

伝送誤りを伴う回線上での SSCOP を用いた TCP トラヒックの高性能化に関する一検討

原 歩美[†] 井戸上 彰^{††}
加藤 聰彦^{††} 鈴木 健二^{††}

携帯端末によるデータ通信増加し、より高速な伝送速度が求められている。しかし、無線回線では有線回線に比べてビット誤り率が大きいため、主として輻輳によるパケット紛失を想定した TCP 誤り回復手順のみでは、効率的な TCP 通信を実現できないことが知られている。これに対して筆者らは、TCP の高速化の方法として、ATM 網で実際に使用されており、効率的な選択再送機能を有する SSCOP (Service Specific Connection Oriented Protocol) を無線区間のデータリンクプロトコルとして採用し、さらに、SSCOP にパケットのセグメンティング・リアセンブリング機能を追加するというアプローチの検討を行っている。本稿では、提案する方式をシミュレーションを用いて詳細に評価した結果について述べる。具体的には、TCP のパケットサイズが 1500 バイト、セグメントサイズが 128 バイトの場合、TCP のみを使用するとまったく通信が不可能であったビット誤り率 $1.0E-04$ において、本アプローチでは 2Mbps 回線上で約 1.2Mbps のスループットを得ることができた。

A Study on High Speed TCP Communication using SSCOP over Radio Link with Packet Losses

AYUMI HARA,[†] AKIRA IDOUE,^{††} TOSHIHIKO KATO^{††}
and KENJI SUZUKI^{††}

According to the increase of mobile data communication, the high performance of the mobile TCP/IP communication is required. However, the congestion control of conventional TCP has a problem that the packet losses over radio link invokes the slow start and congestion avoidance and this decreases the TCP throughput. Against this problem, we propose to use SSCOP, which is used in ATM communication and has a powerful selective retransmission, as the data link protocol over lossy radio links. We also propose to introduce the segmenting and reassembling functions into SSCOP in order to decrease the size of packets over radio links. This paper describes the details of our proposal and the performance evaluation through the software simulation. As a result, our approach has brought 1.2 Mbps throughput over 2.0 Mbps link with $1.0E-04$ bit error rate, where the conventional TCP cannot make communication at all.

1. はじめに

携帯型コンピュータや超小型の携帯情報端末機器が普及するにつれて、モバイルコンピューティングによるデータ通信が増加し、より高速な伝送速度が求められている。このような移動体通信システムを利用したデータ通信においても、TCP をベースとした通信アプリケーションが広く利用されると考えられる。無線回線では通常の有線回線と比較すると伝送誤りの発生頻度が大きいと考えられる。しかし、通常の TCP で

は伝送誤りによるパケット損失は小さく、パケット損失はネットワークの輻輳により生ずると仮定されている。このため、無線回線上では、主として輻輳によるパケット紛失を想定した TCP 誤り回復手順のみでは、効率的な通信を実現できないことが知られている¹⁾。このため、無線回線上の TCP 通信の高速化に関する検討が広く行われている²⁾。文献²⁾では、TCP 通信の高速化の手法として、TCP 自身の機能を変更する方式、無線区間と有線区間で TCP コネクションを分離する方式、無線区間のデータリンクレイヤの専用の再送プロトコルを導入する方式の 3 つについて、性能比較を行っている。その結果、無線区間に専用の選択再送型データリンクプロトコルを導入する方式が最も性能が高いという結論を得ている。その理由としては、この方式では無線区間のみで再送が行われ、TCP 上

[†] 電気通信大学大学院情報システム学研究所
The University of Electro-Communications
^{††} 株式会社 KDD 研究所
KDD R&D Laboratories, Inc.

では重複 ACK やタイムアウトによる再送が行われず、輻輳ウィンドウサイズが減少しないためである。また、無線区間におけるパケットの誤り率を減少させるためには、パケットサイズを小さくすることが1つの方法である。例えば、PHS (Personal Handy-phone System) 用のデータ通信で使用されている PIAFS (PHS Internet Access Forum Standard)³⁾ においても 80 バイトのフレームが使用されている。

そこで本研究では、伝送誤りを伴う無線回線上での TCP の通信高性能化を図るために、ATM 網で実際に使用されており、効率的な選択再送機能を有する SSCOP (Service Specific Connection Oriented Protocol) を無線区間のデータリンクプロトコルとして採用し、さらに、SSCOP にパケットのセグメンティング・リアセンブリ機能を追加するというアプローチを採用する。さらにシミュレーションを用いてこのアプローチの性能評価を行う。本稿では、これらの結果について示す。以下、2章でSSCOPの概要について、3章で本研究で採用したアプローチと性能評価のためのシミュレーションの概要についてそれぞれ述べる。さらに、4章で性能評価の結果について述べ、5章でこれらに対する考察を行う。

2. SSCOP 概要⁴⁾

SSCOP は AAL (ATM Adaptation Layer) タイプ 5 の上位にインタフェースする確認型データ転送プロトコルであり、以下に示す機能を有する。

(1) ユーザデータの順序保存

送信側は、上位レイヤから渡されたユーザデータに順序番号を含む SSCOP トレイラを付加し、SD (Sequenced Data) PDU (Protocol Data Unit) を作成して送出する。受信側は、順序番号に従ってユーザデータを上位レイヤに渡す(デリバリーする)。

(2) SD の送達確認

以下の2つのPDUによりSDの送達確認を行う。

- 送信側から送られた POLL PDU に対して、受信側が送出する STAT (Solicited Status) PDU
- 誤った順序で SD が受信された場合に、受信側が送出する USTAT (Unsolicited Status) PDU

(3) 選択再送による誤り回復

受信側は、受信した SD の順序番号を検査することで、SD の紛失を検出する。紛失した SD の順序番号は、STAT または USTAT によって送信側へ通知する。この際、連続して紛失した SD のグループ(紛失 SD グループ)とそれに続く受信した SD のグループ(受信 SD グループ)

を順に書き込む。送信側は、紛失を通知された SD のみを選択的に再送する。

(4) フロー制御

送信側は、STAT-USTAT 等で受信側から通知されるウィンドウの上限値に基づいて、フロー制御に用いるウィンドウを決定し、その範囲内で SD を送出する。

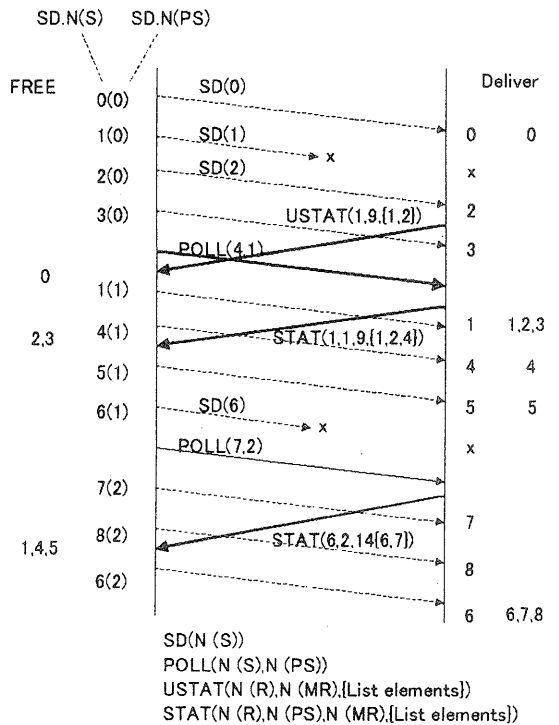


図1 通信シーケンス例

図1にSSCOPの通信シーケンス例を示す。SSCOPでは、SDとPOLLに対してそれぞれ独立な順序番号を設けており、これらを用いて以下のように確認型のデータ転送を行う。

- (1) 正しい順序番号 $N(S)$ を持つ SD (図中の SD(0)) が転送されると、受信側は上位レイヤにデリバリーする。
- (2) 紛失した SD(1) は、次の SD(2) の受信により検知され、USTAT を用いて送信側へ通知される。USTAT には、次に受信すべき SD の順序番号 $N(R)$ (=1)、受信ウィンドウ $N(MR)$ (=9) (ウィンドウサイズ 8 を仮定している)、紛失 SD グループを示す List elements (=1,2) が書き込まれている。送信側はこの USTAT 受信により、送達確認された SD(0) の解放、ウィンドウの更新、紛失を通知された SD(1) の再送を行う。
- (3) 送信側は、一定時間毎あるいは一定の SD 送出

回数毎に、POLL を送出する。POLL は次に送出される SD の N(S) と、自身の順序番号 N(PS) を持つ。POLL(4,1)を受信した受信側は、次に受信すべき SD の順序番号 N(R)(=1)、対応する POLL 順序番号 N(PS)(=1)、受信ウィンドウ N(MR)(=9)、紛失 SD グループ (= [1,2]:SD(1) を表す) と受信 SD(=[2,4]) を示す List elements=1,2,4 を含む STAT を、即座に送出する。

送信側はこの STAT 受信に対して、紛失 SD グループ (= [1,2]) に対する再送処理、受信 SD グループに対応する SD(2).SD(3) の解放、ウィンドウの更新を行う。ただしこの例では、SD(1) が POLL(4,1) 送出後に再送されているため、新たな再送は行わない。これは、SD 送出時点で保持された POLL 順序番号と、STAT 中の N(PS)(=2) が SD(6) 送出時の POLL 順序番号 (=1) より大きいため、要求された SD(6) を再送する。

3. 提案するアプローチとシミュレーションの概要

3.1 提案アプローチの概要

本稿で提案するアプローチを用いたネットワークの構成例を図 2 に示す。この図に示すように、ネットワークの有線区間と無線区間の境界において、ゲートウェイが設置される。このゲートウェイは有線区間に対しては、有線用のメディアアクセス制御 (MAC) プロトコルである PPP を提供し、無線区間に対しては、セグメンティング付きの SSCOP と無線回線用の MAC プロトコルを提供する。さらに両者を中継するリレー機能を有し、有線・無線区間の IP パケットの交換を行う。無線区間においては、IP パケットは定められた最大パケット長に従って、セグメンティングし、それぞれのデータが SSCOP を用いて転送される。パケットの伝送誤りが生じた場合は、SSCOP の選択再送機能を用いて、誤った (セグメンティング後の) パケットのみが再送される。受信側では IP パケットがリアセンブリングされ、IP に渡される。このような方式により、伝送誤りの大きい無線区間のみに対して、転送するパケットサイズを小さくし、効率的な再送方式を導入することが可能となる。

3.2 シミュレーションの概要

本稿で提案するアプローチの性能を評価するために、無線区間に着目して以下のようなシミュレーションを行う。パケット網用シミュレータとしては、カリフォルニア大学バークレー校が開発された REAL⁵⁾ を利用する。REAL はホストやルータ/交換機などのプロトコル処理を模擬するノードと、ノード間を接続する回線を模擬するチャンネル、あらかじめ規定したパラメータによってパケット転送を制御するシミュレーション・エンジンから構成される。また、TCP の DT

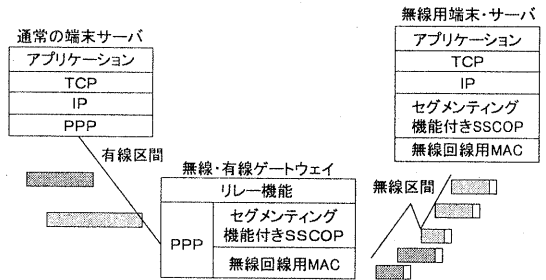


図 2 提案するアプローチを用いたネットワーク構成

パケットを連続的に送出するソースノードと、DT パケットを受信して、その応答 (ACK) パケットを返送するシンクノードが用意されている。TCP ソース/シンクノードは、指定されたウィンドウサイズの範囲内で連続的に TCP セグメントの送信/確認応答の返信を行う。チャンネルは、ある特定の帯域幅と遅延を持つ伝送路を模擬することができ、複数のノードとその間を接続するチャンネルで接続することによって、様々なトポロジーのネットワークをシミュレーションすることができる。

シミュレーションのために、新たに SSCOP をサポートする SSCOP ノード、SSCOP ノードが転送するデータサイズをセグメントサイズに従ってセグメンティングするノード、セグメンティングされたデータをリアセンブリングするノードを実装した。この内、SSCOP ノードについては、2Mbps の伝送速度と指定したビット誤りを発生する伝送路を想定したチャンネルで接続した。シミュレーションのプログラムモジュール構成を図 3 に示す。

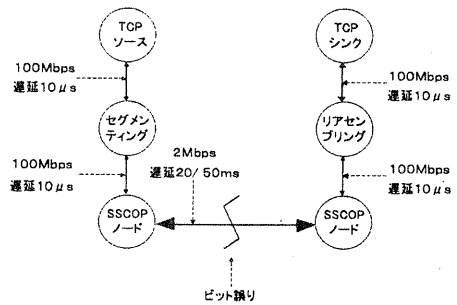


図 3 シミュレーションのプログラムモジュール構成

シミュレーションで使用したパラメータは以下の通りである。伝送遅延を 20ms と 50ms とし、TCP のセグメント長はヘッダを含め 1500 バイトとし、10000 パケットを送出する。TCP ウィンドウサイズ (TCPWIN/パケット数換算) は、連続的にデータ通信が可能ないように応答遅延時間の 2 倍相当以上に設定する。セグメンティング機能においては、パケットのセ

グメントサイズを 2048, 1024, 512, 256, 128 バイトとする。また、SSCOP については、ウィンドウサイズとしてはデータ転送に対して十分大きな値 (512) を使用し、POLL PDU は 16 個の SD PDU の送信のたびまたは 10 秒経過する毎に送出することとした。これらのパラメータ値を用いてシミュレーションを行い、TCP レベルのスループット、再送パケット数、SSCOP による再送を含めた全ての送信パケットによる回線の利用率などを計測する。なお、各パラメータ値においてシミュレーションを 3 回行い、その平均をとることとした。

なお比較のために、SSCOP 機能を外し、TCP ソースと TCP シンクの間 TCP コネクションを設定し、TCP のみを使用した場合のシミュレーションも行うこととした。伝送遅延とウィンドウサイズは TCP に SSCOP を用いた場合と同じとし、伝送路に任意のビット誤りを発生する。パケット数も同じ条件で 10000 パケット、TCP のパケットサイズを 1500 バイトでシミュレーションを行う。

4. 評価結果

4.1 スループットの評価結果

伝送遅延を 20ms、TCP ウィンドウサイズを 16 に設定した場合において、各セグメントサイズ毎に、ビット誤りに対するスループットを測定した。その結果を図 4 に示す。比較のために、この図には TCP のみを使用した場合の結果も示してある。同様に、伝送遅延 50ms、TCP ウィンドウサイズ 36 の場合におけるスループットの結果を図 5 に示す。この場合は TCP のみを用いた通信において、TCP のパケットサイズが 1500 バイトと 500 バイトの双方の結果についても示してある。

これらの測定結果から以下のような結果を得た。

- TCP のみを使用した場合は、ビット誤り率が $5.0E-07$ より大きくなると、急速にスループットが低下するのに対して、SSCOP を用いた場合は誤り率が增大してもスループットの低下を押さえることが可能であった。
- SSCOP を用いた場合においては、ビット誤り率が $1.0E-07$ から $5.0E-06$ までは、各セグメントサイズ間において、スループットはほぼ同一であった。ただし、ビット誤りがほとんどないと考えられる $1.0E-06$ 以下においては、セグメントサイズが小さい場合より大きい場合の方がわずかにスループットがよい。
- しかし、ビット誤り率が $1.0E-05$ 以上になると従って、サイズ毎にスループットの差が出始めた。ビット誤り率が $1.0E-04$ で、セグメントサイズが 128 バイトの場合は、スループットが約 1Mbps となり、2048 バイトでは約 0.3Mbps となる。ビット

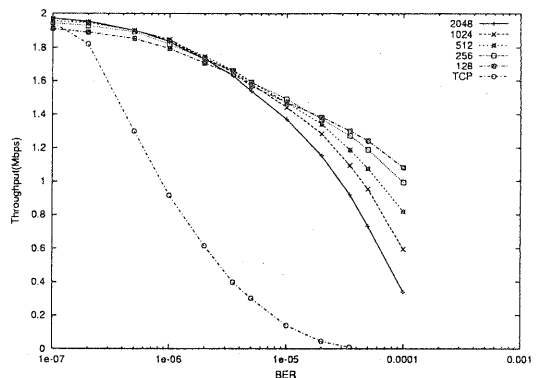


図 4 誤り率に対する TCP スループット (伝送遅延 20ms, ウィンドウサイズ 16)

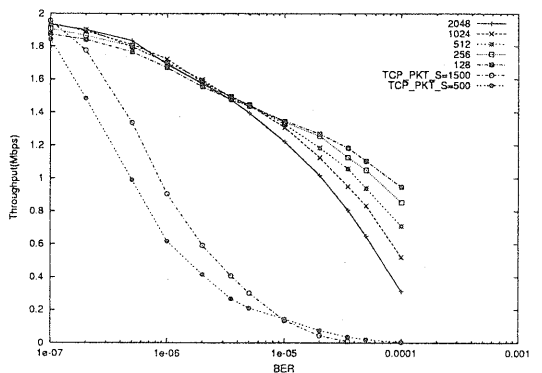


図 5 誤り率に対する TCP スループット (伝送遅延 50ms, ウィンドウサイズ 36)

誤りが発生しないと比べると、128 バイトでは約半分減少し、2048 バイトでは約 85% も減少する。セグメントサイズが大きくなるほど減少している。

- TCP のみを使用し、そのパケットサイズを 500 バイトとした結果では、1500 バイトの場合よりも低いスループットしかえられなかった。

TCP レベルのウィンドウサイズが性能のボトルネックとなっている可能性を調査するために、伝送遅延が 20ms と 50ms において、TCP ウィンドウサイズを 2 倍にして、それぞれシミュレーションを行った。その結果を図 6 と図 7 に示す。

これらの測定結果より、ウィンドウサイズを 2 倍に

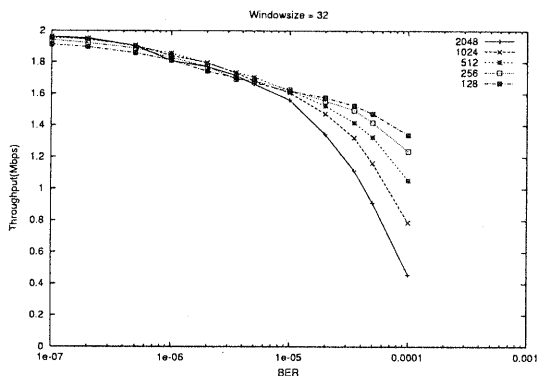


図 6 誤り率に対する TCP スループット (伝送遅延 20ms, ウィンドウサイズ 32)

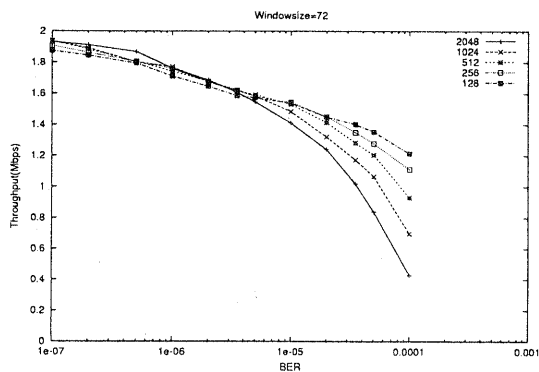


図 7 誤り率に対する TCP スループット (伝送遅延 50ms, ウィンドウサイズ 72)

した方が若干スループットがよいという結果を得た。ビット誤りが $1.0E-04$ の場合では、どのセグメントサイズにおいても、ウィンドウサイズが小さい方より平均値で約 1.3 倍のスループットを得ることができた。

4.2 シーケンス番号の推移

前節の結果においては、ビット誤り率が $1.0E-04$ の場合では、セグメントサイズによりスループットに差が出た。そこで、ビット誤り率が $1.0E-04$ の時に、TCP と SSCOP において送出されているパケットのシーケンス番号が、時間に対してどのように変化しているかを各セグメントサイズ毎に通信開始から 5 秒間測定した。伝送遅延 50ms の場合について、TCP と SSCOP の送出シーケンス番号の時間変化を、それぞ

れ図 8 と図 9 に示す。

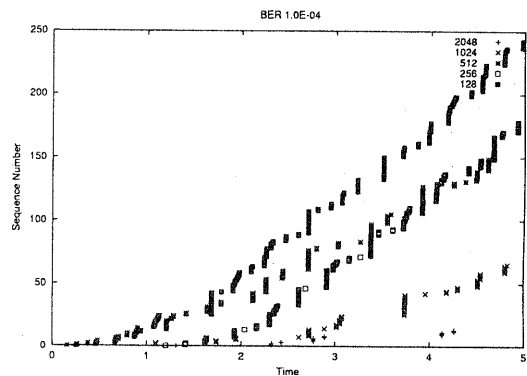


図 8 TCP の送出シーケンス番号の時間変化 (ビット誤り率 $1.0E-04$, 伝送遅延 50ms)

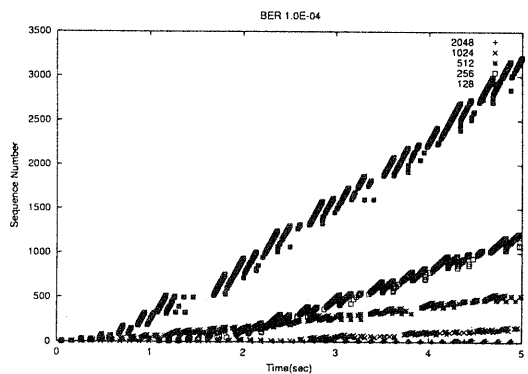


図 9 SSCOP の送出シーケンス番号の時間変化 (ビット誤り率 $1.0E-04$, 伝送遅延 50ms)

TCP におけるシーケンス番号の変化より、セグメントサイズが 128 バイトより増加するにつれて、TCP において送出されるパケット数が減少していることがわかる。この理由は、SSCOP において、セグメントサイズが増加するにつれて、再送されるパケットの数が増加し、SSCOP におけるデータの送出が押さえられているためである。特に、セグメントサイズが 2048 バイトの場合は、頻繁にパケットの再送が行われ、その結果 TCP レベルでのパケットの送信がほとんど行われていない。その結果、この図の範囲ではコネクショ

ン確立直後のスロースタートが完了していないと考えられる。

4.3 再送パケット数と全パケットによる回線利用率

次に、伝送遅延が 20ms と 50ms において、TCP と SSCOP において、各セグメントサイズ毎に、ビット誤り率に対する再送パケット数を測定した。さらに、TCP のみを用いた通信においても、再送パケット数を測定した。その結果 SSCOP を用いる場合は、TCP においては再送はほとんど起こらなかった。伝送遅延が 20ms と 50ms の場合の、SSCOP における再送パケット数および TCP のみを用いた場合の再送パケット数を、それぞれ図 10 と図 11 に示す。

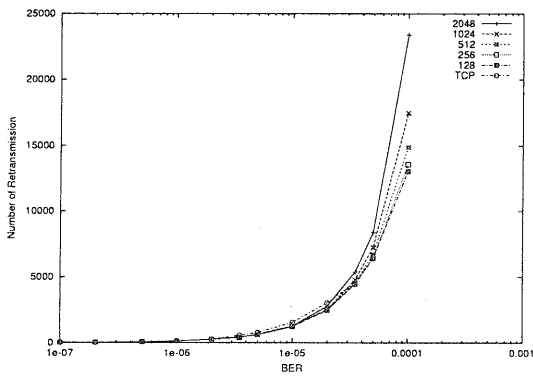


図 10 ビット誤りに対する再送パケット数 (伝送遅延 20ms, ウィンドウサイズ 16)

これらの結果から、SSCOP における再送パケット数は、ビット誤り率が $1.0E-05$ 以上では、急速に増加することが分かる。そして、セグメントサイズ毎に再送パケット数を見ると、ビット誤り率が $1.0E-04$ においては、セグメントサイズが 128 バイトの場合より 2048 バイトの場合の方が約 2 倍の再送パケット数を送出している。TCP のみを使用した結果では、通信が可能であったビット誤り率が $2.0E-05$ 以下の範囲において、SSCOP における再送パケット数とほぼ同一のパケット数が再送された。また、伝送遅延 20ms と 50ms においては、再送パケット数に大きな差は見られなかった。

再送パケット数を測定した結果、SSCOP では、ビット誤り率が大きくなるに従って、再送パケット数も増加することが分かった。よって、SSCOP 間の回線では、送出すべきパケットと再送を要求されたパケットの両方を送出する。しかし、通常のスループットの測定では、回線がどれぐらいの割合で利用されているかが不明である。そのため、再送を含めた全ての送信パ

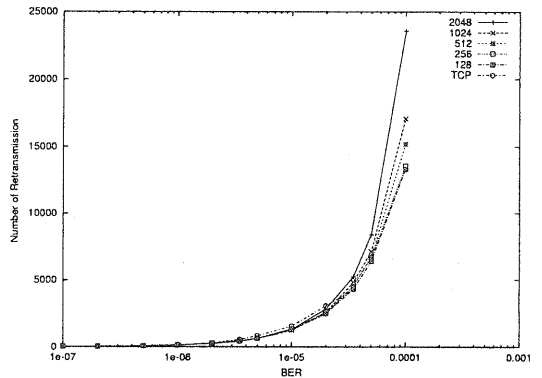


図 11 ビット誤りに対する再送パケット数 (伝送遅延 50ms, ウィンドウサイズ 32)

ケット数と、全パケットを送信するのに要する時間から、再送を含めた全ての送信パケットによる回線の利用率を求めた。伝送遅延 20ms と 50ms において、各セグメント毎の回線利用率を、それぞれ図 12 と図 13 に示す。

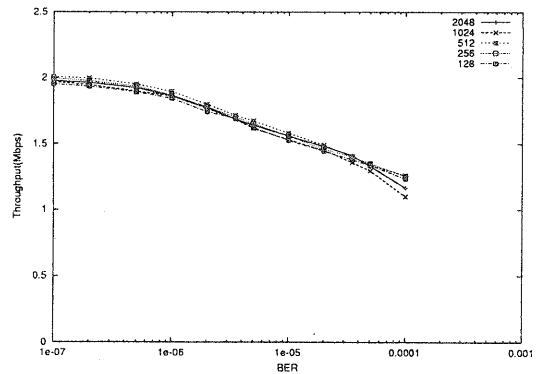


図 12 全パケットによる回線の利用率 (伝送遅延 20ms, ウィンドウサイズ 16)

これらの結果から、いずれのセグメントサイズに対しても、ほぼ同様な回線利用率であることが分かる。また、ビット誤り率が増加するに従って、回線の利用率が徐々に減少しており、ビット誤り率が $1.0E-04$ の場合では、伝送容量の半分まで減少している。さらに、回線の利用率についても伝送遅延が 20ms と 50ms では、あまり差が見られなかった。

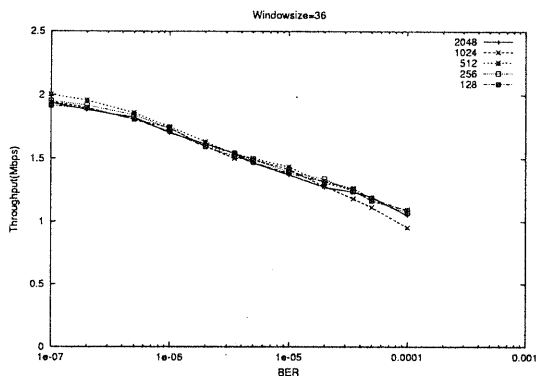


図 13 全パケットによる回線の利用率 (伝送遅延 50ms, ウィンドウサイズ 36)

5. 考 察

- (1) 無線区間のデータリンクプロトコルに SSCOP を採用したことによって、高いビット誤りが発生しても、TCP レベルにおけるスループットの低下を押さえることができた。これは、SSCOP で多くの再送パケットが送出しているにもかかわらず、TCP において再送パケットがほとんどない結果から、SSCOP の選択再送機能による誤り回復が有効に行われ、TCP の輻輳回避が行われなかったためである。
- (2) セグメンティング機能については、ビット誤り率が高くなるに従って、セグメントサイズの小さい方が高いスループットを得ることができた。この理由としては、ビット誤りが頻繁に起こる場合、小さいセグメントサイズでは、紛失したパケットだけを再送することが可能であるが、大きいセグメントサイズでは、そのパケット自身を再送しなければならないためであると考えられる。
- (3) シミュレーションで測定された再送パケット数を検証する為に、再送パケット数を求める式を導出する。ビット誤り率を e 、全体の送ったデータサイズ N ビット、1パケットのサイズを n ビットとする。

送ったパケット数は

$$N/n$$

となり、 n ビット送信して誤らない率は

$$(1-e)^n$$

となる。すなわち、少なくとも 1 ビットでも誤る率は

$1 - (1 - e)^n$
である。従って、1 回目の再送数は

$$\{1 - (1 - e)^n\} \times \frac{N}{n}$$

である。また 2 回目の再送数は

$$\{1 - (1 - e)^n\}^2 \times \frac{N}{n}$$

となる。再送数の総和を求めると、

$$\frac{\{1 - (1 - e)^n\} \times \frac{N}{n}}{1 - \{1 - (1 - e)^n\}}$$

$$= \frac{N}{n(1 - e)^n}$$

と導くことができる。ここで、 N は $10000 \times 1500 \times 8$ ビットで、 n には各セグメントサイズ $\times 8$ ビットを代入する。この式によって算出した再送パケット数の結果を図 14 に示す。

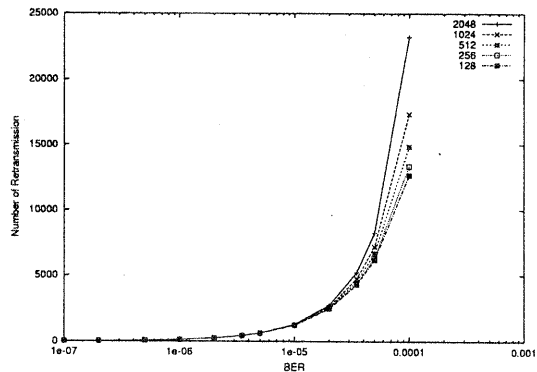


図 14 理論式による SSCOP の再送数

この理論式によって求めた再送パケット数の結果は、シミュレーションにより測定された SSCOP における再送パケット数にほとんど一致した。

6. おわりに

本稿では、無線区間のデータリンクプロトコルとして、実際に ATM 網で使用されており、効率の良い選択再送機能を持つ SSCOP を採用し、さらに、SSCOP にパケットのセグメンティングとリアセンブリング機能を付加して無線区間のパケットサイズを減少させるアプローチを提案した。さらに、この方式を用いて、無線区間で TCP 通信を行った場合の性能を、シミュレーションを用いて評価し、その結果について述べた。評価結果から、セグメンティング機能を付加した SSCOP

の有効性が確認され、さらに、セグメントサイズを小さくした方が TCP レベルにおける性能が向上することが検証された。具体的には、TCP のパケットサイズが 1500 バイト、セグメントサイズが 128 バイトの場合、TCP のみを使用するとまったく通信が不可能であったビット誤り率 $1.0E-04$ において、本アプローチでは 2Mbps 回線上で約 1.2Mbps のスループットを得ることができた。

参 考 文 献

- 1) 井戸上, J. Morten, 田上, 加藤, “伝送誤りを伴う回線上での TCP/IP 通信の為にデータリンクプロトコルに関する一検討” 情処全国大会, 3G-03, March 1998.
- 2) H. Balakrishnan et al., “A Comparison of Mechanisms for Improving TCP Performance over Wireless Links,” Proc. ACM SIGCOMM'96, Aug. 1996.
- 3) PHS インターネット・アクセス・フォーラム, “PHS Internet Access Forum Standard(PIAFS)仕様書,” 第 1 版, March 1997
- 4) 長谷川, 長谷川, 加藤, 鈴木, “ATM 用確認型データ転送プロトコル SSCOP の実装と評価” 情処マルチメディア通信と分散処理研究会, 71-13, July 1995.
- 5) S. keshav, “Real : A Network Simulator,” Computer Science Department Technical Report 88/472, UC Berkeley, Dec. 1998.
振舞いに関する検討” 情処マルチメディア通信と分散処理研究会, 84-2,