

## XTP プロトコルにおけるレート制御の実現方式に関する検討

5 V-4

三宅 優 石倉 雅巳 鈴木 健二

国際電信電話株式会社 研究所

### 1. はじめに

伝送路の高速化にともない、高速通信に対応するデータ通信プロトコルの開発が重要となってきた。高速ネットワークを使用する際の問題点の1つとして、受信処理能力を越える速度でデータが到着することが考えられる。このような場合、到着した全てのデータを処理できないためにおこるオーバーランにより、パケットロスが生じる。高速・高品質なネットワークでは、パケットロスなどによる誤りをできる限り抑え、誤り回復処理が起動されない状態で通信を行なうことが、高スループット化をはかる重要なポイントである。そのため、高速通信に対応したXTPプロトコル<sup>[1]</sup>では、送信側で受信側の処理能力に合わせてデータの転送速度を調整するレート制御方式が採用されている。本稿では、XTPプロトコルにおけるレート制御方式に着目し、実装の際に重要となるレート変更が行なわれるタイミングやレート値の決定法について検討したので報告する。

### 2. XTPプロトコルにおけるレート制御の方法

図1に、XTPプロトコルにおける、レート制御方式を示す。XTPでは、送信側でデータの転送レートを制御するために、以下の値および変数を使用する。

<i>rate</i>	1秒あたりの最大転送バイト数。
<i>burst</i>	複数パケットから構成される1バーストで送信可能な最大バイト数。
RTIMER	レート制御タイマ値で、インターバルタイマとして使用される。値は、 $burst/rate$ 。
<i>credit</i>	RTIMERがタイムアウトするまでに送ることが可能な残りバイト数を表す内部変数。

*rate* および *burst* の値は、受信側から送られてくるプロトコル制御用パケットのCNTL(Control)パケットに含まれている。送信側は、データを送信する毎に、*credit* の値から送信されたユーザデータのバイト数を減らしていく。*credit* がゼロまたは負になった場合には、データの転送を中止する。*credit* の値は、RTIMER 値の間隔で、以下の手続きにしたがって更新される。

- (1) *credit* がゼロまたは負の場合、その *credit* の値に *burst* を加える。
- (2) *credit* が正の場合、それを *burst* で置き換える。

送受信ホスト間に XTP ルータが存在する場合には、そのルータが、送受信ホスト間で通信されている CNTL

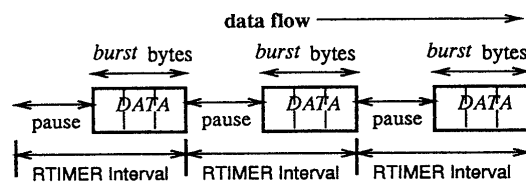


図1: XTP でのレート制御方式

パケットの内容を変更することにより、ルータに対するデータの転送レートを制御することが可能である。また、ルータが独自にRCNTL(Router Control)パケットと呼ばれる制御用のパケットを、データを送信しているホストに送ることにより、そのホストが送信するデータの転送レートを変更することも可能としている。

### 3. レート制御機構を実現する際の考慮点

確立されているコネクション数や転送されるデータ量がダイナミックに変化するネットワークにおいて、レート制御機構を効率的に動作させるには、状況にしたがってレート制御のパラメータを変更する必要がある。そのためには、各コネクションの *rate* および *burst* の値を、適切に管理しなければならない。これらの値を管理する際には、以下の項目が特に重要であると考えられる。

- 受信側での、レートの変更を行なうタイミング
- 受信側での、パラメータ *rate* および *burst* の決定法

これらの項目に特に着目し、様々な状況において最大限にネットワーク性能を引き出せるようレート制御部を実現する必要がある。

### 4. レート制御機構の実現方式

#### 4.1 レートの変更を行なうタイミング

レート変更は、データの受信側によって行なわれるが、その変更を行なうタイミングとしては、以下のような場合が考えられる。

- 受信側の処理能力を越えるデータ転送速度でデータが送られているため、データ転送レートを低くする必要がある場合。
- あるコネクションで通信量が増加したため、その他のコネクションのデータ転送レートを下げることが生じた場合。
- あるコネクションで通信量が減少したため、その他のコネクションのデータ転送レートを上げて通信を行なうことが可能となった場合。

#### 4.2 パラメータ *rate* および *burst* の決定法

本稿では、受信側のホストが常にその受信処理能力を有効に利用できるように転送レートを決定していくことを、前提としている。

レートを計算する際に必要となる値を以下に示す。

- そのホストにおける、最大データ受信処理能力。
- 各コネクションで実行可能な最大転送レート。
- レート変更を行なう時点における、各コネクションでのデータ転送レート。
- 各コネクションに割り当てられている、レート制御のためのパラメータ値。

レートを算出する方式として、本稿では2通りに分類した。1つは、それまでに使用していた *rate* および *burst* の値が不適切なために、パケットロスが頻繁に発生している場合であり、もう1つは、あるコネクションでデータ通信量の増減があったため、各コネクションに対するパラメータの値を変更する必要がある場合である。以下に、各場合において転送レートを定める際の手順を示す。

##### 4.2.1 パケットロスが頻繁に発生している場合

図2で示されるように、受信したデータの一部がほぼ一定間隔で抜け落ちてくる場合、受信側のネットワークインタフェースやプロトコル処理部で処理が行なえない速度でデータを受信し、オーバーランが発生していると考えられる。このような場合は、以下の手順にしたがって現在の値より低めの転送レートを設定し、パケットロスの発生を抑制する必要がある。

- (1) 一定時間内で正しく受信できたデータのバイト数を求める。
- (2) 上記の値から、1秒あたりに受信したデータのバイト数を計算し、これを *rate* とする。
- (3) 次に、シーケンス番号をもとに、連続して正しく受信しているデータの平均バイト数を計算し、これを *burst* の候補とする。
- (4) 計算した *burst* の候補が現在設定されている *burst* の値より小さければ、計算したものを新しい *burst* とする。
- (5) 送信側へ、新しいパラメータ値を含んだ CNTL パケットを送出する。

##### 4.2.2 あるコネクションでの通信量の増減により、レートを変更する場合

ネットワークインタフェースの受信性能やプロトコル処理能力の限界に近い状態などでデータを受信している最中に、別のコネクションにおいて比較的大きなデータの受信が発生すると、オーバーランが発生しパケットロスが生じてしまう。このような状態をできるだけ早く解消するために、そのホストで有するコネクションすべてのレートを一元的に管理し、各コネクション毎のレートを調整する必要がある。

基本的には、各コネクションが公平になるようにレートを決定すべきである。しかし、データ送信能力は送信

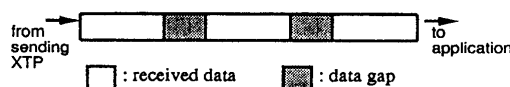


図2: パケットロスが生じている場合の受信データ

側のマシンや、コネクションが張られているパスが通る伝送路の帯域等に依存するため、各コネクションに割り当てたレートができる限り効率良くなるように注意しなければならない。

あるコネクションのデータ通信量が急激に変化したために、パケットロス等が発生したり、受信能力に余裕ができる可能性がある場合には、以下のようにして転送レートの変更を行なう。

- あるコネクションでのデータ受信量が増えた場合
  - (1) データ受信量が増えたコネクションを除いた、現在通信を行なっている各コネクションの実際の転送レートの合計を求める。
  - (2) データ受信量が増えたコネクションで実行可能な最大転送レートがわかっている場合にはその値で、わかっていない場合には現在そのコネクションに与えられている転送レートの値と、上記で求めた現在の転送レートの合計を求める。
  - (3) 求めた合計が、そのホストにおける最大受信処理能力以下であれば、レート変更を行なわない。
  - (4) そうでない場合は、各コネクションのレートの合計が最大受信処理能力以下になるように、各レートの値を同じ割合で低くし、CNTL パケットの送出によって、各コネクションのレートを下げる。

- あるコネクションでのデータ受信量が減った場合
  - (1) 現在通信を行なっている各コネクションの実際の転送レートの合計を調べる。
  - (2) 上記で求めた値と、そのホストにおける最大受信処理能力の差を求める。
  - (3) 通信を行なっているコネクションの中で、レート制御機構により実行可能な最大転送レート以下にレートが抑えられているものに、転送レートの割合にしたがってレートの値を分配し、CNTL パケットの送出によって、各コネクションのレートを上げる。

このような変更を行なった後も、各コネクションの通信レートを監視して、受信能力の最大限を引き出せるよう、適切にレートを制御していく必要がある。

#### 5. まとめ

本稿では、XTP プロトコルでのレート制御におけるレート変更を行なうタイミングや、レート制御に必要なパラメータの決定法について検討した。最後に日頃御指導頂く KDD 研究所 小野所長、浦野次長に感謝します。

#### 参考文献

- [1] "XTP Protocol Definition Revision 3.6," *Protocol Engines Inc.*, PEI 92-10, 11 January 1992.