

XTP プロトコルにおけるレート制御機構の評価

三宅 優 石倉 雅巳 鈴木 健二

国際電信電話（株） 研究所

高速ネットワークに対応する通信プロトコルとして、XTP プロトコルが提案されている。XTP では、オーバランによる受信側でのパケットロスを極力避けるために、送信側が受信側の受信能力に応じた速度でデータを送信するレート制御機構を規定している。XTP におけるレート制御は、バーストとして定めたデータサイズを特定の時間間隔で送出することにより実現しており、高スループットを得るためには、これらを制御する各パラメータを相手の受信能力によって適宜調節することが重要である。本稿では、レート制御における各パラメータとスループットの関係について実験により評価した。また、その検討結果に基づき、レート制御機構の実装方法について検討している。

Evaluation of Rate Control Mechanism in XTP

Yutaka Miyake Masami Ishikura Kenji Suzuki

KDD R & D Laboratories

Ohara 2-1-15, Kamifukuoka, Saitama 356, JAPAN

XTP (eXpress Transfer Protocol) has been proposed as a data communication protocol for high speed network. To achieve high throughput in data communication, XTP utilizes a rate control mechanism, which controls the interval time of transmission data according to the receiving capability of the destination computer. This mechanism prevents over-run errors. This paper describes the relationship between throughput of data transmission and parameters of rate control mechanism in XTP as well as the implementation method of the rate control mechanism.

1 はじめに

伝送路に光ファイバを用いた光通信技術の発展にともない、飛躍的に通信速度が高速化している。LANの世界では、100Mbpsの伝送速度を持つFDDIはすでに実用化されており、さらに高速な伝送速度を達成するギガビットLANに関する研究も行なわれている[1]。また、公衆網においてもATMを用いたB-ISDNの導入が検討されており、伝送速度の高速化は今後に進んでいくものと考えられる。

従来の比較的低速な伝送路を使用しているコンピュータ間通信では、伝送路の速度がスループットを決定する最大の要因であった。しかし、伝送路が高速になるにつれてコンピュータ間通信におけるスループットのボトルネックは、伝送路の速度から計算機の処理能力や通信プロトコルの性能等に移ってきた。これにともない、高速ネットワーク下で高スループットを実現するための様々な研究が行なわれてきている。

このような状況下で、高スループットを得るためのプロトコルも検討されている[2][3]。従来のプロトコルは、低速伝送路において効率良く動作するように設計されており、高速なネットワークで高スループットを得ることが困難となっている。また、ネットワークが高速化されることにより可能となってくるマルチメディアや分散処理などの新しい通信サービスについても、従来のプロトコルで対応することは難しいと考えられる。そこで、ギガビットレベルの高速ネットワークに対応した新しい通信プロトコルが提案されてきている。XTPプロトコルもこのようなプロトコルの1つである[4][5]。

XTPプロトコルでは送信するデータ量を制御するフロー制御機構とは別に、データの転送速度を制御することを目的としてレート制御機構を導入している。ネットワークが高速化することにより、受信能力以上にパケットが到着してしまう事態が考えられるが、レート制御機構はこの事態を抑制するために受信側のデータ受信処理能力等に合わせて送信側のデータ転送レートを調整する機構である。

本稿では、XTPプロトコルのレート制御機構について実際の通信を通じての評価を行ない、その有効性やこの機構の実装に関する検討を評価の結果を基に行なう。

2 XTP とレート制御機構

2.1 XTPプロトコルの概要

XTPプロトコルは、OSI参照モデルにおけるネットワーク層とトランスポート層を一体化し、処理の簡略化と高速ネットワークに対応した機構によってプロトコル面から通信の高速化を目指すものである。このプ

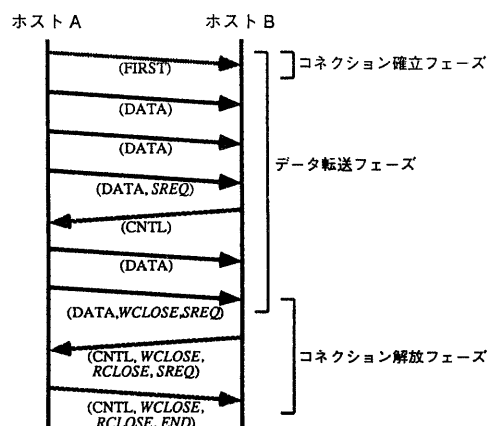


図1: XTPプロトコルにおける基本的なシーケンス

ロトコルは、リアルタイム転送、マルチキャスト、バルクデータ転送といった各種サービスを提供する。また、レート制御、選択再送、複数アドレス形式のサポートといったメカニズムを持っている。

図1に、XTPプロトコルでデータ転送を行なう場合の基本的なシーケンスを示す。XTPのパケットには、運ぶデータの種類や情報により、FIRST、DATA、CNTL (Control)、PATH、ROUTE、RCNTL (Router Control) といった名前が付けられている。また、SREQ (Status requested)、DREQ (Delivery status requested)、WCLOSE (Writer closing)、RCLOSE (Reader closing)、END等のオプションビットフィールドが存在し、これらのフラグを使用して接続が確立されているコンピュータ間の情報交換や接続解放が行われる。

接続確立は、FIRSTパケットの送信と同時にに行なわれ、受信側の応答を待つことなくDATAパケットによるユーザデータの送信が可能となる。接続確立後は、DATAパケットにより相互にデータを送信したり、CNTLパケットによる接続の状態情報の交換を行なう。CNTLパケットは、現在の状態を通信相手に伝えるために常に送信可能である。また、通信相手の状態を得たい場合には、SREQまたはDREQフラグを設定したパケットを送信し、CNTLパケットの送信を通信相手に要求することができる。

接続の解放は、各パケットのヘッダに含まれるWCLOSEフラグまたはENDフラグを設定したパケットを送信することにより開始される。フラグの設定方法によりさまざまな接続解放手順を取ることが可能である。図1の場合では、WCLOSEフラグを設定したパケットが送信されることによりハンドシェイクによる接続解放を開始している。

2.2 レート制御の役割

高速ネットワークを介したコンピュータ間通信を行なう場合、そのネットワークの伝送速度に見合ったスループットでデータの送受信を行うことが期待される。そのためには、受信側で大量の受信バッファを用意し、連続的にデータを受信できるようにする必要がある。ネットワークが非常に高速な場合にデータを連続して送信すると、送信側のデータ送信能力が受信側のデータ受信処理能力以上であったり、受信側で複数の送信相手から同時にデータを受信した場合には、コンピュータのデータ受信処理能力以上にデータが到着してしまう可能性がある。受信側では、処理能力の範囲内でしかパケットを受信することができないため、処理できないパケットは破棄されてしまう。この現象は、一般にオーバーランと呼ばれている。

高速ネットワーク下でスループットを向上させるには、データ誤りの発生をできるだけ少なくして誤り処理の起動を最小限に抑え、送受信処理をスムーズに行なうことが必要である。一般に高速ネットワークでは、光ファイバーなどを通信媒体として使用しているために、ビット誤り率は極めて小さい。よって、オーバーランによるパケットロスが誤り発生の主原因となるため、この発生を抑えることが重要である。

レート制御は、このオーバーランによるパケットロスを防ぐものである。レート制御機構により、送信側のデータ転送速度は受信側で処理可能な速度に調整されるため、オーバーランの発生が低く抑えられることになる。これにより、誤り発生が少ない状態でデータの送受信が可能となり、通信を行なっているコンピュータの性能に見合ったスループットを得ることが可能となる。したがって、高速ネットワーク下で高スループットを得るためには、レート制御機構の役割が非常に重要になってくると考えられる。

2.3 XTP プロトコルにおけるレート制御

本節では、XTP プロトコルにおけるレート制御機構について簡単に説明する。

表 1 にレート制御機構で使用する変数一覧を、図 2 に XTP プロトコルによるレート制御機構が動作している場合のパケット送信の様子および各時点での変数 *credit* の値の変化を示す。ここで使用される変数は各コネクション毎に管理され、他のコネクションでは異なる値を持つことになる。

送信側がデータを送信する場合、送信側の内部変数である *credit* 値が正である必要がある。この値が負の場合、パケットを送信することはできない。*credit* 値が正の場合、パケットを送信し、送信したユーザデータのバイト数を *credit* から減らす。

credit 値は RTIMER 間隔で更新される。この値が負

であると送信ができなくなるが、RTIMER 間隔で更新されることにより値が正になり、送信を再開できる。更新の手順は以下の通りである。

- (1) *credit* 値が 0 または負の場合、*credit* 値に *burst* の値を加える。
- (2) *credit* 値が正の場合、*credit* = *burst* とする。

以上の動作により、送信されるデータの最高スループットは *rate* (bytes/sec) となり、1 バーストで送信されるデータのバイト数は *burst* バイト以下になる。

rate および *burst* の値は、受信側から送られてくる CNTL パケットにより更新される。CNTL パケットには *rate* および *burst* というフィールドが含まれており、これらのフィールドに受信側が希望する *rate* と *burst* の値が含まれる。インターバルタイムである RTIMER の値は、*burst*/*rate* で求められる。

burst の値が 0 の場合には、*rate* および *burst* の値に関係なく送信側はデータを送信することが可能である。*rate* の値が 0 の場合には、DATA パケットの送信を停止しなければならない。しかし、CNTL パケットは送信可能である。

XTP プロトコルの仕様では、FIRST パケットの送信後に連続して DATA パケットを送信することが可能である。この場合、受信側における最適なパラメータ値を得ることなくデータを送信する必要があるため、送信側ではそのホストが保持しているデフォルトのパラメータを用いてデータを送信する。これらの値は、FIRST パケットに含まれる *rate_req* および *burst_req* フィールドにセットされるため、受信側ではこれらのフィールドを調べることで現在送信側で使っているパラメータ値を知ることができる。

その他に、送受信ホスト間に存在する中間ノード(ルータ)から送信側の *rate* や *burst* を変更する手順も規定している。これは、その中間ノードを通過する CNTL パケットの *rate* および *burst* フィールドを書き換えたり、その中間ノードから RCNTL(Router Control) パケットを送信側に対して送ることにより行なわれる。この機能は、データ送受信パスのスループットのボトルネックがデータ受信側の能力ではなく、データを中継する中間ノードでのデータ処理能力によって決定される可能性があるために提供されている。

3 レート制御機構の評価

XTP のレート制御機構を実装した通信プログラムを作成し、レート制御機能がスループットに与える影響についての評価を行なった。本章では、その結果を示す。

表 1: レート制御機構で使用する変数

変数名	内容
rate	1 秒あたりに送信可能なデータのバイト数 (bytes)。
burst	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 (bytes)。
RTIMER	リフレッシュタイム。値は、 $burst/rate$ で、インターバルタイムの間隔を設定する。
credit	RTIMER がタイムアウトするまでに送信可能な残りバイト数。送信側で使用する内部変数。

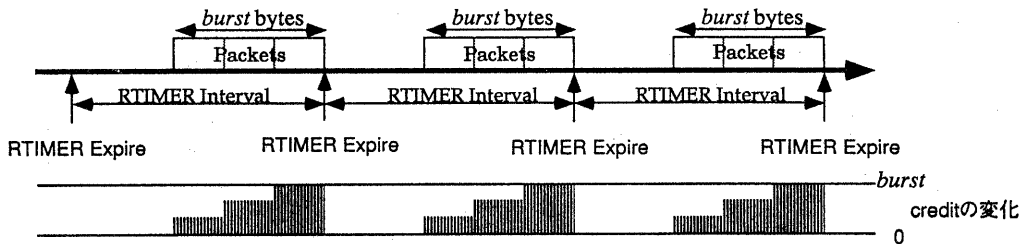


図 2: レート制御機構が動作しているときのパケット送信の様子と credit 値の変化

3.1 評価方法

レート制御機構に対する評価は、レート制御のためのパラメータ *rate* および *burst* 値と、2 台のワークステーション間で通信を行なった時のスループットおよび誤り制御機構の動作回数の関係を調べるにより行なった。ワークステーションは、伝送速度が 10Mbps のイーサネットに接続した。

評価用に作成したプログラムは、TCP/IP プロトコル群の UDP 上で動作する。UDP プロトコルはコネクションレス型プロトコルであり、誤り制御やフロー制御などの処理は行なわれない。今回の目的は XTP プロトコルの評価ではなくレート制御機構の有効性を評価することであるため、フロー制御機構等の影響がない UDP プロトコルを下位プロトコルとして使用した。

通信プログラムは、レート制御のためのパラメータ *rate* および *burst* の値にしたがってデータを連続して送信するように作成した。データ転送中にパラメータの変更は行なわない。また、フロー制御機構は実装していない。パケットロスが生じた場合には、*go-back-n* にしたがってパケットを再送するように誤り制御機構を実装した。評価の対象がスループットの変化であるため、UDP パケットで送信されるユーザデータのサイズは、イーサネットでの最大パケットサイズになるように 1454 バイトとした。

評価を行うために使用したシステムの構成を図 3 に示す。評価は、受信処理能力差がある 2 つのコンピュータ (Sun3/260, SPARCstation 1) に対してそれぞれデータを送信した場合について行った。

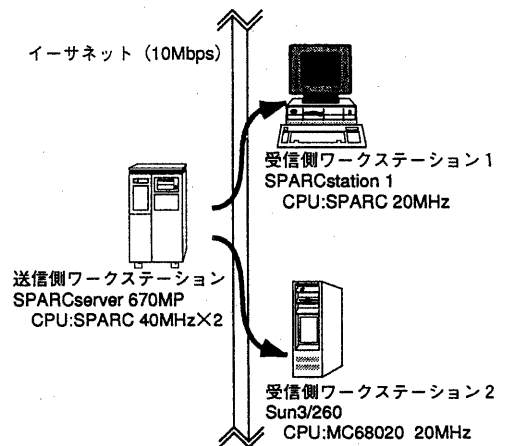


図 3: 評価に用いたシステム構成

3.2 パラメータの値とスループットの関係

パラメータ *burst* の値を固定にし、*rate* の値を 100,000 (bytes/sec) から 100,000 ずつ増加させた時のスループットを計測した。パラメータ *burst* の値は 1 パケットのデータサイズ (1454 バイト) の倍数となるように、SPARCstation 1 に送信する場合は、1454、14540、43620、72700、101780 の各バイト数を、Sun3/260 に送信する場合は、1454、7270、14540、29080 の各バイト数を用いた。SPARCstation 1 に送信した場合の結

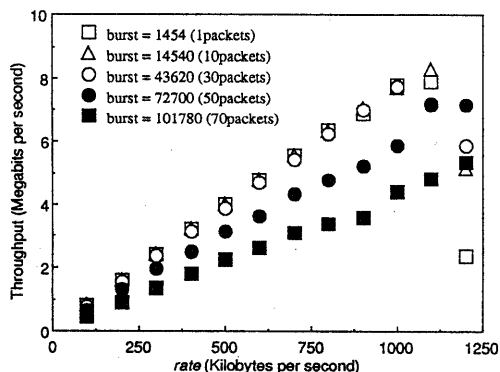


図 4: レート制御のためのパラメータとスループットの関係 (受信側 WS:SPARCstation 1)

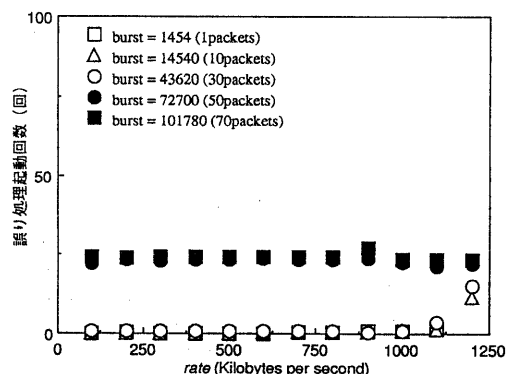


図 6: レート制御のためのパラメータと誤り処理起動回数の関係 (受信側 WS:SPARCstation 1)

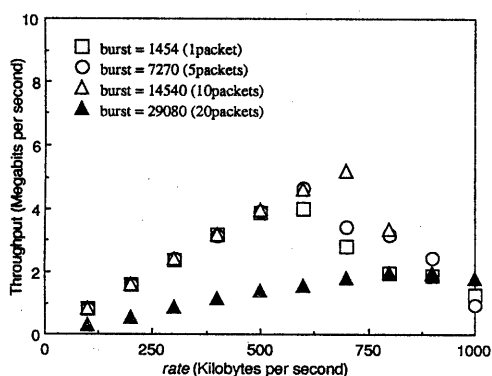


図 5: レート制御のためのパラメータとスループットの関係 (受信側 WS:Sun3/260)

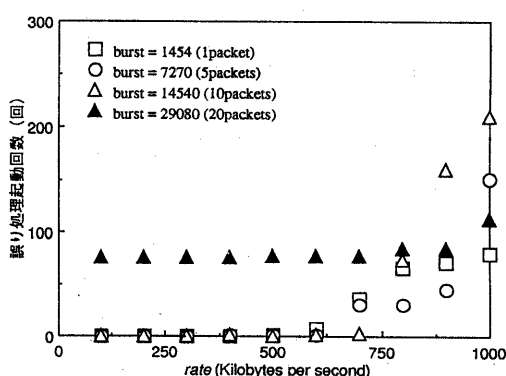


図 7: レート制御のためのパラメータと誤り処理起動回数の関係 (受信側 WS:Sun3/260)

果を図 4 に、Sun3/260 に送信した場合の結果を図 5 に示す。

データ受信側が SPARCstation 1 の場合、burst 値が 43620 以下であれば、rate で指定された値に近いスループットが得られた。イーサネットの伝送速度は 10Mbps であるので、それ以上の速度でデータを送信することは不可能であるが、今回は最高約 8Mbps 以上のスループットを得た。最高のスループットを得た状態からさらに rate の値を増加していくと、スループットが大きく低下している。

burst が、72700、101780 といった比較的大きな値になると、rate で指定した転送速度から得られるスループットよりも、少しずつ低いスループットとなった。rate の値が小さい段階でもこの現象が発生しているため、burst の値が不適切な場合は高スループットを

得ることが困難であることがわかる。

データ受信側が Sun3/260 の場合は、最高スループットは約 5.2Mbps となった。最高スループットを得た地点からさらに rate を増やしていくと、徐々にスループットが低下していった。burst 値が 29080 であっても、SPARCstation 1 で burst 値が 72700、101780 の場合と同じくスループットが低下している。これらより、受信コンピュータの処理能力差にスループットは依存し、レート制御で使用するパラメータ値の最適点も異なることが確認された。

3.3 パラメータの値と誤り処理起動回数の関係

パケットロスが発生したために行なわれる誤り処理は、高スループットを達成する上で大きな障害となる。そこで、図4および図5のスループット値を計測する際に、同時に誤り処理の起動回数も計測した。その結果を、データ受信側がSPARCstation 1の場合は図6に、データ受信側がSun3/260の場合は図7に示す。図における誤り処理の起動回数は、1000パケットを正確に受信するために行なわれた誤り処理機構の動作回数で表してある。この回数は、受信中にオーバーランが発生した回数とほぼ同じであると考えられる。

図4および図5と比較してみると、これらの図においてほぼ $rate = \text{スループット}$ の関係が成立しているところでは、ほとんど誤り処理が行なわれていない。スループットが $rate$ で指定した値より低いところでは、誤り処理が多く行なわれている。よって、スループットが低下しているところでは、誤り処理がかなりの頻度で行なわれていることが確認された。

4 レート制御機構の実装に関する考察

XTPにおけるレート制御の仕様では、 $rate$ および $burst$ のパラメータをもとにして、どのようにデータを送信するかメカニズムと、そこで必要となるパラメータをデータ受信側から送信側に知らせる手順のみが決められている。本章では、XTPプロトコルのレート制御機構を実装する際に重要な問題点について検討する。

4.1 レート制御機構を実装する際の基本方針

本節では、レート制御機構を実装する際の前提として重要な項目を挙げ、それらについて考察する。

(1) そのホストでの受信処理能力をできる限り使う

レート制御におけるパラメータ $rate$ の値が、オーバーランが発生しないような十分小さな値であれば、誤りが発生しないスムーズなデータ転送が可能である。しかし、レート制御機構の目的は高スループットを得ることであり、受信側で処理可能な最高の転送レートでデータを送信することが望ましい。したがって、データ受信処理能力に余裕があるにもかかわらずレート制御機構によりスループットが抑えられることがないよう、レート制御機構を実装すべきである。

(2) 各コネクション間で公平性を保つ

XTPのレート制御における各種パラメータの管理は、各コネクション毎に行なわれるが、複数のコネク

ションが存在する場合には各コネクション間が公平になるようにパラメータの値を決められなければならない。コネクション毎にパラメータの値を管理していると、プライオリティの低いコネクションが大きな $rate$ の値を維持し続けたり、すでに大きな $rate$ の値でコネクションが確立されている状態で新たにコネクションを確立した場合に、小さな $rate$ 値しか得ることができないといった問題が発生してしまう。したがって、他のコネクションと連動してパラメータの値を決める必要がある。

4.2 パラメータの変更を行なうタイミング

ホスト間で通信を行っている時には、ホスト、ネットワーク、通信を行なっているホスト間の通信経路に存在する中間ノード等の状態は時々刻々変化している。したがって、レート制御機構での最適なパラメータ値もこれらの状態に適応させて変えていく必要がある。本節では、パラメータ $rate$ および $burst$ を変更するタイミングとして以下のような場合を挙げ、それぞれの場合について考察する。

(1) FIRSTパケット受信時

XTPでは、FIRSTパケットの受信によりコネクション確立が行われるため、送信側はFIRSTパケットの送信後、続けて複数のDATAをパケット送信することができる。この時点では、受信側からのレート制御を行っていないために、好ましくない速度でデータが送られてくる可能性がある。このような場合に対処するために、FIRSTパケットを受信した際にはその中にある $rate_req$ および $burst_req$ フィールドの値を確認し、その値が不適切な場合にはCNTLパケットを送信し、 $rate$ および $burst$ の値を変更する必要がある。

(2) 現在のパラメータでは、送受信処理能力を越えている場合

データを受信している状態で、頻繁にパケットロスによる誤りが発生している場合は、レート制御に用いるパラメータの値が不適切なために、オーバーランが発生していると考えられる。このような場合には、パラメータの再計算を行ない、パケットロスが生じないようなデータ転送レートでデータを送信するようにすべきである。

また、別のコネクションでのデータ転送速度が増加した場合も、パラメータの再計算を行う必要があると考えられる。

(3) 受信処理能力に余裕があり、データ転送レートの増加が可能な場合

複数確立しているコネクションの中で、高速にデータ転送を行っていたあるコネクションでデータ転送が中断すると、そのホストとしての単位時間当たりのパケット受信数は減ることになる。したがって、データ受信処理能力に余裕ができる。この時点で、データ転

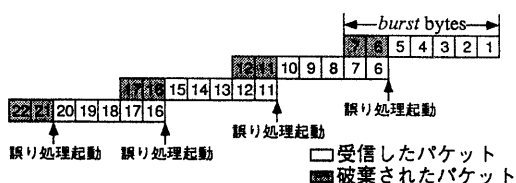


図 8: *burst* 値が適切な値を超えている時の受信状態

送がレート制御機構により抑制されているコネクションがあれば、そのコネクションでのデータ転送レートを向上させて、スループットを向上させることが可能である。よって、各コネクションでのデータの受信状況を常に監視し、受信能力に余裕が生じた場合には各コネクションで使用されているレート制御のためのパラメータの値を再計算する必要がある。

4.3 パラメータ *rate* および *burst* の計算法

本節では、パラメータを変更する際の *rate* および *burst* の計算法について検討する。

4.3.1 パケットロスが頻繁に発生している場合

パラメータ *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は、即座にパラメータの値を適切な数値に変更しなければならない。

パラメータ *burst* の値が大き過ぎる場合は、図 6 で *burst* 値が 72700、101780 の場合および図 7 で *burst* 値が 29080 の場合などで示されているように、*rate* の値が十分小さい値であってもパケットロスがコンスタントに発生する。受信側で連続して受信できるパケット数に制限があり、その値以降のパケットは受信側で破棄されてしまうためにこのような結果になったと考えられる。

この場合、図 8 に示されるように、連続して受信できたところからデータ転送が再開されるという状態が繰り返される。したがって、連続して受信可能なパケット数は、以下のような式から求めることができる。

$$\text{連続受信可能なパケット数} = \frac{\text{受信パケット数}}{\text{誤り処理起動回数}}$$

ここで、受信パケット数および誤り処理起動回数は単位時間内に計測したものである。評価に使用されたコンピュータである SPARCstation 1 の場合、図 6 に示されているように、*burst* 値が 72700、101780 の場合には 1000 パケット受信するために約 24 回誤り処理を起動している。したがって、約 40 パケットの連続受信が可能である。Sun3/260 の場合は、*burst* 値が 29080 の場合に 1000 パケット受信するために約 75 回の誤り処

理を起動しているので、約 13 パケットの連続受信が可能となる。*burst* 値が不適切な場合には、このようにして連続受信可能なパケット数を計算し、これをもとにしてパラメータを調整すればよい。

burst 値が不適切でないにもかかわらず、誤り処理が多く発生している場合には、*rate* 値が大き過ぎると考えられる。その場合は、単位時間当りにの受信したすべてのパケット数から、受信したデータの転送レートを計算し、この値をパラメータ *rate* を決める基準とすればよい。

4.3.2 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

データを受信側において、受信処理能力に余裕があるにもかかわらず転送レートがレート制御機構により抑制されてしまうことが考えられる。オーバランが発生している場合は、各コネクションで発生した状況を知ることができるが、受信処理能力に余裕がありデータの転送速度をあげることが可能であるかを知ることができない。したがって、ホスト全体のコネクションの受信状況を統一的に管理して、受信能力を最大限に利用してデータ受信を行なえるような機構が必要である。

各コネクションでの受信状況を一括管理してパラメータ値を決めるには、以下のような情報が必要である。

- そのホストでの最大受信能力
- 各コネクションでの現在のパラメータ値
- 各コネクションでの現在のスループット

そのホストでの最大受信能力は、パケットロスが生じたためにパラメータ値を計算する際に推定することが可能である。

パラメータ値の更新は以下の手順にしたがって行う。

- (1) パラメータ値を変更するタイミングにおいて、各コネクションでの現在のスループットの合計とそのホストでの最大受信能力と比較する。
- (2) 合計が最大受信能力より顕著に低い値であれば、各コネクションでの現在のスループット値とそのコネクションで設定されているパラメータ *rate* の値を比較し、ほぼ同じ値であれば *rate* 値によりスループットが抑えられていると考えられるので、そのコネクションをパラメータ変更を行なう候補とする。
- (3) 候補となった各コネクション間で公平となるようにコネクション毎の *rate* 値を計算する。それらの値の合計と他のコネクションの現在のスループッ

トの合計がそのホストの最大受信能力となるように値を決定する。

- (4) 新たにパラメータが変更された各コネクションは、送信相手に対し新しいパラメータを設定したCNTLパケットを送信して、送信側のパラメータを更新を行なう。

5 まとめ

本稿では、XTPプロトコルにおけるレート制御機構について検討した。

まず、XTPのレート制御において重要であるパラメータ *rate* および *burst* の値にしたがってデータを転送する評価用通信プログラムを作成し、パラメータの値と通信スループットの関係を明らかにした。その結果、高スループットを得るには転送レートを適切な値に制御してデータを送信する必要があること、ホストにより受信データ処理能力に差があるため、ホスト毎に最適なパラメータ値が存在することが確認された。

次に、レート制御機構を実装する際に必要な問題点について検討した。その中で、レート制御で 사용되는パラメータ *rate* および *burst* 値を変更するタイミングおよびこれらの値を計算するための方法を明らかにした。

今回の評価では、伝送速度が10Mbpsのイーサネットでは評価を行ったが、今後はさらに高速な伝送媒体を用いた評価を行い、高速ネットワークにおけるレート制御機構の効果的な実装方式等について検討を続けていく。

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] E. A. Arnould, et al, "The Design of Nectar: A Network Backplane for Heterogeneous Multicomputers," *Technical Report*, CMU-CS-89-101, School of Computer Science, Carnegie Mellon University.
- [2] W. A. Doeringer, D. Dykeman, M. Kaiserswerth, B. W. Meister, H. Rudin and R. Williamson, "A Survey of Light-Weight Transport Protocols for High-Speed Networks," *IEEE Trans. Commun.*, Vol. 38, pp. 2025-2039, November 1990.
- [3] T. F. La Porta and M. Schwartz, "Architectures, Features, and Implementation of High-Speed Transport Protocols," *IEEE Network Magazine*, pp. 14-22, May 1991.
- [4] "XTP Protocol Definition Revision 3.6," *Protocol Engines Inc.*, PEI 92-10, 11 January 1992.
- [5] W. T. Strayer, B. J. Dempsey and A. C. Weaver, "XTP: The Xpress Transfer Protocol," *Addison-Wesley Publishing Company, Inc.*, ISBN 0-201-56351-7, 1992.
- [6] 三宅、石倉、鈴木、"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," 情報処理学会第45回全国大会、5V-04、1992.