

2V-3 RTPを用いたリアルタイム映像通信のための 送信レート FastStart アルゴリズムの検討

松浦友彦[†] 佐藤宏明^{††} 二木一^{†††}

キヤノン（株） 情報技術研究所

1. はじめに

インターネットなどのヘテロで帯域保障のないネットワークを介した連続メディアのリアルタイム伝送に関する研究や実装が盛んに行われている。インターネットは接続形態が端末ごとに多様であり、ネットワークの状況も時間とともに変動するため、End-to-End の実効帯域幅を予め知ることは困難である。しかし、実効帯域幅を超えてデータが送信されると輻輳が生じ、伝送遅延やジッタの増大、パケットロス率の上昇などの障害が発生する原因となる。この問題を解決するために、End-to-End において計測可能なネットワークの状況に関する情報を利用して送信データ量の制御を行う研究がなされ、そこで用いるプロトコルとして RTP(Real-time Transport Protocol)が標準化されている。

我々も先に End-to-End の遅延から、通信路上の各ノードのバッファに滞留している総データ量（以下、ネットワークバッファ量）を推定し、それを一定に保つように送信レートを制御する CORBED[2] を提案した。CORBED はクライアントがモデムや ISDN で接続した時のような明確な狭帯域通信路が存在する環境で特にその有効性を示したが、安定性を重視した制御アルゴリズムであるため、RTP セッションの開始から高いリンク使用率に達するまでに多くの時間がかかるという問題があった。本報告では、この問題を解決するために追従性を重視した FastStart アルゴリズムの追加を提案する。

2. CORBED 概要

A Fast Start Algorithm for RTP-based Real-time Video Transmission

Tomohiko Matsuura

Canon Inc. Information Technology Laboratory

53, Imaikami-cho, Nakahara-ku, Kawasaki-shi, Kanagawa 211-8501, Japan

E-mail: matsuura@cme.canon.co.jp

[†] 現在、キヤノン（株）DR 事業推進センター

^{††} 現在、キヤノン（株）3D システム開発センター

CORBED はサーバ〜クライアント間のラウンドトリップ時間から推定したネットワークバッファ量を一定に保つように送信レートを制御するサーバ主導型の送信レート制御アルゴリズムである。

ネットワークバッファ量 Best の推定は、一定時間毎にクライアントから届けられる RTCP 受信者報告パケットから得られる現在のラウンドトリップ時間 RTT と受信レート Rr を用いて、

$$\text{Best} = Rr * (\text{RTT} - \text{RTTmin}) \quad (1)$$

のように推定される。ここで RTTmin はネットワークバッファ量が0の時のラウンドトリップ時間である。次に式(1)により推定されたネットワークバッファ量 Best を目標ネットワークバッファ量 Bdes に保つように送信レート Rs を決定する。ここでも RTCP 受信者報告パケットから入手可能な現在の受信レート Rr と RTCP 受信者報告パケットの到着間隔 RRint を用いることにより、

$$Rs = Rr + (Bdes - \text{Best}) / \text{RRint} \quad (2)$$

のように送信レートを決定することができる。

3. FastStart アルゴリズム

CORBED のように流量データの累積・積分的評価を行ってネットワークの状況を判断すれば安定した輻輳回避が可能であるが、大きく送信レートを増加させるには多数の RTCP 受信者報告パケットが必要である。ところが RTCP 受信者報告パケットの發送間隔は RTP 標準で最短5秒と定められているため、RTP セッションの開始後に CORBED を使用して高いリンク使用率に達するまでには多くの時間が必要であった。

そこで我々は、RTP セッション開始直後の低送信レート時に輻輳が発生する可能性が低いことと、通信路の上り下りのデータ量が非対称であるため伝送遅延の大ききの割に RTCP 受信者報告パケットは素早く届くことに着目し、受信レートのみを監視

することで比較的安全で追従性の良い FastStart アルゴリズムを構成した。本アルゴリズムによりクライアントの受信レートに対してサーバの送信レートを先行的に上昇させることができる。構成した FastStart アルゴリズムは以下の手順で動作する。

- ① ある初期送信レートで送信を開始する。
- ② 一定間隔で届く RTCP 受信者報告パケットからクライアントの受信レートを参照する。
- ③ 受信レートが送信レートに達している場合、送信レートを α 倍に高め、手順②に戻る。
- ④ ある時間内に受信レートが送信レートに達しなかった場合、本アルゴリズムを終了する。

4. 実験および考察

CORBED 及び FastStart を RTP 上で動作する映像通信アプリケーションに実装し、送信レートの挙動を観測した。サーバは研究室内の LAN に直接接続されており、クライアントは 28.8Kbps モデムまたは 64Kbps ISDN で上記 LAN にダイヤルアップ接続を行う。目標ネットワークバッファ量 Bdes は、リンク使用率を減少させないような下限とバッファあふれを生じさせないような上限の間で適切に設定する。ここでは[2]に習い 2000bytes と設定した。FastStart の送信レート乗数 α は、送信レートの上昇速度と安全性のトレードオフにより設定する。ここでは TCP の制御アルゴリズムを参考にし、2.0 と設定した。FastStart のタイムリミット(手順④)はネットワークの応答性に依存し、RTT, RTCP の発行間隔及びジッタなどを基に設定可能である。ここでは 5 RRint (= 約 25 秒) とした。

同じ条件による実験を数回繰り返して行い、送信レートが通信路の帯域幅の上限付近に達するまでの平均的な時間を比較した結果、28.8Kbps モデムの場合で約 50%, 64Kbps ISDN の場合で約 30% に短縮することができた。64Kbps ISDN の場合の一実験例を図 1 に示す。同図(a)は CORBED のみを、同図(b)は FastStart を併せて使用した場合の結果である。FastStart が約 60 秒間実行され、その後 CORBED に移行してネットワークバッファ量を 2000bytes に維持していることがわかる。また、同

図(c)は送信レートがオーバーシュートした場合の例であり、一時的にネットワークバッファ量が目標値を超えている。これは FastStart アルゴリズムによる過剰なデータ送信が原因であり多少のパケットロスが生じているが、CORBED に移行した後に最小限の障害で回復している。

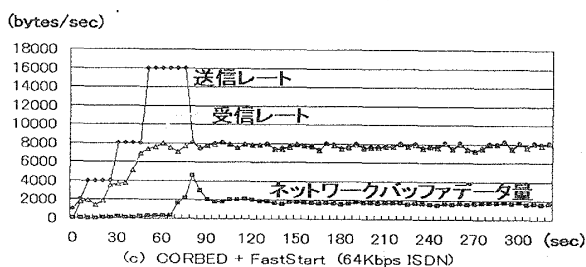
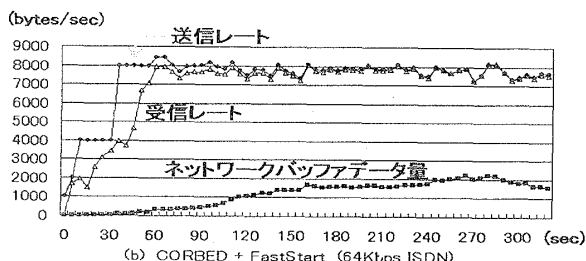
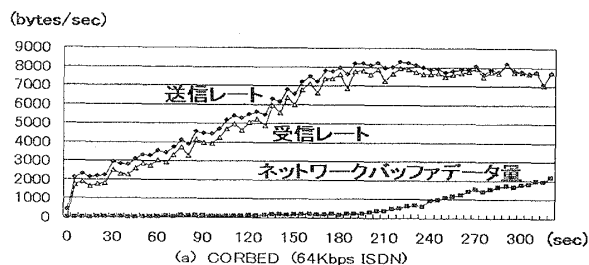


図 1: 送信実験例

5. まとめ

今回導入した FastStart アルゴリズムにより、RTP セッション開始直後に送信レートが通信路の帯域幅を充分に利用するレベルに到達するまでの時間を大幅に短縮することができた。

最近検討されている RTP の Long Fat Pipe 対応に伴い、RTCP 受信者報告パケットの發送間隔をより小さくしようとする動きがあるが、我々の提案する FastStart アルゴリズムは、RTCP の頻度が増加した場合にも有効で、より実用的な手法になると考えられる。

- [1] J-C. Bolot and T. Turletti: A rate control mechanism for packet video in the Internet, Proc. IEEE Infocom '94, pp.1216-1223, Tronto, Canada, June 1994.
- [2] 矢野晃一, 佐藤宏明, 瀬崎薫: 連続メディアのリアルタイム転送に適したラウンドトリップ時間に基づくレート制御法, 電子情報通信学会, 信学技法 IN98-23, pp.85-90, 1998