

インターネットにおける複数経路データ転送手法*

2F-6

渡部 直樹 グンティダー・ロットウィブンチャイ 相田 仁†
東京大学大学院新領域創成科学研究科‡

1. はじめに

2つの装置間でデータの転送を行う場合、経路を複数設定することにより通信性能が向上できる。これは1つの経路に障害が発生した場合に正常な別の経路を用いることで信頼性の高めることができるのはもちろんであるが、複数の経路を同時に利用して通信することによる信頼性の向上と通信時間の短縮が可能であることを意味する。この複数経路データ転送をインターネットに適用することで、今後増加していきであろう電子商取引、電子決済、行政上の各種証明書・申請手続など高い信頼性を必要とする作業を円滑に進めていくことが可能である。そこで本稿ではインターネットで複数経路データ転送を可能にする通信プロトコルの提案を行う。

2 複数経路データ転送方式

複数経路データ転送方式とは、データを送信する際に送信元ノードから受信先ノードへの転送ルートをもつては複数設定し、それぞれのルートにデータを転送する方式である。複数経路でデータを転送する場合、信頼性の向上を目指すかスループットの向上を目指すかで適用手法が異なってくる。以下にその手法を説明する。

2.1 信頼性の向上を目的とする場合

信頼性を重視する場合は、経路の障害や輻輳時でも通信が途切れることなく正常に行える必要がある。したがって図1に示すように全経路に同時にパケットを送信する方法が考えられる。

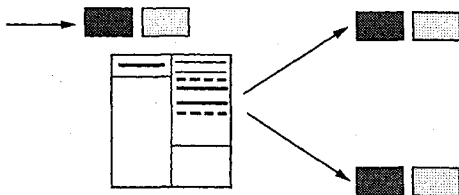


図 1: copy to each path

この方法を用いることにより、複数経路のうちある1つの経路で障害や輻輳が発生し、データの損失や大幅な遅延などが生じた場合でも他の経路から送信されたデータによって大きな遅延なくそのデータを補うことができ、送受信ノード間で見た場合に通信を中断することなく続けることができる。

また、障害・輻輳が生じていない通常の通信状態においても、受信ノード側において到着するデータのうち、同じもので最も早く到着したデータを使うことで、全経路中で最も効率良く経路を選択でき、単独の経路で通信を行うよりもスループットの向上が測れる。

2.2 スループットの向上を目的とする場合

通信時間を短縮してスループットの向上を考える場合、各経路を並列に利用してデータを転送する方法が考えられる。データの転送方法として図2に示すようにパケットごとに異なる経路へ分配する方法が考えられる。

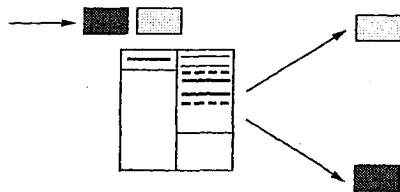


図 2: distribute to each path

各経路に障害が無い場合は経路数倍のスループットが得られることになる。またどこかの経路に障害が発生した場合は、その経路を使用せず他の経路を利用するという動的な経路の切り替えも可能である。

複数経路データ転送を IP 層で行うことも研究されているが [1]、TCP 層で実行する場合には、端末側で複製方式と分配方式を選択的に利用することが可能であり、コネクションの確立の際には複製方式を使用し、安定した後に分配方式に切り替えてスループットの向上を図ることもできる。したがってさらなる通信性能の向上が期待できる。

そこで TCP 層で複数経路データ転送を行うことを考え、実際にインターネットで広く使用されている TCP を拡張して新しいコネクション指向のプロトコル M/TCP[2] を提案する。

*Multipath data transmission on the Internet

†Naoki Watanabe, Rojviboonchai Kultida, Hitoshi Aida

‡Graduate School of Frontier Sciences, The University of Tokyo

3 提案する TCP オプション

TCP との互換性を保持するため、提案するヘッダは TCP ヘッダにオプションの形で付加することを考える。

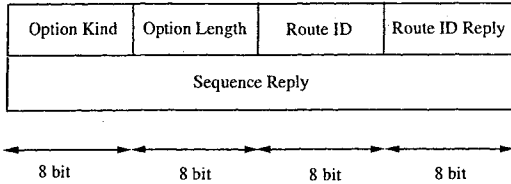


図 3: 提案する TCP オプション

提案方式ではオプションフィールドに Route ID と Route ID Reply, Sequence Reply を付加する。

Route ID はデータがどの経路で送信されているかを表す。一方で Route ID Reply はどの経路から受信したデータに対して ACK を返しているかを表す。これは M/TCP では送信データに対する ACK がデータを送った経路と別の経路で返ってくることもあるため、どの経路からのデータに対する ACK かが分からなければ RTT の計算や輻輳ウィンドウの更新ができなくなるからである。

Sequence Reply は ACK を返す際に、送られてきたデータの先頭シーケンス番号を知らせるのに利用する。これはデータを受信してその ACK を返す場合は、ACK フィールドに次に受信を期待するセグメントの番号を入れて返すので、送信側は相手に届いたセグメントの終わりの番号しか知ることができない。分配方式の場合では、相手に届いたセグメントの開始番号も送信側で知る必要があるため、このフィールドを設けて送信側はどこからどこまで相手に届いたかを通知する。

これらのオプションがどのように使われるかを、Route 1 から来たデータを Route 2 へ返す場合を例にとって図 4 に示す。

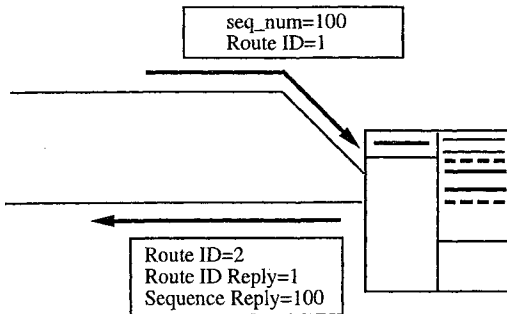


図 4: データ受信時の処理

まず送信側では Route 1 の経路にデータを送り出すので、Route ID を 1 にセットする。

受信側でデータを受け取ると、それに対する ACK を返すことになるのだが、Route 1 からのデータを受け取ったので Route ID Reply を 1 にする。また受け取ったデータのシーケンス番号が 100 であったので、Sequence Reply を 100 に設定する。最後に ACK を Route 2 へ返す場合は Route ID を 2 に設定して ACK を送り出すことになる。

4 コネクションの確立と解放

TCP ではスリーウェイハンドシェイクを用いてコネクションの確立を行うが、M/TCP でも同様にスリーウェイハンドシェイクを用いる。M/TCP では受信側で全経路から SYN を受け取ったところで ACK を全経路に返すことにし、送信側でも ACK を全経路から受け取って初めて次の状態へ移行するものとする。何度か再送を行っても全経路から SYN が到着しない場合は RST フラグをセットしたセグメントを全経路に送信し、初期状態に戻す。

一方、コネクションの解放の場合であるが、M/TCP では FIN や ACK がセットされたセグメントが全経路に送られるが、そのうち一つが届いた時点ですぐ、それに対する ACK を返して次の状態へ移行して構わない。ただし全てのセグメントがネットワーク上からなくなることを保証するため、最後の 2MSL(Maximum segment life) タイムアウトは一番遅い経路にあわせて設定する必要がある。

5 まとめ

本稿ではまず複数経路データ転送方式について複製方式と分配方式を説明し、この手法をインターネットに適用するため、TCP 層で複数経路データ転送を実現することを考えてプロトコルの提案を行い、その動作について説明した。

今後は提案プロトコルの検討を進め、シミュレーションを行って、複製方式についてはパケットロスの改善による信頼性の向上、分配方式についてはスループットの向上について評価を行う予定である。

参考文献

- [1] 林孝典、山崎真一郎、森田直人、相田仁、武市正人、土居範久：“インターネットを用いた複数経路データ伝送方式に関する一検討”，電子情報通信学会総合大会,B-7-78,1999.
- [2] 中村成貴：“複数経路コネクション指向プロトコル M/TCP の提案”，東京大学大学院工学系研究科情報工学専攻修士論文,2001