


P2P/ブロードバンド時代の 新・TCP/IP 入門

村上 健一郎 法政大学ビジネススクール イノベーション・マネジメント研究科 教授 

第6回 なぜ宣伝どおりのスピードが出ない? - その1 -

7月号では、複雑なIP/TCPヘッダーの中で最も重要なIPアドレスとTCPポートについて説明しました。また、IPアドレスをもとに、パケットの中継をするルーターがどのような処理を行っているのかについても明らかにしました。今回と次回は、TCPヘッダーが絡む通信速度の上限について2回に分けて考えてみます。

[Q1]

光ファイバーのインターネット接続サービスに加入したのに、なぜ、宣伝どおりのスピードが出ないのでしょうか？ 速度がベストエフォートなら、料金の支払いをベストエフォートにしたいのですが……。パソコンのパラメーター設定にも原因があると聞いたんですが……。

[A1]

「確認応答」は相手に届いた印

料金を払って入った高速道路だって、渋滞の時は歩いたほうが速いということがしばしばです。また、最高時速70kmのボンコツ車では、高速道路を時速100kmで走るなんて無理ですね。一方、最高時速320kmのフェラーリで豪速を楽しむと、後ろから赤色灯のついた車が追いかけてくること間違いありません。

車ではなく道路のほうが悪いことだってあります。普通のドライバーや車では、パリ・ダカールラリーのようなとんでもない路面には耐えられないでしょう。

このように考えてみると、速度が出ない原因はさまざまで、その原因も時刻や状況によって変化することがわかります。ネットワークの場合も同じです。ここでは、ネットワークのスピードが抑えられる原因をトランスポートレイヤーとインターネットレイヤーに着目して分析していきましょう。

まず、ネットワーク全体の概要を頭に描きましょう。図1(a)がそれです。ここでの構成要素はたった3つ、通信を行うペア(1対)のコンピュータと、その間でパケットを中継するインターネットです。この図を見ると単にコンピュータ同士が線でつながっているのしか思えません。ところが、TCPのパケットの流れを考えると、実は、パケットの往復(ラウンドトリップ)を考えなければならないのです。これが図1(b)です。

送信側のコンピュータ(図ではパソコン)からのデータの入ったパケットは、インターネットを通過して受信側のコンピュータ

(図ではサーバー)に到着するのですが、そのデータが到着したことを知らせる「確認応答」パケットが逆に送られてくるのです。確認応答があって初めて、データが相手に確実に受信されたことがわかります。もし、ある時間以内に確認応答のパケットが送信側のコンピュータに到着しなければ、送信側のコンピュータは、受信側のコンピュータに誤りが発生したものと考えて、パケットを再転送します。

データの転送の仕組みを見よう

実は、第5回で説明したTCPヘッダーの中で重要な3つの領域について、まだ説明していませんでした。それは、シーケンス番号(順序番号)、ACK(確認応答)番号、それにウィンドウサイズです。

まず、TCPでは、送信側は個々のデータの「バイトごと」に識別番号を付けます。これがシーケンス番号です。また、それを受信したコンピュータのTCP側では、受理したバイトの位置を送信側に示すため(つまり、何番目までのバイトを受信したか

を知らせるため、受信したデータの「最後のバイトに付けられたシーケンス番号 + 1」の番号、言い換えると、次に受信を期待しているシーケンス番号を、確認応答パケットの ACK 番号で伝えます。このとき、自分が現在受信できる余裕のあるバイト数をウィンドウサイズとして一緒に伝えます。これで、ACK 番号にウィンドウサイズを足した番号までのデータの送信を許可したことになります。

たとえば、図2の①では、シーケンス番号が α (seq = 0)、データ長が1024バイト (len = 1024) のパケットを送っています。このパケットのデータ部にある最後のバイトのシーケンス番号は1023ということになります。①に対し、受信側では、次に受信を期待する番号1024 (= 1023 + 1) を ACK 番号とした確認応答パケット②を送っています。このウィンドウサイズは2048バイト (win = 2048) です。受信側では、これだけの長さのデータ (2048バイト) を受信する余裕があるということです。逆に、この長さまでは受信側の確認応答 (ACK なしに、送信側はデータパケットを送ることができます。

送信側では、②のACK番号に対応したシーケンス番号以降のデータを、パケット③で1024バイト分送っています。しかし、③に対する受信側からの確認応答を待つ必要はありません(前述したように、受信側は2048バイトを受信する余裕を持っている)。それを待つことなく、次のシーケンス番号2048 (= 1024 + 1024) から1024バイトのデータをパケット④で送っています。このように、確認応答なしに次のパケットも送ることができるので、転送速度を向上させることができます。受信側では、③と④に対する確認応答としてそれぞれ確認応答⑤と⑥を送り返しています。

ここで、⑤が転送中にエラーが発生してパケットの破棄が起こり、送信側で受信できなかったと仮定しましょう。送信側では、④に対する確認応答⑥は受信で

きましたが、②で許可されたウィンドウサイズ(2048バイト)を③で使い切ってしまったので、③に対応する確認応答⑤が受信されない限りこれ以上のデータは

送信できません。TCPでは、このようなデータや確認応答パケットが破棄された場合に備えて、データパケットの送信時にタイマーを起動しておきます。ある時間以

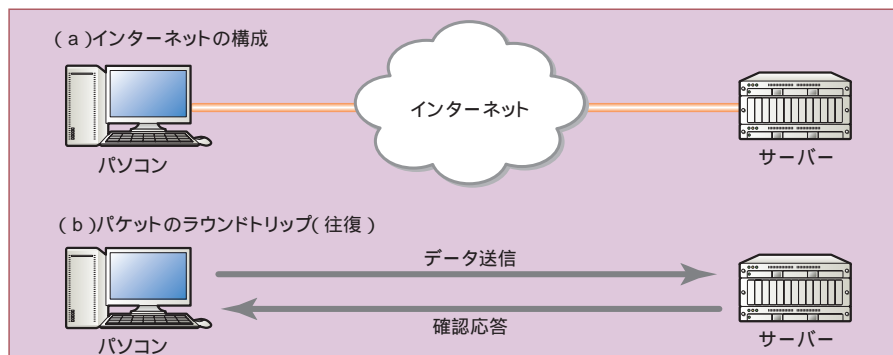


図1 インターネットの構成とTCPの確認応答

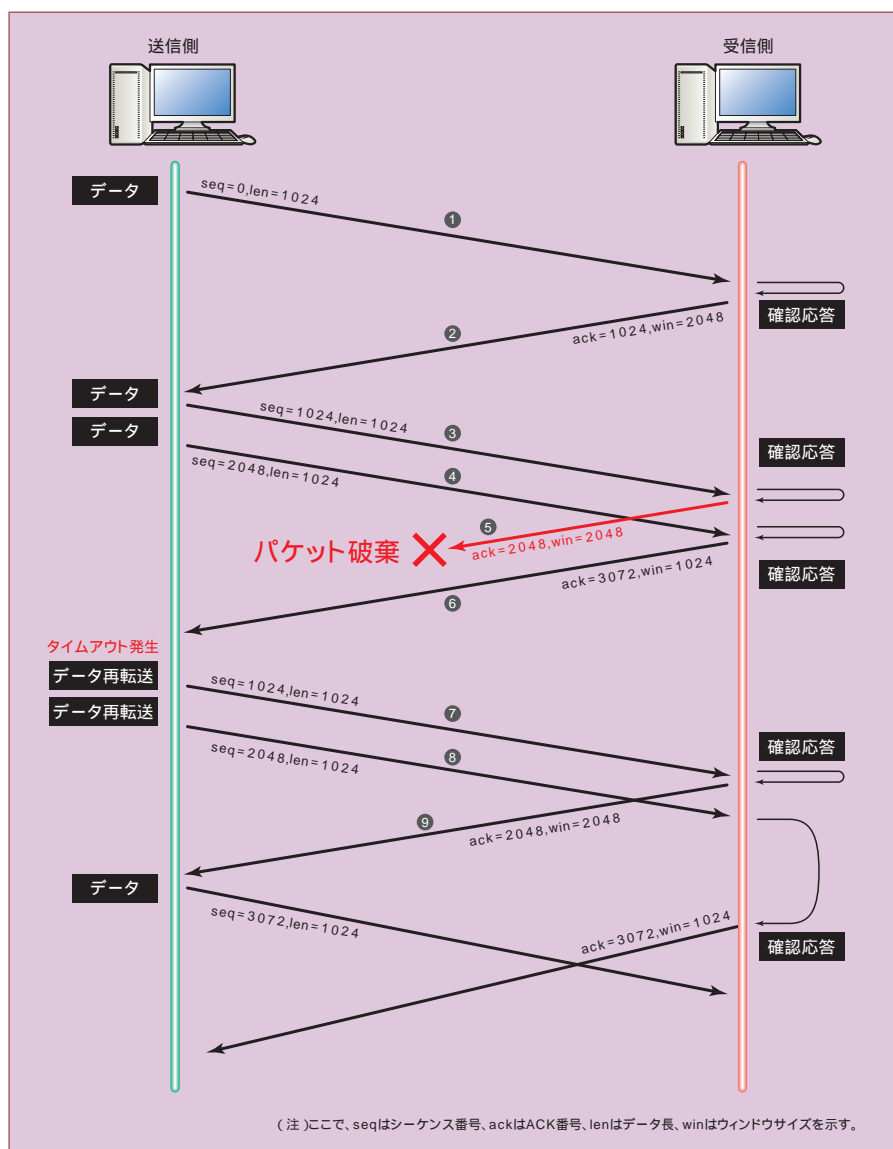


図2 TCPデータの転送の様子

内にそれに対応する確認応答が受信できなかった場合、データもしくは確認応答パケットが破棄されたものと考え、再送信します。これが、図2の⑦と⑧です。TCPでは、確認応答がなかったシーケンス番号(この場合、seq = 1024)以降を全部送り直すことにもなっているので、すでに確認された④についても⑥で再送信されています。

以上のように、TCPの転送は、確認応答を受信しなくても送信側でデータを送ることのできるウィンドウサイズの通知という仕組みを持つため、パケットごとに確認応答を待つ必要がなく、転送効率を向上させることができます。このウィンドウサイズが大きいほど転送効率がよくなります。しかし、この効果が出ない状況もいくつかあります。たとえば、転送遅延が大きすぎて、次のウィンドウサイズを知らせる確認応答パケットがなかなか到着しないときです。そうであれば、非常に大きいウィンドウサイズを知らせておけば済みそうですが、それがかえって逆効果を生む場合もあります。

ウィンドウサイズが大きいと、あまりに多くのパケットが一度に送られてしまい、輻輳(混雑)が発生してルーターでパケットが捨てられることがあるのです。つまり、一概には適切な値がわからないというこ

とです。適切な値というもの、通信する相手がどのくらい離れているか(これはラウンドトリップ時間に影響する)にもよりますし、ネットワークがどのくらい混雑しているかによっても違います。

最大ウィンドウサイズの悩み

悩みの一例がWindows 2000のウィンドウサイズでした。このSP1(Service Pack 1)とSP2では最大ウィンドウサイズは17520バイト(1460バイト長のデータ12個分)でした。Service PackとはWindowsオペレーティングシステムの更新のことで、機能追加、機能の変更、バグの修正などがなされたものです。それぞれの更新を識別するために、リリースされるごとにSP1、2、3……のように識別番号が更新されていきます。現在の最新の更新は、SP4になっています。

SP3とSP4では、転送速度を向上させるため、たとえば、100Mbpsのネットワークでは64Kバイトに設定されるようになりました。しかし、バースト的に(瞬時に大量の)パケットが送られた場合、ネットワークの一時的な混雑によって、パケットが捨てられ、かえって再転送が発生して転送速度が低下することがあったのです。

そこで特定の脆弱性に対応するパッチ

(修正プログラム)の中で、SP3以降でのウィンドウサイズの64Kバイトを元の17520バイトに戻したところ、SP3以降で向上していた転送速度が元に戻って1/4ほどに低下する場合があります。転送速度T(bps)は、ウィンドウサイズ(byte)に8をかけてビット単位に直したものを遅延(秒)で割った値なので、同じ遅延でもウィンドウサイズが小さくなると転送速度が低下してしまいます(図3)。

このように、万能の値を決めるというのは難しいということです。そこで、ネットワークの状況を判断して自動的にウィンドウサイズの最大値を制御するための輻輳ウィンドウというものも利用されています。

輻輳ウィンドウの巧妙な仕組み

実は、送信側では、受信側から送られてきたウィンドウサイズだけでなく、もう一つ、輻輳ウィンドウというものを使用してネットワークの混雑を回避しています。

確認応答で通知されたウィンドウサイズと、この輻輳ウィンドウとを比較し、どちらか小さいほうをウィンドウサイズとして採用しています。この輻輳ウィンドウは、実に巧妙に決定されています。まず、最初は送るパケットを1パケット長(最大のTCPセグメント長)分だけにし、相手から確認応答が返るごとに、その2倍に拡大していきます。もし、相手からのウィンドウサイズが小さければ、そこでウィンドウサイズの拡大は止まります。これは、ネットワークの余裕のほうか、受信側コンピュータの受信バッファの余裕よりも大きいことを意味します。ですから、単に受信側からのウィンドウサイズを考慮すればよいことになります。

それでは、ネットワークが混雑していた場合はどうなるのでしょうか？輻輳ウィンドウのサイズがだんだん大きくなっていくと、受信側からの確認応答なしに送信側が送ることのできるパケットの数も大きくなっていきます。そして、ついにネット

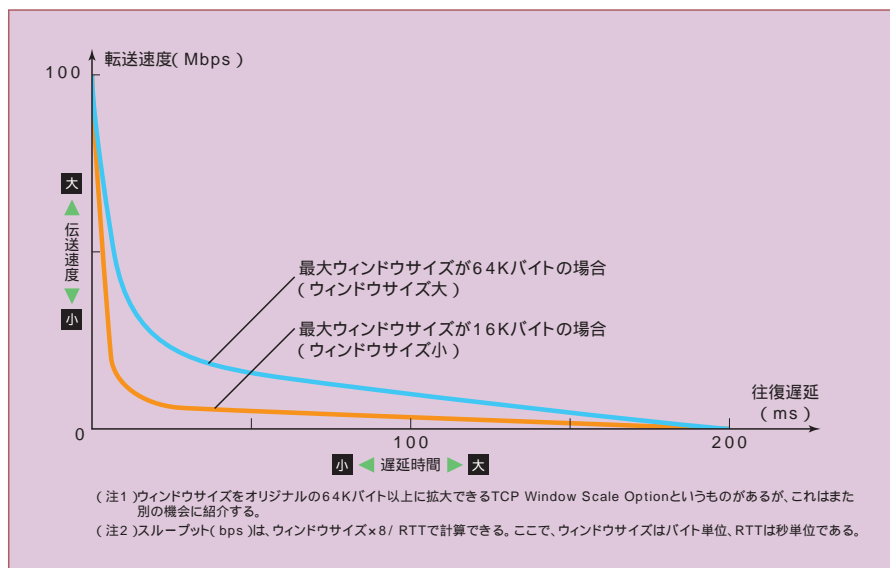


図3 遅延(往復の時間)と転送速度との関係

ワークの限界に近づいて輻輳が始まると、ルーターはパケットを捨てます。このことは、相手からの確認応答パケットが返ってこないことで、送信側も気づきます。このとき、送信側では、ネットワークが混雑していて、その能力以上にパケットを送り込んでしまったと考え、輻輳ウィンドウサイズを小さくします。しかし、いつしかネットワークに余裕ができて、より多くのパケットが送れるようになるかもしれません。そこで、再度、次第に輻輳ウィンドウサイズを大きくするを行います。この様子を図4に示します。

要するに、ちょっと頭を出してみてもルーターに叩かれたら頭を引っ込めるといった動作を繰り返すのです。こうして、何も測定手段がないにもかかわらず、TCPはネットワークの状態を継続的に観測します。そして、システム全体が最適な利用率になるように制御し、しかも、それによって、自分も最適な速度で通信ができるように努力しているのです。

ルーターのもぐら叩き

すでに述べたパケットの破棄は、ルーターからしてみればもぐら叩きのようなものです。このもぐら叩きにも工夫があるのです。

ルーターは、受信バッファがパケットであふれそうになると、その中からパケットを捨てます。簡単な方法は、あるパケットがルーターで受信されてルーターのバッファがある値を超えたとき、そのパケットを捨てるということです。この方法では、一番後に入ってきたパケットを捨てることになります。これをTD(Tail Drop)と呼びます。しかし、これは平等にネットワークを使用するという点では問題があります。なぜならば、あるコンピュータからのパケットが順番を待って行列して(これを「待ち行列」という)たくさん並んでいても、それが捨てられずに、とにかく最後に受信されたパケットが捨てられている

からです。これでは、平等になりません。

ところが、世の中には、頭のいい人がいます。待ち行列にあるパケットにランダムにピストルを発射して当たったパケットを捨てようというのです。これをRED(Random Early Drop)と呼びます。同じコンピュータからのパケットがたくさん待ち行列に入っている場合には、その数に比例して玉の当たる確率が高くなります。逆にパケットが少ない場合、玉の当たる確率は低くなります。ですから、公正さが増すというわけです(図5)。つまり、これまで行儀よくネットワークを使用していた人には、転送速度の向上という利益をもたらします。もっとも、確率的に決めるわけですから、流れ玉に当たって損をするパケットもあるわけですが.....。

ルーター以外にも、このような動作をするものがあります。それは、「帯域制御装置(伝送速度の制御装置)」というもので

す。これは、サーバーに特定のユーザーから一度にウェブの閲覧要求が押し寄せないように、サーバーの前などに接続しておくものです。ユーザー側の回線がいくら速くならうとも、これで帯域制限(速度制限)されていたのでは、それ以上の転送速度の向上は見込めません。このような装置には、帯域制限の方法にいくつかのバリエーションがあります。1つは、確認応答パケットに含まれるウィンドウサイズを小さく変更してしまうものです。もう1つは、ルーターのもぐら叩きと同じように、パケットを捨てることで転送元からのパケットにブレーキをかけるものです。

*

さて、次回(第7回)は、今回説明したウィンドウサイズと同様に、ネットワークレイヤーで効率を左右する「最大IPパケット長」について考えてみます。

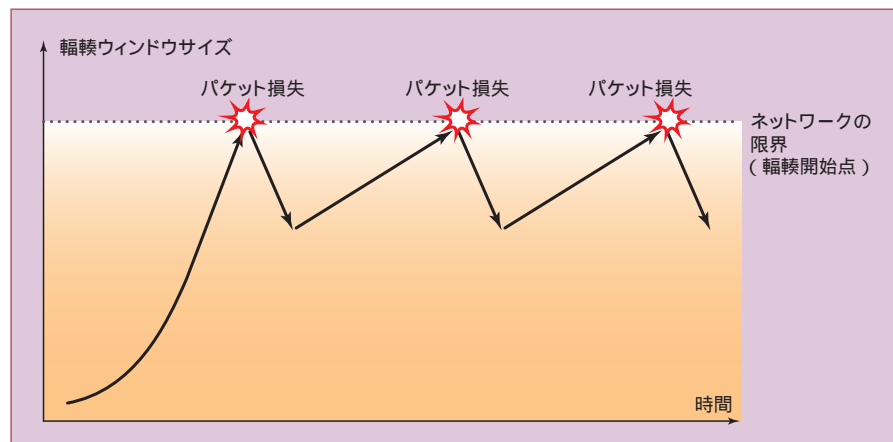


図4 輻輳ウィンドウの時間経過

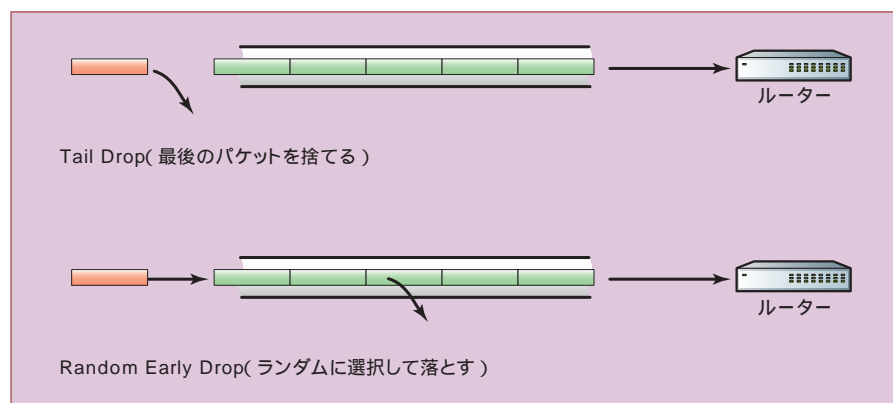


図5 TD(Tail Drop)とRED(Random Early Drop)



[インターネットマガジン バックナンバーアーカイブ] ご利用上の注意

このPDFファイルは、株式会社インプレスR&D(株式会社インプレスから分割)が1994年～2006年まで発行した月刊誌『インターネットマガジン』の誌面をPDF化し、「インターネットマガジン バックナンバーアーカイブ」として以下のウェブサイト「All-in-One INTERNET magazine 2.0」で公開しているものです。

<http://i.impressRD.jp/bn>

このファイルをご利用いただくにあたり、下記の注意事項を必ずお読みください。

- 記載されている内容(技術解説、URL、団体・企業名、商品名、価格、プレゼント募集、アンケートなど)は発行当時のものです。
- 収録されている内容は著作権法上の保護を受けています。著作権はそれぞれの記事の著作者(執筆者、写真の撮影者、イラストの作成者、編集部など)が保持しています。
- 著作者から許諾が得られなかった著作物は収録されていない場合があります。
- このファイルやその内容を改変したり、商用を目的として再利用することはできません。あくまで個人や企業の非商用利用での閲覧、複製、送信に限られます。
- 収録されている内容を何らかの媒体に引用としてご利用する際は、出典として媒体名および月号、該当ページ番号、発行元(株式会社インプレス R&D)、コピーライトなどの情報をご明記ください。
- オリジナルの雑誌の発行時点では、株式会社インプレス R&D(当時は株式会社インプレス)と著作権者は内容が正確なものであるように最大限に努めましたが、すべての情報が完全に正確であることは保証できません。このファイルの内容に起因する直接のおよび間接的な損害に対して、一切の責任を負いません。お客様個人の責任においてご利用ください。

このファイルに関するお問い合わせ先

株式会社インプレスR&D

All-in-One INTERNET magazine 編集部

im-info@impress.co.jp