒と 106 秒の時点ではそれぞれ Cross Traffic によるプライマリパスの利用可能帯域の低下を検知してオルタネイティブパスへのハンドオーバーが行われている。しかし、106 から 113 秒の間にプライマリパス(Path1)で再送タイムアウトが発生し、スロースタートを行ったために大幅にスループットが低下してアプリケーションの要求しているスループット(2Mbps)を下回ったためにハンドオーバーを行っ

ている。その後、一定時間経過後(10秒後)に設定されるチェックポイントでハンドオーバーが発生し、プライマリパスが Path1 に復帰している。143秒時点では、Cross Traffic のフロー数は増加していないが、プライマリパスの `cwnd` の値が低くなった場合に、オルタネイティブパスへのデータ転送量が少なくなり輻輳によるパケットロスの検出が難しくなったことと、SRTT の正確性が低下したためだと考えられる。159秒の時点でも、再送タイムアウトによる大幅なスループットの低下でチェックポイントが設定されている。これらの実験結果から、ボトルネックリンクが不安定で継続的に輻輳が発生しているオルタネイティブパスへのハンドオーバーは、再送タイムアウトなどによって異常値が検出された場合に正常に機能しない場合があるので、この問題に対する対策が必要である。

5. おわりに

本稿ではネットワークの状態を考慮せずにプライマリパスを決定することで、適切な通信経路をプライマリパスとして選択できれば、得ることができるはずのスループットを損なってしまうという問題を明らかにした。そして、この問題を解決するための1つの手法を提案した。この提案手法では、輻輳制御時にオルタネイティブパスへのデータ転送を行うことで計測された `cwnd` と RTT を基にスループットを計算し、プライマリパスのハンドオーバーを行う。そして、提案手法を実装し実験を行った。

今後の課題として、提案手法のオルタネイティブパスのスループットの計測方法ではプライマリパスの `cwnd` の値が低くなった場合に、オルタネイティブパスへのデータ転送量が少なくなり輻輳によるパケットロスの検出が難しくなる場合がある。また、今回の実験では SRTT や `cwnd` の値に対して重みをつけていないため、これらの値に係数を設定して許容範囲を持たせること、単位時間内に一定回数ハンドオーバーの条件を満たした場合のみハンドオーバーを行うことなどを検討している。そして、実験2のように再送タイムアウトの直後に異常値を計測してしまいハンドオーバーを行ってしまう可能性があるためハンドオーバーの条件の整備をさらに進める必要がある。

参考文献

- [1] R. Stewart, "Request for Comments 4960 : Stream Control Transmission Protocol", 2007
- [2] 西山尚志, 榎原茂, 飯田勝吉, 山口英, "SCTP パス切り替えに関する研究 : プライマリパスの変更", 電子情報通信学会技術研究報告. IN, 情報ネットワーク 102(693), pp71-76, 2003
- [3] 十鳥幸祥, 舟阪淳, 石田賢治, 小畑博靖甘, "SCTP におけるパス切替方式に関する考察", 電子情報通信学会技術研究報告. IN, 情報ネットワーク 103(493), pp19-24, 2003-12-05
太田政宏, 西山尚志, 榎原茂, 飯田勝吉, 山口英,
- [4] SCTP のパス切り替えに関する研究 パス切り替えの振動の調査, 電子情報通信学会技術研究報告. NS, ネットワークシステム 102(691), pp65-69, 2003-02-27

- [5] Manoja Dahal, Dilip Kr. Saikia, RTT Based Congestion Control and Path Switching Scheme for SCTP, Conference: Communication Technology, 2006. ICCT '06. International Conference
- [6] Fang-Yie Leu, Fenq-Lin Jenq, Fuu-Cheng Jiang, A path switching scheme for SCTP based on round trip delays, Computers & Mathematics with Applications (Impact Factor: 1.7). 11/2011; 62(9):pp3504-3523
- [7] Dong Phil Kim, Dong Hwa Lee, Seok Joo Koh, Yong Jin Kim, Adaptive Primary Path Switching for SCTP Handover, Advanced Communication Technology, 2008. ICACT 2008. 10th International Conference
- [8] Andrew Kelly, Philip Perry, John Murphy, A Modified SCTP Handover Scheme for Real Time Traffic, In HETNETs Working Conference, Ilkley, England