1U－5

長距離超高速インターネット (4)
－ ボトルネック －

村上 健一郎† 釘本 健司‡ 天海 良治† 岡 敦子‡

NTT 基礎研究所† NTT ソフトウェア研究所‡

1 はじめに

我々は、156Mbps の同期デジタルハイアラーキ SDH (Synchronous Digital Hierarchy) を使用した長距離超高速インターネットの実験を進めている [1]。そこで使用しているプロトコルは、TCP/IP (Transmission Control Protocol / Internet Protocol) である。本論文では、実験結果 [2][3][4] や理論上の検討をもとにしてボトルネックを整理し、実験で得られた知見について議論する。

2 さまざまなボトルネック

2.1 TCP/IP プロトコルの理論限界

TCP/IP のヘッダには、プロトコルの限界となり得る２つの領域、即ち、IP のデータグラム識別番号 IP-ID と TCP の順序番号 (Sequence Number) とがある。これらはそれぞれ 16bit と 32bit 長である。前者は、IP データグラムを送り出す際につけられるユニークな番号である。そして、それが途中のルータで細分化された場合に、転送先のノードが同じ IP ID をもつデータグラムを集めて再組み立てできるようになっている。このため、データグラムがネットワーク内に生存する時間 TTL (Time-To-Live) 内には同じ番号を持つデータグラムを転送できない。TTL は TCP で指定される 8bit の値である。そして、IP データグラムの最大長は 64Kbyte なので、その転送速度の上限は、64(Kbyte) * 64(K datagrams) / TTL(sec) で与えらえる。つまり、TTL が 15 秒の場合には 2.2Gbps、また、TTL が 255 秒の場合には 129Mbps が理論上の限界となる。

一方、順序番号は 4Gbyte まで表現できるが、これも転送先のノードで重複した値を持つ TCP セグメントが受信されないようにしなければならない。従って、転送速度の上限は、4096(Mbyte) / TTL(sec) となり、TTL が 15 秒の場合には 2.2Gbps、TTL が 255 秒の場合には 129Mbps である。

なお、データグラムの TTL はルータに中継されるごとに 1 だけ減らされ、0 になった時点で破棄される。これは、ルータがデータグラムを中継するのに一秒を要すると想定しているからである。しかし、実際にはより短かい時間で中継されるので生存時間は TTL で指定された値よりも短かくなる。従って、限界スピードは計算値 (129Mbps から 2.2Gbps) よりは大きいものとなる。

Ken-ichiro Murakami, Takeshi Kugimoto,
Yoshiji Amagai, and Atsuko Oka
NTT Basic Research Laboratories, NTT Software Laboratories

2.2 プロトコル実装のボトルネック

2.2.1 ヘッダ処理の速度

Jacobson[5] は、TCP/IP プロトコル処理の最適化を行い、IP を 37 命令と 7 回のメモリアクセス、TCP を最大 60 命令と 22 回のメモリアクセスで処理できることを確認した。この結果から 1 パケットの処理 (1 パケットの受信とそれに対する確認応答 ACK の処理) に 400 命令の実行を仮定すると、10MIPS のプロセッサでは 40μsec で処理ができることになる。つまり、毎秒 25,000 パケットの処理ができる。ここでパケット長を 4Kbyte とすると、転送速度は 800Mbps になる。これは十分な処理速度が達成可能であることを示している。

2.2.2 メモリアクセスによるボトルネック

上記の計算ではメモリアクセスのオーバヘッドが考慮されていない。メモリアクセスは、インタフェースとシステム空間との間のコピー、チェックサム計算、システム空間からユーザ空間へのコピーで 4 回ものメモリサイクルが必要となる。しかも、オーバヘッドはパケット長に比例する。ここではヘッダ処理に伴うメモリアクセスを無視し、1byte 当りのアクセス時間を平均 50nsec、そして、4 回のメモリアクセスが必要な処理系を想定する。この場合、パケット長を 4Kbyte とすると、1 パケット当りの処理に 1.6msec 程度を要することになる。これは毎秒 600 パケット、つまり、40Mbps 程度の転送速度しか出ないことになり、ヘッダ処理よりもメモリアクセスがボトルネックになることを示している。なお、インタフェースのメモリをシステム空間に置いたり、コピーの代わりに仮想空間の切り替えを行うなどの最適化を行うことによって、メモリサイクルを減らすことも可能である。

2.3 アルゴリズムの不整合とボトルネック

長距離超高速インターネットの環境は、これまでのアルゴリズムの前提とは大きく異なっている。例えば、より大きな最大セグメント長 MTU (Maximum Segment Size) が使用されたり、伝送遅延と帯域の積 DBP (Delay Bandwidth Product)、つまり、ネットワーク内に蓄積されるパケットの量 [2] が膨大なためにさまざまなアルゴリズムとの不整合が発生する。

2.3.1 大きな MTU によるアルゴリズムの不整合

SWS(Silly Window Syndrome) とは、TCP の極端に小さいウィンドーサイズによってほとんどヘッダだけしか含まない細分化された TCP セグメントが発生する現象である。これによる有効帯域の減少やヘッ

ダ処理オーバヘッドの増大などの問題に対し、転送ノードと受信ノードの両方でSWSを回避する以下のNagleのアルゴリズムが使用される。BSD UNIXの実装によると、受信側では、ACKを行っていないデータがMTUの2倍を超えてしかも受信バッファが空になった場合、または、受信バッファサイズの35%が満たされた場合、または、ACKが500ms以上遅延した場合に初めてACKを行う。また、送信側ではMTU分のウィンドーが開くか、すべての転送済にデータのACKを受信した場合に限って送信する。

ところがTCPバッファよりも大きなMTUを持つ高速回線では、送信バッファと受信バッファの組み合わせによっては、意図しない定期的な待ち状態が発生する場合がある。即ち、送信側は受信側からACKが返ってくるのを待つ一方、受信側はACKの条件がそろうための更なるデータが送信側から送られてくるのを待ってACKを遅延する。そして、このデッドロックは受信側の遅延タイマ切れによって解けるが、これが極端なスループットの低下となって現われる。

2.3.2 帯域制御アルゴリズムの不整合

TCPでは、エンドノード間のフローコントロールにネットワーク側の輻輳制御を重畳している。まず、データを転送してからそれに対応するACKが受信されるまでの時間RTT(Roung Trip Time)を常に計測し、それを再転送のタイマへ反映させる。また、通常のウィンドーの他に輻輳ウィンドーと呼ばれるネットワークの負荷を反映させたウィンドーも持っている。そして、利用可能な帯域を探るために次第に輻輳ウィンドーを大きくしてゆき(Slow-start)、最初のパケットのロスが発生した時を輻輳の発生と考え、輻輳ウィンドーを絞る(Congestion Avoidance)。ネットワークでは、利用できる帯域が常に変動しているので、この動作をくり返すことになる[3]。

しかし、長距離超高速インターネットでは、パケットのロスを意図的に利用するような帯域の調整は大きなオーバヘッドを招く。それは、DBPが非常に大きいため、パケットのロス検出時にはすでに多くのパケットが先送りされており、TCPのGo-back-Nアルゴリズムではロスのあったパケット全部を再転送しなければならないからである。また、パケットロスが検出されて再転送が開始される間まで、ウィンドーが尽きて長い間転送が停止したままとなる。

3 得られたいくつかの知見

3.1 DBPを考慮した帯域制御方式

現在の帯域制御アルゴリズムの最も大きな問題は、静的に設定されたウィンドーサイズがDBPを超える場合、回線の帯域以上に輻輳ウィンドーを拡大しようとして無駄なSlow-startとCongestion Avoidanceが繰り返されることである。これによるパケットのロスは大きなパフォーマンスの低下をもたらす[2][3][4]。

この問題を改善するため、回線自体の帯域を見積もり、それ以上の輻輳ウィンドーの拡大を行わないようにする方法や拡大を試る頻度を調整する方法などが考えられる。ここでは、(1) スループットの履歴を保持して回線の帯域を見積もる方法 (2) ウィンドーベースではなく転送レートベースで制御する方法 (3) ウィンドーベースとレートベースのハイブリッド方式 (4) 状況に応じて動的に適切なアルゴリズムの切り替えを行う方法などが検討課題である。これらの方法には、計測が簡単で、しかも、正確な見積もりができることや従来との互換性が要求される。

3.2 同期制御とバックトラック

長距離のネットワークでは、伝送遅延によってネットワークをまたぐ同期に大きなコストを要する。すでに述べた帯域制御アルゴリズムの問題は、転送元ノードと転送先ノード間とのフローに関する同期の問題と考えることができる。しかし、この問題はアプリケーションを含めたあらゆるレイヤにもあてはまる。例えば、頻繁なリンクレイヤでのリンクの確立は遅延による待ち時間の増大を招き、スループット低下の要因となる。また、アプリケーションレイヤにおいても同様で、スループット低下を防止するためには、同期の頻度をできるだけ減らすような非決定的なパラダイムに基づくプログラミングが必要である。しかし、同期点の間隔が大きくなると、前の同期点までのバックトラック(トランスポートレイヤにおける再転送に相当する)が発生した場合に、そのオーバヘッドが大きくなるため、そのトレードオフを考慮する必要がある。

3.3 光速と高速通信

高速ネットワークでは、その速度が上れば上るほど、パケットの物理的なデータ長が短かくなる。そして、全部のデータを光ファイバに入れる時間とファイバ中をデータが伝播する時間が逆転する。そのため、いくら高速(広帯域)の伝送装置を持ってきても、遅延が光速にバウンドされ性能が向上しない。伝送装置のレベルで2.4Gbpsや10Gbpsを超える速度の伝送装置を使用できても、光速を超えるような伝送媒体を手に入れることは不可能であるので、エージェント同士が双方向の通信を行う高次の通信では、これまでとはまったく異なる伝送遅延に対応したプログラミングパラダイムが必要となる。

4 おわりに

本論文では、長距離超高速インターネットの実験や理論上の検討をもとにして、さまざまなボトルネックをいくつかのカテゴリに分類した。また、実験で得られたいくつかの知見について議論し、今後のエージェント間通信には非決定性を指向したプログラミングパラダイムが必要なことを明らかにした。

参考文献

[1] 岡、釘本、天海、村上: 長距離超高速インターネット(1) 実験概要について、第50回情報処理全国大会、1995

[2] 釘本、岡、村上、天海: 長距離超高速インターネット(2) スループット、第50回情報処理全国大会、1995

[3] 天海、村上、釘本、岡: 長距離超高速インターネット(3) 特性解析、第50回情報処理全国大会、1995

[4] 村上、天海、釘本、岡、伊藤正樹、後藤、伊藤光泰: 長距離超高速インターネット, *NTT R&D*, Vol. 43, No. 9, 1994. pp.973-pp.982

[5] V.Jacobson: Some Design Issues for High-speed Networks, Networkshop '93, Nov. 1993.