

5 V - 3

選択的応答情報によるトランスポートプロトコルの帯域制御方法の検討

渥美幸雄 近藤栄一

(株) 超高速ネットワーク・コンピュータ技術研究所

1. はじめに

TCPを代表とするトランスポートプロトコルの帯域制御（輻輳制御）では、ネットワークの負荷状況に応じて適切に送信量を調整して、安定な高スループットを実現することが必要である。一方、誤り回復の迅速化を狙いとした選択的応答（SACK）の機能が昨年RFCとして規定された[1]。SACKは基本的には誤り制御の手段であるが、パケット紛失状況をより正確に把握できるため、本情報を利用したより有効な帯域制御が可能となる。

本稿では具体的なトランスポートプロトコルとしてTCPを対象として、従来方式での性能上の問題を示した後に、SACKに基づいてネットワーク負荷状況をより適切に反映する帯域制御方法を提案する。

2. 問題点

2.1 TCPの帯域制御

BSD-LiteのTCP[2]の帯域制御の概要を以下に示す。

(1) パケット紛失事象検出時のssthresh/cwnd削減

パケット紛失事象を検出すると送信量を削減するが、紛失状況（1パケット紛失なのか、複数個の紛失なのか）に依らず、帯域制御に関する変数であるssthreshとcwndを以下のように一律に削減する。

$$\text{ssthresh} = \max(\min(cwnd/2, adwind), 2 * MSS)$$

$$cwnd = ssthresh$$

(2) ACK受信時のcwnd増加

- スロースタートフェーズ*

$$cwnd = cwnd + MSS$$

- 輻輳回避フェーズ*

$$cwnd = cwnd + MSS * MSS / cwnd + MSS / 8$$

即ち、輻輳回避フェーズでは、cwnd値に相当する個数分毎に1パケット増であり、RTT(ラウンドトリップタイム)が大きい長距離回線では収束に時間を要する。

2.2 輻輳回避フェーズでの性能問題

(1) 問題概要

cwndの増加に関して、スロースタートフェーズではACK受信毎にMSS(最大セグメントサイズ)分づつ増加するのに対して、輻輳回避フェーズでは少しずつ徐々に行われる所以、パケット紛失事象検出後のssthreshとcwndの設定が不適切だとスループット性能への影響が大きい。即ち、パケット紛失検出時にssthreshとcwndは一律に削減のため、ネットワークの負荷状況を反映した送信量の設定とならない恐れがある。ここでは輻輳回避フェーズでの具体的な性能上の問題をx-Simシミュレータを使用して定量的に示す。

(2) x-Simの概要[3]

x-Simはアリゾナ大で開発されたプロトコル評価用のシミュレータであり、UNIX上のアプリケーションとして動作する。シミュレーション用のプロトコルコードは、実装コードに非常に近いものであり、シミュレータで試験確認したコードの実機への実装が非常に容易なことを特徴としている。

(3) 評価モデル

パケット紛失事象を検出後の輻輳回避フェーズでの平均スループットを比較する。従来方式(cwndを1/2削減)で、cwndが一定値Pになるまで(tR時間)の平均スループットThOと、cwndを適切値Sに設定時のtR時点までの平均スループットThNとを比較する(図1)。

評価ネットワーク構成と条件を図2に示す。平均スループットの情報収集を容易に行うため、コネクション設定直後から輻輳回避フェーズとなるように、ssthreshとcwndの初期値を設定した。

(4) 結果

評価例を表1に示す。本例ではパケット紛失事象検出時のcwnd削減を、cwnd=1/2ではなく3/4とすることにより約2割の改善が可能となる。従って、帯域制御のパラメータは、ネットワーク負荷状況に応じて適切に削減することが重要である。

3 SACK概要

SACK機能はTCPオプションとしてRFC2018で規定されている。輻輳制御との関係は、本RFCでは既存機能が保存されるべきとだけあり、SACKを

利用する方法については触れていない。文献[4]ではSACKの各種の可能性を指摘している。

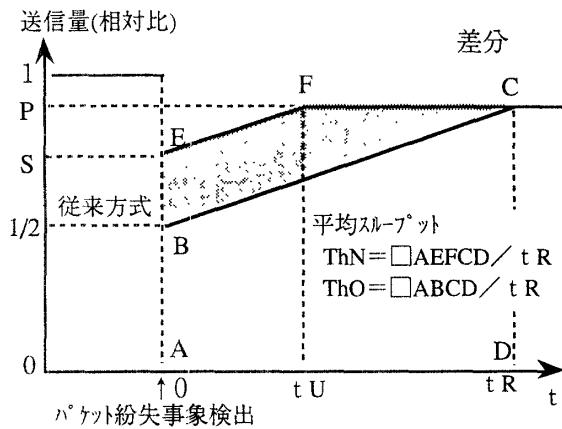


図1 平均スループット比較のモデル



図2 評価ネットワーク構成と条件

表1 平均スループット改善比 (ThN/ThO)

ハ'ケット紛失事象 検出時のcwnd	RTT=100msec	RTT=10msec
64KB	1.23	1.24
128KB	1.24	1.25

[評価条件] P = 7/8, S = 3/4,

回線速度 = 155Mbps、パケット長 = 4KB

4 提案方式

SACK情報は、受信側で非連続受信（即ち、パケット紛失により）となったパケット全てについて、受信できた区間を具体的に通知するものである。送信側では本情報から逆に、受信できなかった区間の紛失パケットのデータ量を算出できる。

提案する帯域制御の方法は、この紛失データ量の情報(LostDT)に基づいて、ネットワークの負荷状況を適切に反映した送信量の削減を行うものである。即ち、紛失データ量はネットワークの負荷状況を示していると考えられるためである。

基本的な新送信量の算出法の考え方、

$$\text{新送信量} = \text{旧送信量} \times d$$

d : 削減係数 (LostDTに依存)

であり、紛失パケット数に基づいて削減する(表2)。

例えば、紛失数が1個の場合はd1、2個以上の場合はd2で削減し、d1=3/4、d2=1/2を考え

られるが、適切な値は今後の評価による。なお、不要な過削減を避けるため、最初のSACK受信時点での最大未確認シーケンス番号を記憶(chseq)しておき、以降に受信するACKのシーケンス番号と比較して、1 RTT経過の判断に使用する。

表2 ACK受信時の処理概要

	通常状態	SACK状態
SACKなし	・従来処理	・従来処理 ・通常状態へ
SACKあり	・(処理1) ・SACK状態へ	・(処理2)

ackseq : ACKの受信シーケンス番号

(処理1)

chseq ← 最大未確認シーケンス番号

lostDTb ← SACK情報からの紛失パケット数

lostDTb = 1 の場合

ssthresh × d1, cwnd × d1

lostDTb > 1 の場合

ssthresh × d2, cwnd × d2

(処理2)

chseq ≥ ackseq の場合

lostDTb = 1 の場合

lostDT ← SACK情報からの紛失パケット数

lostDT = 1 の場合

(削減なし)

lostDT > 1 の場合

ssthresh × d1, cwnd × d1

lostDTb ← lostDT

lostDTb > 1 の場合

(削減なし)

chseq < ackseq の場合

処理1 と同じ

5 おわりに

SACK情報に基づいて適切な帯域制御（送信量の調整）を行う方法を述べた。今後、提案方式のミュレーション評価を進める予定である。

[参考文献]

[1] M.Mathis, et. al., "TCP Selective Acknowledgment Options", rfc2018, 1996.10

[2] Gary R.Wright, W.Richard Stevens,

"TCP/IP Illustrated, Volume2 : The Implementation "

[3] <http://www.cs.arizona.edu/xkernel/>

[4] K.Fall and S.Floyd, "Simulation-based Comparison of Tahoe, Reno, and SACK TCP", <ftp://ftp.ee.lbl.gov/papers/>