

# 特別研究報告

題目

TCP/IP ネットワークにおける  
制御理論を用いた RED ルータの特性解析

指導教官

村田 正幸 教授

報告者

米良 祐一郎

平成 12 年 2 月 23 日

大阪大学 基礎工学部 情報科学科

TCP/IP ネットワークにおける制御理論を用いた RED ルータの特性解析

米良 祐一郎

内容梗概

パケット交換ネットワークにおいてデータ転送系サービスを効率的に収容するためには、フィードバック型輻輳制御が不可欠である。現在のインターネットではフィードバック型の輻輳制御機構として、ウィンドウ型のフロー制御方式である TCP (Transmission Control Protocol) が用いられている。しかし、TCP の輻輳制御機構は、もともとルータのアルゴリズムを考慮せずに動作するように設計されているため、ネットワーク全体の性能が制限されてしまうという問題がある。そこで近年、エンド-エンド間で動作する TCP の輻輳制御機構を補助するために、ルータにおける輻輳制御機構 (パケットスケジューリング機構) がいくつか提案されている。この中で、現在もっとも有望と考えられ、実際に実装されつつあるのは、ルータにおいて意図的にパケット棄却を発生させる RED (Random Early Detection) ルータである。しかし、これまで RED ルータの特性は十分には明らかにされていない。例えば、RED ルータの有効性は制御パラメータの設定に大きく依存するが、制御パラメータをどのように決定すればよいかはこれまでまったく明らかにされていない。そこで本報告では、まず、TCP によってフロー制御されたトラヒックに対する RED ルータの特性を数学的に解析し、この解析結果を用いて、さまざまなネットワーク環境下において、RED ルータの制御パラメータをどのように決定すればよいかを明らかにしていく。その結果、エンド-エンド間の伝搬遅延時間に影響を与えるパケット待ち行列長の平衡点は、RED ルータの最大パケット棄却確率に大きく依存することを示す。また、TCP のコネクション数もしくは (RED ルータの処理能力  $\times$  TCP の往復伝搬遅延時間) が大きくなるにつれ、RED ルータの制御パラメータの安定領域が大きくなることを示す。さらに、ネットワーク全体の過渡特性を最適化するには、RED ルータの最小のしきい値を慎重に決定する必要があることを示す。以上の解析結果にもとづき、RED ルータの制御パラメータの決定方法を提案する。

## 主な用語

インターネット、TCP (Transmission Control Protocol)、RED (Random Early Detection)  
ルータ、制御理論、定常状態解析

# 目次

1	はじめに	5
2	解析モデル	8
3	RED ルータの定常状態解析	12
3.1	状態遷移方程式の導出	12
3.2	平均状態遷移方程式の導出	14
3.3	平衡点の期待値の導出	16
4	RED ルータの安定性解析	18
5	数値例による考察	20
5.1	平衡点の期待値に関する考察	20
5.2	近似解析の妥当性の検証	20
5.3	安定条件に関する考察	22
6	RED ルータの制御パラメータ決定方法の提案	29
7	まとめ	31

## 目次

1	解析モデル	8
2	パケット棄却確率 $p_b$ と平均パケット待ち行列長 $\bar{q}$ の関係	10
3	典型的なウィンドウサイズの変化の様子	15
4	平衡点におけるウィンドウサイズの期待値と TCP のコネクション数の関係 ( $B = 2$ [パケット/ms], $\tau = 1$ [ms])	21
5	平衡点における現在パケット待ち行列長の期待値と TCP のコネクション数 の関係 ( $B = 2$ [パケット/ms], $\tau = 1$ [ms])	21
6	シミュレーション結果との比較 ( $N = 1$ ) ( $B = 2$ [パケット/ms], $\tau = 1$ [ms], $N = 1$ )	23
7	シミュレーション結果との比較 ( $N = 5$ ) ( $B = 2$ [パケット/ms], $\tau = 1$ [ms], $N = 5$ )	24
8	最大パケット棄却確率を変化させた時の安定領域 ( $B = 2$ [パケット/ms], $N = 1$ , $\tau = 1$ [ms])	25
9	最大パケット棄却確率を変化させた時の安定領域 ( $B = 2$ [パケット/ms], $N = 5$ , $\tau = 1$ [ms])	25
10	最大パケット棄却確率を変化させた時の安定領域 ( $B = 10$ [パケット/ms], $N = 1$ , $\tau = 1$ [ms])	26
11	最大パケット棄却確率を変化させた時の安定領域 ( $B = 2$ [パケット/ms], $N = 1$ , $\tau = 5$ [ms])	27
12	制御パラメータと $ s_i $ の最大値との関係 ( $B = 2$ [パケット/ms], $N = 1$ , $\tau = 1$ [ms], $max_p = 0.1$ )	28
13	制御パラメータと $ s_i $ の最大値との関係 ( $B = 2$ [パケット/ms], $N = 5$ , $\tau = 1$ [ms], $max_p = 0.1$ )	28

## 1 はじめに

パケット交換ネットワークにおいて、データ転送系サービスを効率的に収容するためには、フィードバック型輻輳制御が不可欠である。フィードバック型輻輳制御の目的は、ネットワーク内部でのパケット棄却を防ぐとともに、ネットワーク資源の有効利用を図ることにある。現在のインターネットではフィードバック型の輻輳制御機構として、ウィンドウ型フロー制御方式である TCP (Transmission Control Protocol) が用いられている。例えば、現在広く利用されている TCP Reno (TCP バージョン Reno) では、ネットワークからのフィードバック情報として、ネットワーク内部でのパケット棄却を利用する [1, 2]。これは、パケット棄却の発生が、ネットワークが輻輳状況にあることを意味すると考えられるためである。TCP Reno の基本的な動作は以下の通りである。TCP Reno は、まずパケット棄却が発生するまで、送信側ホストのウィンドウサイズを連続的に増加させる。送信側ホストのウィンドウサイズが利用可能帯域を超えると、余分なパケットはネットワーク内のルータが持つバッファに蓄えられることになる。送信側ホストがさらにウィンドウサイズを増加させると、ルータにおいてバッファあふれが発生し、その結果パケット棄却が発生する。送信側ホストは、同じシーケンス番号を持つ ACK パケットを複数受信することなどにより、これらのパケット棄却を検出し、ウィンドウサイズを減少させる。ウィンドウサイズが減少されると、ネットワーク内の輻輳は解消される。この結果、パケット棄却が発生しなくなるため、送信側ホストは再びウィンドウサイズを増加させる。このように TCP の輻輳制御機構は、ウィンドウサイズを少しずつ増加させてゆき、パケット棄却が発生した時点でウィンドウサイズを減少させる。TCP の輻輳制御機構はこのような処理を繰り返す。

TCP の輻輳制御機構は、もともとルータのアルゴリズムを考慮せずに動作するように設計されている。つまり、TCP の輻輳制御機構は、ルータがどのように動作するかについては何も仮定していない。これは、実際のネットワークにおいては、ルータにおけるパケット処理順序やパケット棄却アルゴリズムは、送信側ホストには分からないからである。現在のインターネットのように、ネットワーク内に複数の輻輳制御機構や、異なる動作アルゴリズムを持つルータが存在する場合は、TCP の輻輳制御機構がルータのアルゴリズムから独立していることは望ましい。しかし、TCP の輻輳制御機構にこのような一般性を持たせてい

るために、ネットワーク全体の性能が制限されてしまうという問題もある。

エンド-エンド間で動作する TCP の輻輳制御機構を補助するための、ルータにおける輻輳制御機構がいくつか提案されている [3-5]。この中で、現在もっとも有望と考えられているのは、ルータにおいて意図的にパケット棄却を発生させる RED (Random Early Detection) ルータである [4]。RED ルータの基本的なアイデアは、ルータに到着するパケットを確率的に棄却することにより、平均パケット待ち行列長 (ルータのバッファに蓄えられているパケット数の平均値) を低く抑えるというものである。つまり、RED ルータでは、平均パケット待ち行列長にほぼ比例した確率で、到着するパケットを確率的に棄却する。RED ルータの動作アルゴリズムは非常に単純であるため、現実のルータにも容易に実装することが可能である。文献 [4] において著者らは、バッファが一杯になって始めてパケットを棄却する Tail-Drop ルータと比較して、RED は以下のような利点を持つと主張している。(1) RED ルータは平均パケット待ち行列長を低く抑えることが可能であるため、その結果、エンド-エンド間の転送遅延も小さく抑えられる。(2) RED ルータでは、Tail-Drop ルータのように、バースト・トラフィックに対して偏りのあるパケット棄却 (バースト的に到着するパケットをより多く廃棄してしまう) が発生しない。(3) RED ルータでは、Tail-Drop ルータで発生するような、TCP コネクションが同期してしまうという問題を防ぐことができる。しかし、これまで RED ルータの特性は十分には明らかにされていない。

RED ルータの特性をシミュレーション実験により評価した研究はいくつかなされている [4, 6, 7]。しかし、RED ルータに関して明らかにされていない問題も多く残されている。その中で最も困難な問題は、RED ルータの制御パラメータをどのように決定すればよいかという問題である。RED ルータの有効性は、制御パラメータの設定に大きく依存することがわかっている [6, 8, 9] が、制御パラメータをどのように決定すればよいかはまったく明らかにされていない。例えば、文献 [4] において著者らは、制御パラメータの推奨設定値を提案しているが、これは経験的に得られたガイドラインであり、理論的な根拠はまったくない。実際、RED ルータの特性を理論的に解析した研究は非常に少ない。文献 [10] では、バースト的なトラフィックおよびバースト的でないトラフィックに対する RED ルータの性能解析を行っている。しかし、ここで用いられている解析モデルは非常に単純なものに限定されてい

る。例えば、RED ルータへの入力トラヒックが集団・ポアソン到着過程でモデル化されており、TCP の輻輳制御機構の動的なふるまいをまったく考慮していない。また、文献 [11] では、RED ルータの動的なふるまいを解析しているが、RED ルータへの入力トラヒックは再生過程もしくは MMPP (Markkov Modukated Poisson Process) のどちらかに限定されている。RED ルータは、もともと TCP の輻輳制御機構と協調して動作するように設計されているので、RED ルータの特性を正しく把握するためには、TCP の輻輳制御機構のふるまいを考慮に入れる必要がある。

本報告の目的は、TCP によってフロー制御されたトラヒックに対する、RED ルータの特性を数学的に解析することである。そのため TCP の輻輳制御機構の動作をモデル化することにより、定常状態における RED ルータの特性を解析する。また、いくつかの数値例を用いて、RED ルータの制御パラメータが、ネットワーク全体の性能にどのような影響を与えるかを明らかにする。また解析結果を利用することにより、さまざまなネットワーク環境下における RED ルータの制御パラメータを決定方法を提案する。

本報告の構成は以下のとおりである。まず、2 章では本解析で用いるネットワークモデルを定義し、あわせて RED ルータの動作アルゴリズムを簡単に説明する。3 章では RED ルータに TCP トラヒックが加わった場合の、RED ルータの定常特性を解析する。続いて 4 章では、現代制御理論を適用することにより、定常状態においてネットワークが安定するためのパラメータ条件を導出する。5 章では、制御パラメータが RED ルータの性能にどのような影響を与えるかを数値例により考察する。また、解析結果とシミュレーション結果を比較することにより、本解析の妥当性を検証する。6 章では、本報告で得られた解析結果にもとづき、RED ルータの制御パラメータをどのように決定すればよいかを提案する。最後に 7 章では、本報告のまとめと今後の課題について述べる。



## 2 解析モデル

本報告で用いる解析モデルを図 1 に示す。解析モデルは、単一の RED ルータと  $N$  本の TCP コネクションによって構成されている。ここでは、TCP コネクションの往復伝搬遅延時間はすべて等しいとし、 $\tau$  [ms] と表記する。またルータの処理能力を  $B$  [パケット/ms] と表記する。ここでは図に示すルータがネットワーク全体のボトルネックであると仮定する。つまり、すべてのホスト-ルータ間、ルータ-ルータ間の伝送速度はルータの処理速度よりも速いものとする。

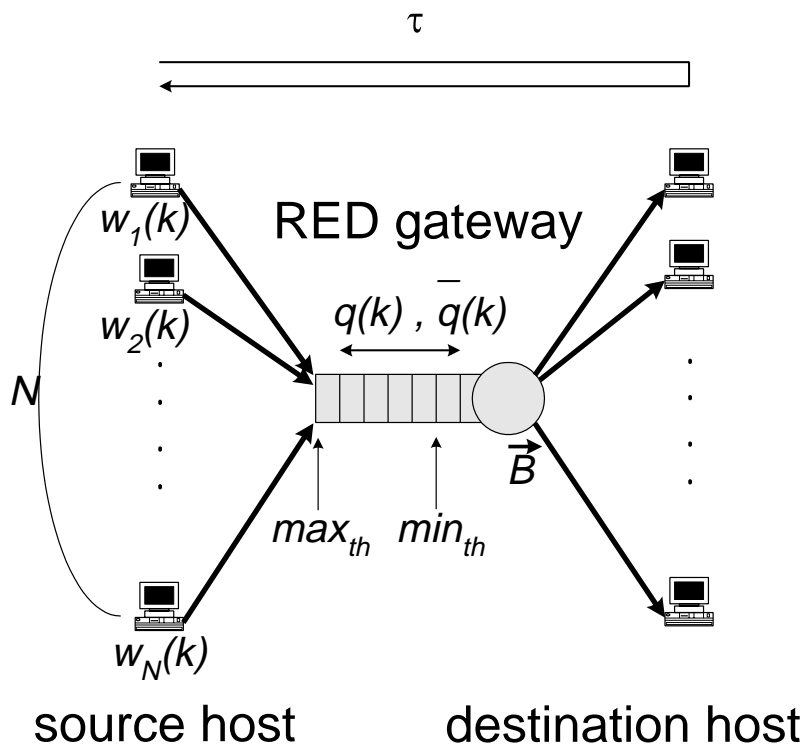


図 1: 解析モデル

すべての送信側ホストでは、TCP Reno 版 [2, 12] の輻輳制御機構をモデル化する。各送信側ホストにはウィンドウサイズが定められており、その値は TCP の輻輳制御機構によって制御される。送信側ホストのウィンドウサイズを  $w$  とすれば、この送信側ホストは、パケットを送出してから対応する ACK (ACKnowledgement) パケットを受信するまでの間に

$w$  個のパケットを送出することが許される。つまり、送信側ホストは 1 ラウンド・トリップ時間中に、 $w$  個のパケットをバースト的に送送することが可能である。そこで本解析では、ネットワーク全体を、単位スロット時間が TCP コネクションのラウンド・トリップ時間であるような離散時間システムとしてモデル化する。なお、 $n$  ( $1 \leq n \leq N$ ) 番目の送信側ホストの、 $k$  番目のスロットにおけるウィンドウサイズを  $w_n(k)$  [パケット] と定義する。また、すべての送信側ホストは常に伝送するデータをもっていると仮定する。つまり、すべての送信側ホストは、各スロットにおいて必ず  $w_n(k)$  個のパケットを送出するものとする。

RED ルータにはいくつかの制御パラメータがある。まず、 $min_{th}$  と  $max_{th}$  はそれぞれ最小および最大のしきい値である。これらのしきい値は、RED ルータに到着するパケットに対する棄却確率を決定するために用いられる。また、RED ルータは平均パケット待ち行列長を保持している。これは、ローパスフィルターの一種である、指数平均 (EWMA; Exponential Weighted Moving Average) を用いて現在のパケット待ち行列長から計算される。つまり、現在パケット待ち行列長と平均パケット待ち行列長をそれぞれ  $q$  および  $\bar{q}$  とすれば、RED ルータはパケットが到着するたびに、平均パケット待ち行列長  $\bar{q}$  を以下の式で更新する。

$$\bar{q} \leftarrow (1 - w_q)\bar{q} + w_q q \quad (1)$$

ここで、 $w_q$  は平均パケット待ち行列長に対する現在パケット待ち行列長の重みを決定する制御パラメータである。本解析では、 $k$  番目のスロットにおける現在パケット待ち行列長と平均パケット待ち行列長を、それぞれ  $q(k)$  [パケット] および  $\bar{q}(k)$  [パケット] と定義する。なお本解析では、 $q$  および  $\bar{q}$  は 1 スロット内では一定であると仮定する。文献 [10] に示されているように、 $w_q$  の値が小さい場合にはこの仮定は妥当なものである。

このようにして得られた平均パケット待ち行列長を用いて、RED ルータは到着するパケットに対する棄却確率  $p_b$  を計算する。具体的には、RED ルータはパケットの棄却確率  $p_b$  を以下のように決定する。

$$p_b = \begin{cases} 0, & \text{if } \bar{q} < min_{th} \\ 1, & \text{if } \bar{q} \geq max_{th} \\ max_p \left( \frac{\bar{q} - min_{th}}{max_{th} - min_{th}} \right), & \text{otherwise} \end{cases} \quad (2)$$

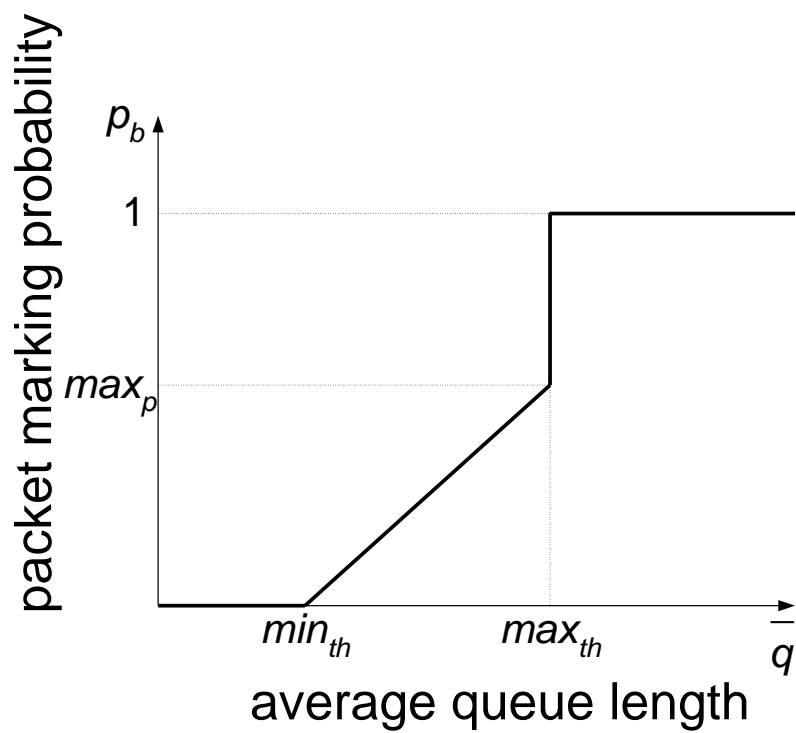


図 2: パケット棄却確率  $p_b$  と平均パケット待ち行列長  $\bar{q}$  の関係

ここで、 $max_p$  は最大の packets 棄却確率を決定する制御パラメータ (図 2) である。なお、RED ルータは TCP のコネクションを区別しないため、同じ packets 棄却確率  $p_b$  がすべての packets に対して用いられる。参考のため、表 1 に文献 [4] の著者らが推奨している RED ルータの制御パラメータの値を示す。

表 1: 推奨されている RED の制御パラメータの値

$min_{th}$	最小しきい値	5	[パケット]
$max_{th}$	最大しきい値	15	[パケット]
$max_p$	最大 packets 棄却確率	0.1	
$w_q$	指数平均の重み	0.002	

### 3 RED ルータの定常状態解析

#### 3.1 状態遷移方程式の導出

まず、各スロットにおけるパケット棄却確率を導出し、あわせてパケット棄却確率の持つ意味を説明する。 $k$  番目のスロットの先頭で、 $n$  番目の送信側ホストはネットワークに  $w_n(k)$  個のパケットを送出する。RED ルータは送信側ホストから到着するパケットを確率的に棄却する。2 章で説明したように、本解析では 1 スロット内の平均パケット待ち行列長  $\bar{q}$  は一定と仮定している。この時、パケット棄却確率  $p_b$  も一定となるため、 $k$  番目のスロットにおけるパケット棄却確率を  $p_b(k)$  とする。式 (2) より、 $p_b(k)$  は以下の式で与えられる。

$$p_b(k) = \max_p \left( \frac{\bar{q}(k) - \min_{th}}{\max_{th} - \min_{th}} \right) \quad (3)$$

実際には、RED ルータは到着するパケットを以下の式で与えられる確率で廃棄する。

$$\frac{p_b(k)}{1 - count \cdot p_b(k)} \quad (4)$$

ここで、 $count$  は最後にパケット棄却が発生してから、現在までに (棄却されずに) 通過したパケット数である。到着するパケットをこのような確率で棄却するため、棄却されずに連続して通過するパケット数 (パケットが棄却されてから、次にパケットが棄却するまでに通過するパケット数) は  $\{1, 2, \dots, 1/p_b(k)\}$  の値を取る確率変数となる。

$$P_k[X = n] = \begin{cases} p_b(k), & 1 \leq n \leq 1/p_b(k) \\ 0, & \text{otherwise} \end{cases} \quad (5)$$

$k$  番目のスロットにおける、棄却されずに連続して通過するパケット数の期待値を  $\bar{X}(k)$  とすると、 $\bar{X}(k)$  は以下の式で与えられる。

$$\begin{aligned} \bar{X}(k) &= \sum_{n=1}^{\infty} n P_k[X = n] \\ &= \frac{1/p_b(k) + 1}{2} \end{aligned} \quad (6)$$

次に、図 1 に示す解析モデルの状態遷移方程式を導出する。以降の定常状態解析では、この状態遷移方程式が解析の基礎となる。 $n$  番目の送信側ホストから送出されたパケットが、

RED ルータでまったく棄却されない場合、これらのパケットに対応する ACK パケットが、1 ラウンドトリップ時間後に送信側ホストに返される。この時、TCP の輻輳制御機構は送信側ホストのウィンドウサイズを 1 だけ増加させる。一方、RED ルータでパケットが 1 つ以上棄却された場合、TCP の輻輳制御機構は送信側ホストのウィンドウサイズを  $1/2$  に減少する。なお、ここではすべてのパケット棄却は重複する ACK パケットにより検出できると仮定している [12]。

$w_n(k)$  個のパケットから、少なくとも 1 つのパケットが棄却される確率は以下の式で与えられる。

$$\frac{w(k)}{1/p_b(k)} = w(k) p_b(k) \quad (7)$$

ゆえに、 $k+1$  番目のスロットにおけるウィンドウサイズは以下の式で決定される。

$$w(k+1) = \begin{cases} \frac{w(k)}{2}, & \text{with probability } w(k) p_b(k) \\ w(k) + 1, & \text{otherwise} \end{cases} \quad (8)$$

TCP はウィンドウ型のフロー制御方式であるため、 $k$  番目のスロットで送出されるパケットは、すべて  $k+1$  番目のスロットが始まるまでに転送が完了していると仮定する。その結果、 $k+1$  番目のスロットにおける現在パケット待ち行列長は以下の式で与えられる。

$$q(k+1) = \sum_{n=1}^N w_n(k) - B \left( \tau + \frac{q(k)}{B} \right) \quad (9)$$

ここで、右辺の第 1 項は  $k$  番目のスロットで到着するパケットの総数であり、第 2 項は  $k$  番目のスロットでルータから出力されるパケットの総数である。なお、1 スロットの長さは、往復の伝搬遅延時間  $\tau$  とバッファにおける遅延時間  $q(k)/B$  の和で与えられている。

次に、 $k+1$  番目のスロットにおける平均パケット待ち行列長  $\bar{q}(k+1)$  を導出する。2 章で説明したように、RED ルータはパケットが到着するごとに、式 (1) を用いて平均パケット待ち行列長の値を更新する。今、平均パケット待ち行列長は 1 スロット内で一定であると仮定しているため、 $\bar{q}(k)$  は以下の式で得られる。

$$\bar{q}(k+1) = (1 - w_q) \sum_{n=1}^N w_n(k) \bar{q}(k) + \frac{w_q \{1 - (1 - w_q) \sum_{n=1}^N w_n(k)\}}{1 - (1 - w_q)} q(k) \quad (10)$$

図 1 に示した解析モデルは、ここまでに得られた状態遷移方程式 (式 (8)–(10)) によって記述される。また、システムの状態ベクトル  $\mathbf{x}(k)$  は以下のように与えられる。

$$\mathbf{x}(k) = \begin{bmatrix} w_1(k) \\ \vdots \\ w_N(k) \\ q(k) \\ \bar{q}(k) \end{bmatrix} \quad (11)$$

### 3.2 平均状態遷移方程式の導出

RED ルータは到着するパケットを確率的に棄却する。このため式 (8) の状態遷移方程式には確率的なふるまいが含まれている。以降では、定常状態におけるネットワーク全体のふるまいを解析するため、TCP コネクションと RED ルータの平均的な動作を記述する「平均状態遷移方程式」を導入する。

まず、RED ルータが一つもパケットを棄却しないスロットの列によって構成される「シーケンス」という概念を導入する。一つのシーケンスは、パケットが棄却された次のスロットから、次にパケット棄却が発生するまでのスロットによって構成される。単一のシーケンス内での、典型的なウィンドウサイズの変化の様子を図 3 に示す。この図では、 $k-1$  番目のスロットでパケット棄却が発生している。 $k$  番目のスロット以降、ウィンドウサイズ  $w_n(k)$  は式 (8) に従って変化する。つまり、送信側ホストから到着するパケットがひとつも棄却されなければ、ウィンドウサイズは次のスロットで 1 だけ増加する。もし一つ以上のパケットが棄却された場合、ウィンドウサイズは  $1/2$  に減少する。つまり、ウィンドウサイズが増加し始めてから、パケット棄却によりウィンドウサイズが  $1/2$  に減少するまでを一つのシーケンスとしている。

定常状態解析では、複数のスロットで構成されるシーケンスを一つの大きなスロットとみなして、ネットワーク全体を離散時間システムとして考える。ここで、 $\bar{s}$  をシーケンスを構

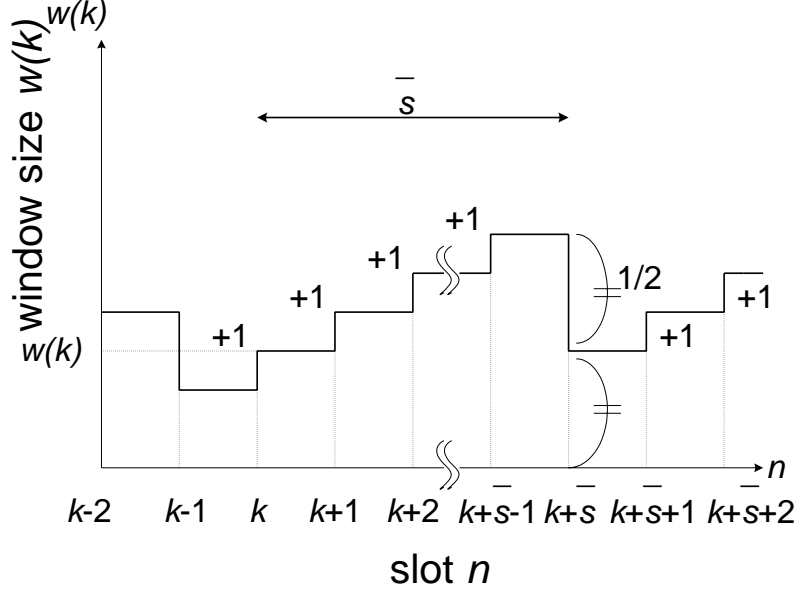


図 3: 典型的なウィンドウサイズの変化の様子

成するスロット数の期待値とする。この時、 $\bar{s}$  は次式を満足する。

$$\bar{X}(k) = \sum_{i=0}^{\lceil \bar{s} \rceil - 1} \sum_{n=1}^N w_n(k+i) \quad (12)$$

紙面の都合上、以下では送信側ホストのウィンドウサイズがすべて同期している場合のみを示す。ただし、以下に示す定常状態解析は、送信側ホストのウィンドウサイズが同期していない場合にも拡張することは容易である。今、 $k$  番目のスロットにおいて、すべての送信側ホストのウィンドウサイズを等しく  $w(k)$  と定義する。この場合、式 (12) は以下のようなになる。

$$\begin{aligned} \bar{X}(k) &= \sum_{i=0}^{\lceil \bar{s} \rceil - 1} N w(k+i) \\ &= N \left\{ \bar{s} w(k) + \frac{\bar{s}(\bar{s}-1)}{2} \right\} \end{aligned} \quad (13)$$

これを  $\bar{s}$  について解くことにより、以下の式が得られる。

$$\bar{s} = \frac{1}{2} \left\{ 1 - 2w(k) + \frac{\sqrt{N^2(1 - 2w(k))^2 + 8N\bar{X}(k)}}{N} \right\}. \quad (14)$$



2つの連続したシーケンス ( $k$  番目のスロットから  $k + \bar{s}$  番目のスロット) 間のウィンドウサイズ  $w(k)$  の平均状態遷移方程式は、式 (8) から以下のように与えられる。

$$\begin{aligned} w(k + \bar{s}) &= \frac{w(k + \bar{s} - 1)}{2} \\ &= \frac{w(k) + \bar{s} - 1}{2} \end{aligned} \quad (15)$$

同様に、シーケンス間の  $q(k)$  の平均状態遷移方程式は、式 (9) から以下のように与えられる。

$$\begin{aligned} q(k + \bar{s}) &= N w(k + \bar{s} - 1) - B \left( \tau + \frac{q(k + \bar{s} - 1)}{B} \right) \\ &\simeq \frac{N w(k + \bar{s} - 1) - B \tau}{2} \\ &= N w(k + \bar{s}) - \frac{B \tau}{2} \end{aligned} \quad (16)$$

ここで、 $q(k + \bar{s} - 1)$  を  $q(k + \bar{s})$  によって近似している。なお、 $q(k + \bar{s} - 1)$  と  $q(k + \bar{s})$  の差の上限値は、TCP のコネクション数  $N$  によって与えられる。

次に、シーケンス間の  $\bar{q}(k)$  の平均状態遷移方程式を導出する。式 (10) より、 $\bar{q}(k + \bar{s})$  は  $w(k + \bar{s} - 1)$  と  $q(k + \bar{s} - 1)$ 、 $\bar{q}(k + \bar{s} - 1)$  によって以下のように与えられる。

$$\begin{aligned} \bar{q}(k + \bar{s}) &= (1 - w_q)^{N w(k + \bar{s} - 1)} \bar{q}(k + \bar{s} - 1) \\ &\quad + \frac{w_q \{1 - (1 - w_q)^{N w(k + \bar{s} - 1)}\}}{1 - (1 - w_q)} q(k + \bar{s} - 1) \end{aligned} \quad (17)$$

各スロット間で現在パケット待ち行列長が大きく変化しないと仮定すれば、 $\bar{q}(k + \bar{s})$  は近似的に以下の式で与えられる。

$$\bar{q}(k + \bar{s}) \simeq (1 - w_q)^{\bar{X}(k)} \bar{q}(k) + \frac{w_q \{1 - (1 - w_q)^{\bar{X}(k)}\}}{1 - (1 - w_q)} q(k) \quad (18)$$

これらの平均状態遷移方程式 (式 (15)、(16)、(18)) は、それぞれウィンドウサイズ、現在パケット待ち行列長、平均パケット待ち行列長の平均的な変化をあらわしている。

### 3.3 平衡点の期待値の導出

1章で述べたように、TCP の輻輳制御機構では、送信側ホストのウィンドウサイズは絶えず振動するため、定常状態においても一定値に収束することはない。そこで以下では、定

常状態における TCP コネクションと RED ルータの動作を解析するために、「平衡点の期待値」を導出する。ここで  $w^*$ 、 $q^*$ 、 $\bar{q}^*$  をそれぞれウィンドウサイズ  $w(k)$ 、現在パケット待ち行列長  $q(k)$ 、平均パケット待ち行列長  $\bar{q}(k)$  の平衡点の期待値とする。平衡点の期待値は式 (15)、(16)、(18) において  $w(k) = w(k + \bar{s})$  などと置くことによって容易に求められる。

$$w^* = \sqrt{\frac{1}{4} + \frac{1}{3N} \left( \frac{max_{th} - min_{th}}{max_p(\bar{q}^* - min_{th})} + 1 \right)} - \frac{1}{2} \quad (19)$$

$$q^* = N w^* - \frac{B\tau}{2} \quad (20)$$

$$\bar{q}^* = q^* \quad (21)$$

なお、 $w^*$  は定常状態におけるウィンドウサイズの平均値を意味するのではなく、ウィンドウサイズの最小値の平均値を意味している。ウィンドウサイズの平均値は、 $w^*$  から以下のようにして求めることができる。

$$\lim_{k \rightarrow \infty} \frac{\sum_{i=0}^{\bar{s}} w(k+i) \left( \tau + \frac{q(k+i)}{B} \right)}{\sum_{i=0}^{\bar{s}} \left( \tau + \frac{q(k+i)}{B} \right)} \simeq \frac{3w^*}{2} - 1 \quad (22)$$

ここで、 $q(k+i)$  ( $0 \leq i \leq \bar{s}$ ) を  $q(k)$  で近似している。なお、式 (20) と式 (21) を解くことにより、 $w^*$  と  $q^*$  を求めることは可能であるが、冗長となるためここには含めない。

## 4 RED ルータの安定性解析

本章では、現代制御理論を適用することにより、定常状態における RED ルータの安定性について検討を行う。RED ルータへの入力トラヒックが TCP の輻輳制御機構によってフロー制御されている場合、送信側ホストのウィンドウサイズは定常状態においても安定せず、いつまでも振動する。さらに、RED ルータの動作は、平均パケット待ち行列長から決定される確率によって変化する。このため、そのままでは定常状態における RED ルータの安定性を議論するのは困難である。そこで本章では、3 章で得られた平均状態遷移方程式を用いて、定常状態における RED ルータの平均的な安定性（例えば、ウィンドウサイズの平均値が一定値に収束するかどうか）について検討を行う。なお、以下では 3 章の結果を用いて、送信側ホストのウィンドウサイズがすべて同期している場合のみを示す。ただし、以下に示す安定性解析は、送信側ホストのウィンドウサイズが同期していない場合にも適用可能である。

3 章において、図 2 に示す離散時間システムの平均状態遷移方程式は式 (15)、(16)、(18) によって与えられた。また、この離散時間システムの平衡点の期待値は式 (20)–(21) で与えられた。ここでは  $k$  番目のスロットにおけるシステムの状態ベクトルと、平衡点の期待値とのベクトルとの差  $\delta \mathbf{x}(k)$  を導入する。

$$\delta \mathbf{x}(k) \equiv \begin{bmatrix} w(k) - w^* \\ q(k) - q^* \\ \bar{q}(k) - \bar{q}^* \end{bmatrix} \quad (23)$$

$w(k)$ 、 $q(k)$ 、 $\bar{q}(k)$  を平衡点の近傍で一次近似することにより、 $\delta \mathbf{x}(k + \bar{s})$  は状態遷移行列  $\mathbf{A}$  と  $\delta \mathbf{x}(k)$  を用いて以下のように書くことができる。

$$\delta \mathbf{x}(k + \bar{s}) = \mathbf{A} \delta \mathbf{x}(k) \quad (24)$$

平均状態方程式 (式 (15)、(16)、(18)) で与えられる離散時間システムの平衡点の近傍における安定性は、状態遷移行列  $\mathbf{A}$  の特性方程式の解によって決定される [13]。つまり、システムの平衡点が局所漸近安定となるためには、

$$|s\mathbf{A} - \mathbf{I}| = 0 \quad (25)$$

によって得られる特性方程式の全ての解  $s_i (1 \leq i \leq 3)$  が、 $|s_i| < 1$  でなければならない (システムの安定条件)[14]。なお、状態遷移行列  $A$  がこの安定条件をみたすかどうかは、Jury の判別法などを用いて容易に調べることができる。

## 5 数値例による考察

### 5.1 平衡点の期待値に関する考察

3章で得られた平衡点の期待値 (式 (20)– (21)) は RED ルータの性質をいくつか示している。例えば、式 (20) や式 (21) は、TCP のコネクション数  $N$  もしくは最大パケット棄却確率  $max_p$  が増加するにつれ、定常状態におけるウィンドウサイズやパケット待ち行列長が減少することを示している。このような傾向は、図 4 および図 5 から知ることができる。これらの図では、TCP のコネクション数  $N$  を変化させた場合に、ウィンドウサイズおよび現在パケット待ち行列長の平衡点における期待値がどのように変化するかを示している。ここでは、ネットワークパラメータとして、RED ルータの処理能力  $B = 2$  [パケット /ms]、TCP コネクションの往復の伝搬遅延時間  $\tau = 1$  [ms] を用いている。なお、TCP のコネクション数  $N$  は 1 本から 20 本まで、最大パケット棄却確率  $max_p$  は 0.001 から 0.4 まで変化させている。RED ルータのその他の制御パラメータの値については、(表 1) に推奨されている値を使用している。これらの図より、平衡点におけるウィンドウサイズは TCP のコネクション数  $N$  に大きく依存していることがわかる。一方、平衡点におけるパケット待ち行列長は、TCP のコネクション数にはあまり依存しておらず、最大パケット棄却率  $max_p$  によってほぼ決定されていることがわかる。これは、RED ルータの平均パケット待ち行列長を制御するためには、最大パケット棄却率  $max_p$  を適切に設定する必要があることを意味している。

### 5.2 近似解析の妥当性の検証

本報告では解析においていくつかの仮定をおいている。そこで以下では、解析から得られる数値結果をシミュレーション結果と比較することにより、本解析の妥当性を検証する。シミュレーション実験では、ネットワーク・シミュレータ ns [15] を使用した。また、ネットワーク・モデルとしては図 1 と同じものを使用した。シミュレーションパラメータとして、RED ルータの処理速度  $B = 2$  [パケット/ms] (パケットサイズ 1000 [バイト] の時およそ 1.5 [Mbps/s] に相当する)、TCP のコネクションの往復伝搬遅延時間  $\tau = 1$  [ms] を用いた。

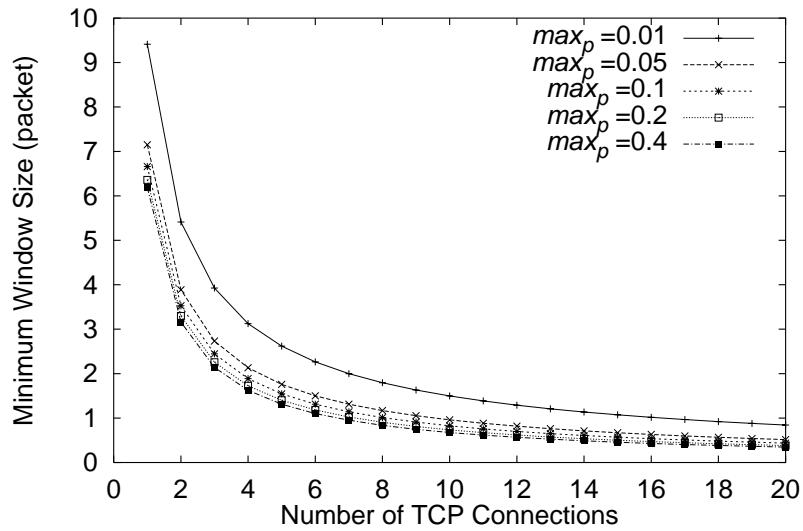


図 4: 平衡点におけるウィンドウサイズの期待値と TCP のコネクション数の関係 ( $B = 2$  [パケット/ms],  $\tau = 1$  [ms])

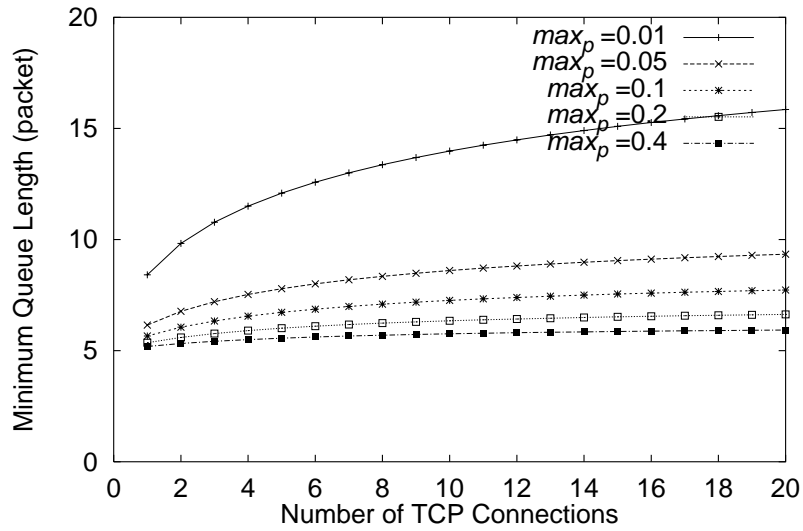


図 5: 平衡点における現在パケット待ち行列長の期待値と TCP のコネクション数の関係 ( $B = 2$  [パケット/ms],  $\tau = 1$  [ms])

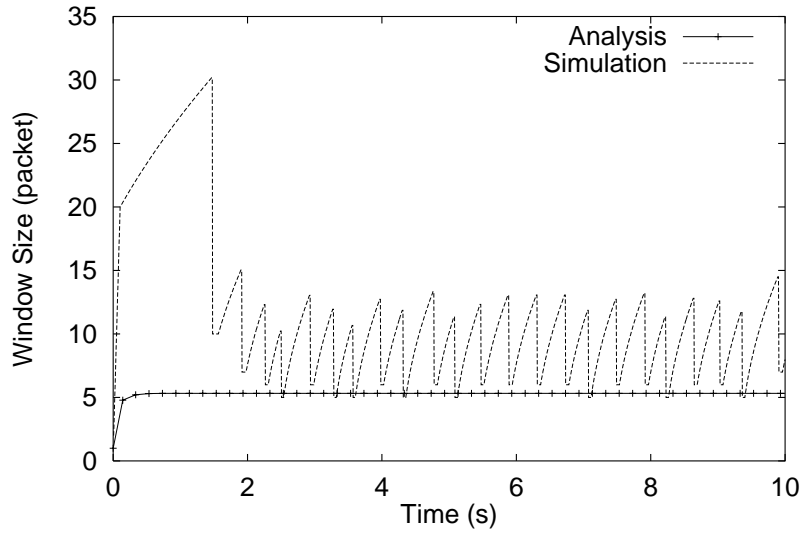
なお、TCP のコネクション数  $N$  を 1 または 5 とし、RED ルータの制御パラメータには表 1 の値を使用した。

図 6 および図 7 に、それぞれ  $N = 1$  または  $N = 5$  の場合の、シミュレーション実験により得られたウィンドウサイズとパケット待ち行列長の変動を示す。図中には、本解析で得られるウィンドウサイズおよびパケット待ち行列長の値  $w(k)$  および  $q(k)$  もあわせて示している。なおこれらの値は、本解析で得られた平均状態遷移方程式 (式 (15)、(16)、(18)) から数値計算により求めている。これらの図より、解析結果から得られたウィンドウサイズおよび現在キュー長の値が、シミュレーション結果と一致していることがわかる。つまり、解析から得られたウィンドウサイズ (平均のウィンドウサイズの最小値) が、シミュレーションから得られたウィンドウサイズの最小値とほぼ一致している。

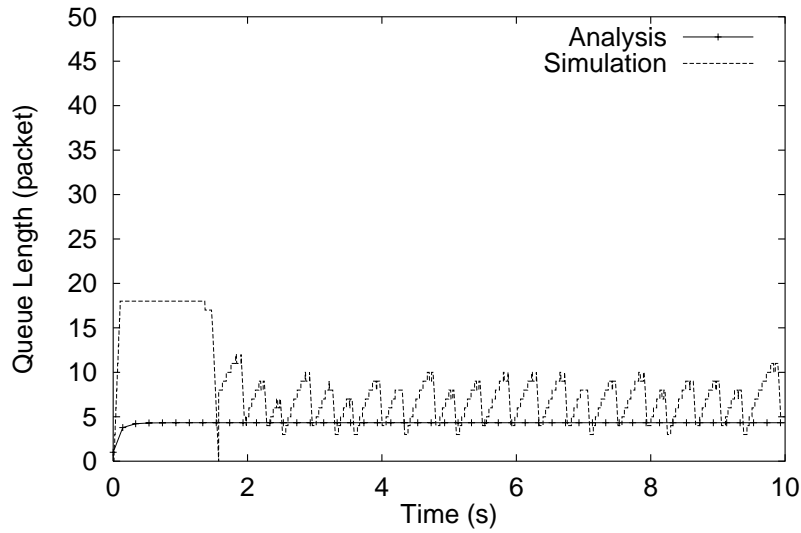
### 5.3 安定条件に関する考察

図 8 および図 9 にシステムの安定領域を示す。これらの図では、TCP のコネクション数をそれぞれ  $N = 1$ 、 $N = 5$  としている。グラフの X 軸は制御パラメータである最小のしきい値  $min_{th}$ 、Y 軸は制御パラメータである最大のしきい値  $max_{th}$  であり、最大パケット棄却確率  $max_p$  を変化させた時の、安定条件の境界線をプロットしている。これらの図では、 $(min_{th}, max_{th})$  の点が境界線の左側にある時、システムは安定となることを意味している。その他のパラメータに関しては、図 4 と同じ値 ( $B = 2$  [パケット / ms]、 $\tau = 1$  [ms]) を使用している。これらの図より、最大パケット棄却確率  $max_p$  が小さくなるにつれ、システムを安定化するパラメータ  $(min_{th}, max_{th})$  の領域が大きくなっていることがわかる。すなわち、最大パケット棄却確率  $max_p$  を小さく設定すれば、最小のしきい値  $min_{th}$  をある程度大きくしてもシステムを安定化できることがわかる。また、図 8 と図 9 を比較することにより、TCP のコネクション数  $N$  が増加すると、システムを安定化するパラメータ領域が大きくなっていることがわかる。

次に、RED ルータの処理能力  $B$  や TCP コネクションの往復の伝搬遅延時間  $\tau$  が増加した場合に、安定領域がどのように変化するかについて検討する。図 10 は、RED ルータの処理能力  $B = 10$  [パケット / ms] とした場合の安定領域を示している。その他のパラメー



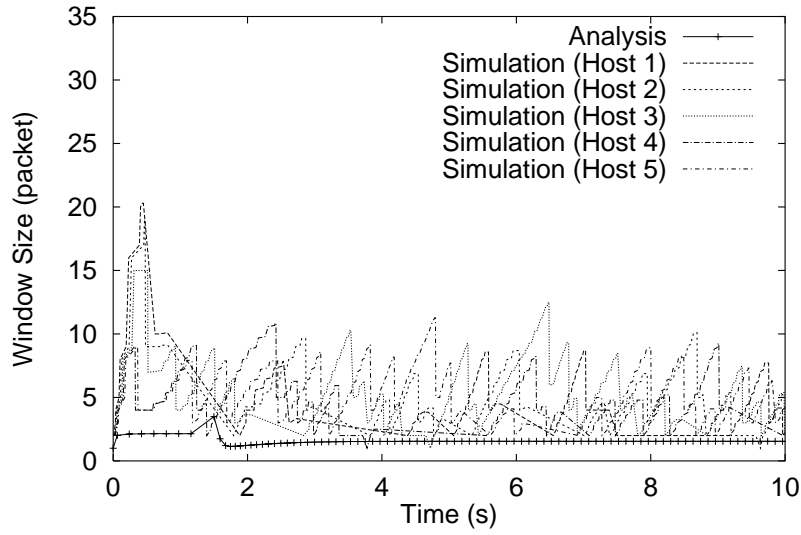
(a) ウィンドウサイズ



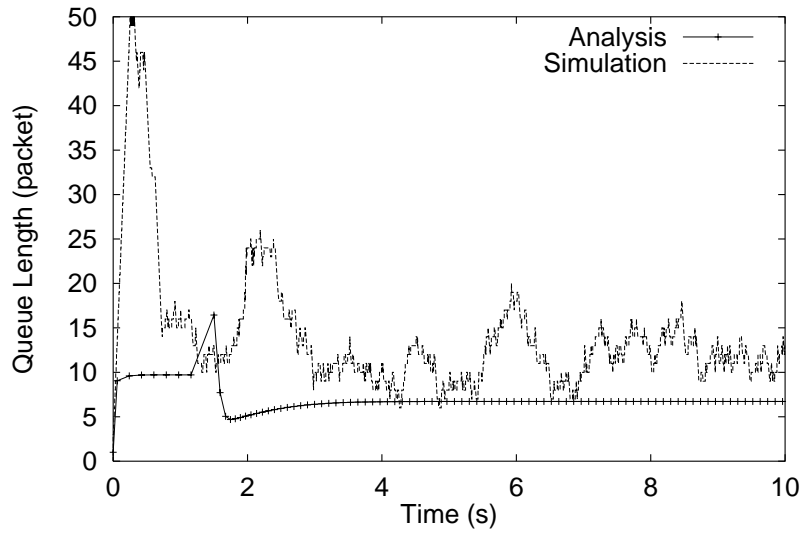
(b) パケット待ち行列長

図 6: シミュレーション結果との比較 ( $N = 1$ ) ( $B = 2$  [パケット/ms],  $\tau = 1$  [ms],  $N = 1$ )





(a) ウィンドウサイズ



(b) パケット待ち行列長

図 7: シミュレーション結果との比較 ( $N = 5$ ) ( $B = 2$  [パケット/ms],  $\tau = 1$  [ms],  $N = 5$ )

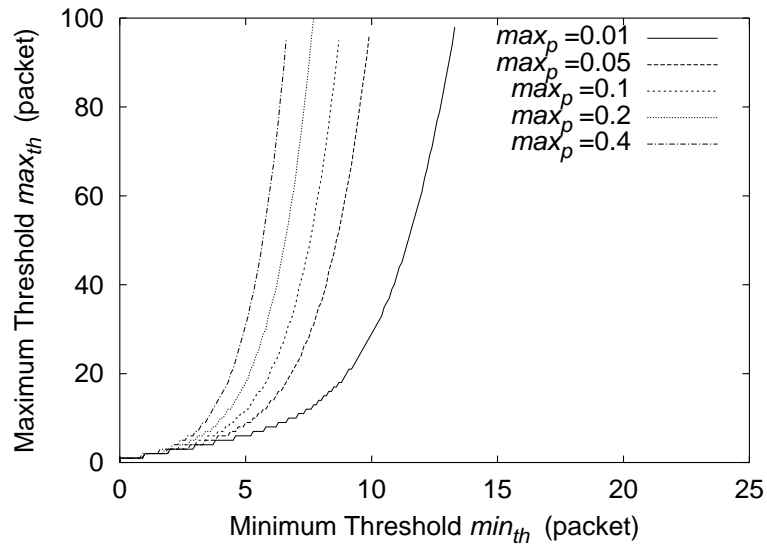


図 8: 最大パケット棄却確率を変化させた時の安定領域 ( $B = 2$  [パケット/ms],  $N = 1$ ,  $\tau = 1$  [ms])

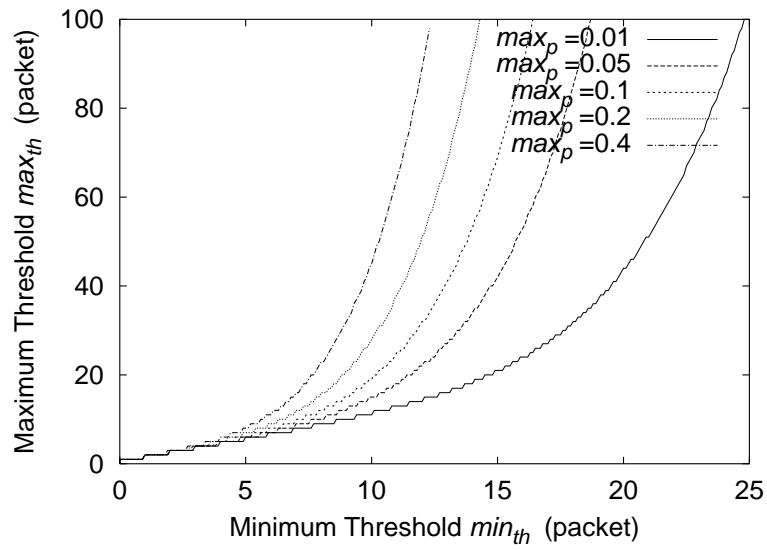


図 9: 最大パケット棄却確率を変化させた時の安定領域 ( $B = 2$  [パケット/ms],  $N = 5$ ,  $\tau = 1$  [ms])

タについては、図 8 と同じ値を用いている。また図 11 に、TCP コネクションの往復の伝搬遅延時間  $\tau = 5$  [ms] とした時の安定領域を示している。その他のパラメータについては、図 8 と同じ値を用いている。図 10 と図 11 を比較すると、安定条件の境界線がまったく同じであることがわかる。これは、対象とするネットワークモデルの安定性を考える場合、RED ルータの処理能力が大きくなることと、TCP コネクションの往復伝搬遅延時間が大きくなることが等価であることを意味している。このことは、式 (15)、(16)、(18) において、 $B$  および  $\tau$  がすべて  $B \times \tau$  という積の形で含まれていることからわかる。

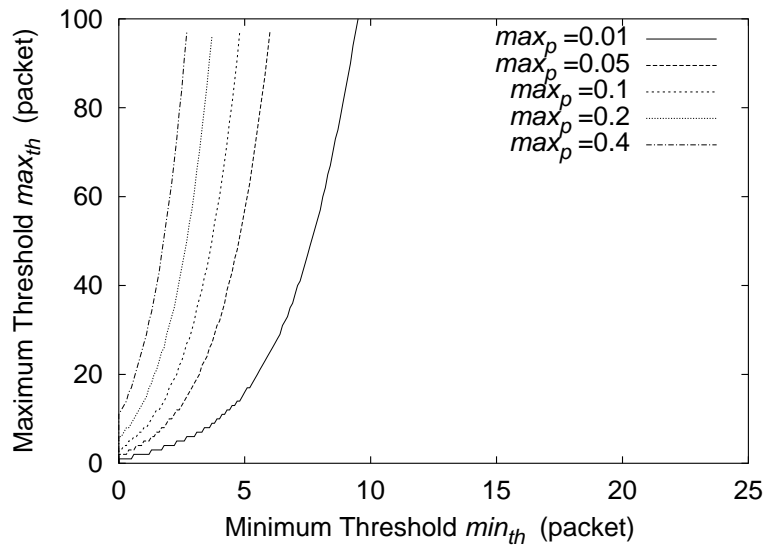


図 10: 最大パケット棄却確率を変化させた時の安定領域 ( $B = 10$  [パケット/ms],  $N = 1$ ,  $\tau = 1$  [ms])

4 章で説明したように、システムの安定性は状態遷移行列の特性方程式の解によって決まる。さらに、システムの過渡特性 (平衡点までの収束速度) もまた、特性方程式の解の大小によって決定される。図 12 および図 13 に、最小しきい値  $min_{th}$  および最大しきい値  $max_{th}$  を変化させた時の、特性方程式の解  $s_i$  ( $1 \leq i \leq 3$ ) における  $|s_i|$  の最大値を示す。これらの図は、それぞれ図 8 および図 9 において、最大パケット棄却確率  $max_p = 0.1$  の場合に対応している。X 軸は最小のしきい値  $min_{th}$ 、Y 軸は最大のしきい値  $max_{th}$ 、Z 軸は  $|s_i|$  の最大値を示している。 $|s_i|$  の値が小さいほど過渡特性が良い (すなわち、収束が速い) こと

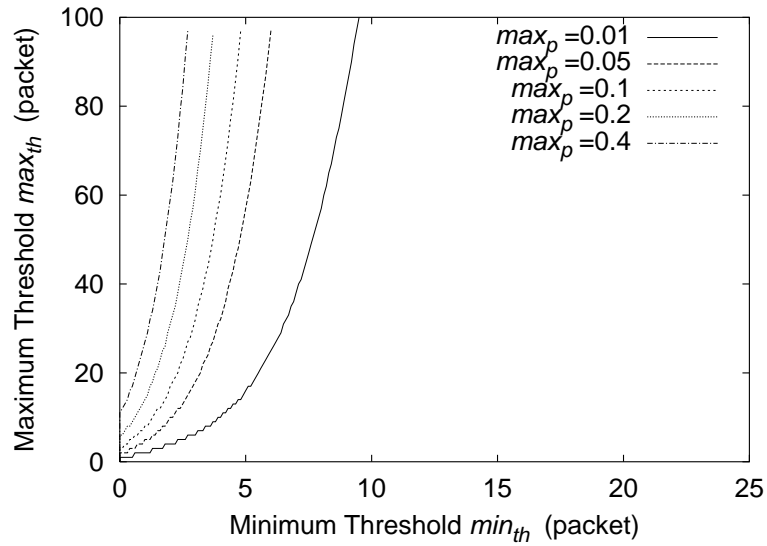


図 11: 最大パケット棄却確率を変化させた時の安定領域 ( $B = 2$  [パケット/ms],  $N = 1$ ,  $\tau = 5$  [ms])

を意味している。また、 $|s_i| \geq 1$  の領域ではシステムが不安定になることを意味している。これらの図から、システムの過渡特性を最適化する  $min_{th}$  の値は、 $max_{th}$  の値にほぼ無関係に決まることがわかる。さらに、 $min_{th}$  の値によりシステムの過渡特性を大きく変化することから、最小のしきい値  $min_{th}$  を慎重に選ぶ必要があることがわかる。

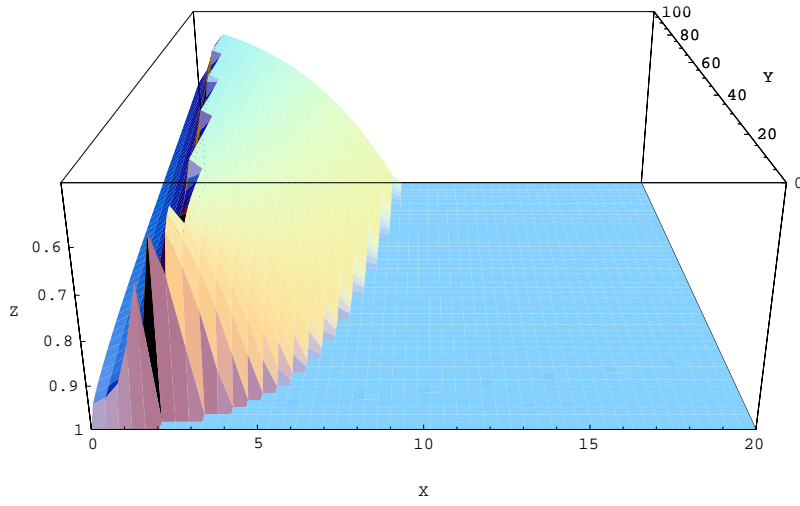


図 12: 制御パラメータと  $|s_i|$  の最大値との関係 ( $B = 2$  [パケット/ms],  $N = 1$ ,  $\tau = 1$  [ms],  $max_p = 0.1$ )

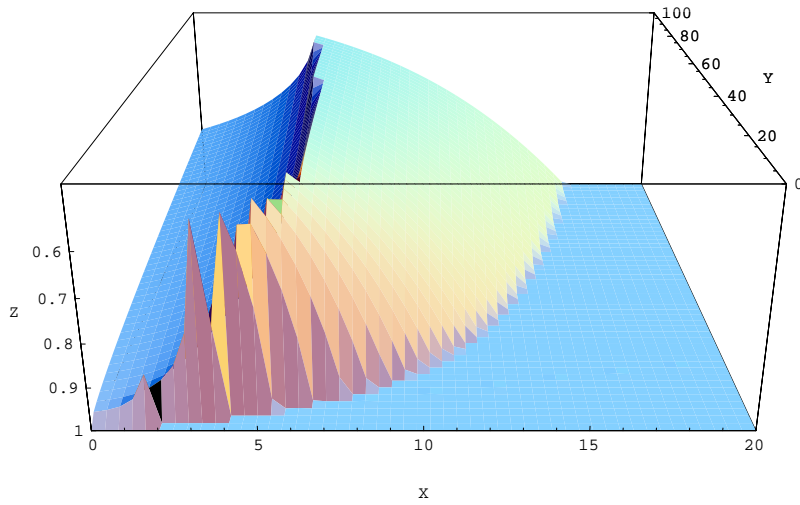


図 13: 制御パラメータと  $|s_i|$  の最大値との関係 ( $B = 2$  [パケット/ms],  $N = 5$ ,  $\tau = 1$  [ms],  $max_p = 0.1$ )

## 6 RED ルータの制御パラメータ決定方法の提案

以上の解析および数値例から、RED ルータの特性 (定常状態での平衡点、収束特性、過渡特性) は、その制御パラメータの設定に大きく依存することがわかった。例えば、エンド-エンド間の伝搬遅延時間に影響を与えるパケット待ち行列長の平衡点は、最大パケット棄却確率  $max_p$  に大きく依存することがわかった。また、TCP のコネクション数もしくは (RED ルータの処理能力  $\times$  TCP の往復の伝搬遅延時間) が大きくなるにつれ、制御パラメータの安定領域が大きくなることがわかった。さらに、過渡特性を最適化するには、最小のしきい値  $min_{th}$  を慎重に決定する必要があることがわかった。

そこで最後に、本報告で得られたこれらの結果にもとづき、RED ルータの制御パラメータの決定方法を提案する。RED ルータにおいてあらかじめ分かっているシステムパラメータは、RED ルータ自身の処理能力  $B$  と RED ルータのバッファサイズだけである。従って、その他のシステムパラメータである、TCP のコネクション数  $N$  や TCP コネクションの往復伝搬遅延時間  $\tau$  を、何らかの方法で測定もしくは推測する必要がある。ここでは、RED ルータの制御パラメータを以下のように決定すればよいと考えられる。

### 1. RED ルータにおいて TCP のコネクション数 $N$ を計測する

RED ルータにおいて TCP のコネクション数を推測するには、例えば、一定期間中にルータに到着するパケットのヘッダを解析し、送信側ホストの IP アドレス、受信側ホストの IP アドレス、ポート番号から TCP のコネクション数を推測する方法が考えられる。しかしこの方法では、観測期間が短かすぎる場合にはすべての TCP コネクションを検出できないという問題がある。また逆に、観測期間が長すぎる場合にはすでに転送を完了した TCP コネクションもカウントしてしまう。4 章の結果より、TCP のコネクション数  $N$  が増加するにつれ、制御パラメータの安定領域は大きくなるという傾向があることがわかっている。このため、ある程度観測期間を短くし、TCP のコネクション数を安全側に見積ればよいと考えられる。

### 2. RED ルータにおいてすべての TCP の往復伝搬遅延時間を計測し、その平均値 $\tau$ を計算する

一般にルータにおいて、各 TCP コネクションの往復の伝搬の遅延時間を推測すること

は困難である。本解析では、すべての TCP コネクションの伝搬の遅延時間が等しいと仮定している。しかし実際のネットワークにおいては、当然ながら各 TCP コネクションの往復の伝搬遅延時間は異なっている。例えば、RED ルータにおいてすべての TCP コネクションの往復の伝搬遅延時間を計測し、その平均値を利用するという方法が考えられる。しかし、単純に往復の伝搬遅延時間の平均値を利用してよいかどうかは、今後さらなる検討が必要である。ただし、4 章の結果より、TCP の往復の伝搬遅延時間  $\tau$  が増加するにつれ、制御パラメータの安定領域は大きくなるという傾向があることがわかっている。このため、安全側で動作させるためには、往復の伝搬遅延時間をある程度小さめに見積るべきと考えられる。

### 3. 最大のしきい値 $max_{th}$ をバッファサイズに設定する

2 章で説明したように、RED ルータは平均パケット待ち行列長が最大のしきい値  $max_{th}$  を超えた場合、到着するすべてのパケットを棄却する。このため、最大のしきい値  $max_{th}$  を、RED ルータのバッファサイズよりも小さく設定することは意味がない。このため、最大のしきい値  $max_{th}$  は単に RED ルータのバッファサイズに設定すればよいと考えられる。

### 4. 本報告の解析結果にもとづき、最大パケット棄却確率 $max_p$ および最小のしきい値 $min_{th}$ を決定する

最小のしきい値  $min_{th}$  および最大パケット棄却率  $max_p$  は、RED ルータの平均パケット待ち行列長やネットワーク全体の安定性や過渡特性に大きな影響を与える。このため、最大パケット棄却率  $max_p$  および最小のしきい値  $min_{th}$  については、3 章および 4 章で得られた解析結果にもとづいて慎重に設定する必要がある。つまり、最大パケット棄却率  $max_p$  は、式 (21) で与えられるパケット待ち行列長が大きくなりすぎない範囲で、できるだけ小さな値に設定する。また、最小のしきい値  $min_{th}$  は、4 章の状態遷移行列の特性方程式の解が最小になるように決定すればよい。

## 7 まとめ

本報告では、TCP によってフロー制御されたトラヒックに対する、RED ルータの特性を数学的に解析した。まず、ネットワーク全体を離散時間システムでモデル化し、システムの状態 (送信側ホストのウィンドウサイズ、RED ルータの現在パケット待ち行列長および平均パケット待ち行列長) が、時間的にどのように変動するかを記述する状態遷移方程式を導出した。ただし、RED ルータは到着するパケットを確率的に棄却するため、得られた状態方程式は確率を含んだものとなる。そこで、定常状態におけるネットワーク全体のふるまいを解析するため、TCP コネクションと RED ルータの平均的な動作を記述する、平均状態安定方程式を導出した。さらに、ここで得られた平均状態安定方程式を用いて、定常状態において TCP のウィンドウサイズや RED ルータの現在パケット待ち行列長、平均パケット待ち行列長が、平均的にどのような値に収束するかを導出した。さらに、平均状態方程式より、ネットワーク全体が平均的に安定するための条件を、現代制御理論を適用することにより導出した。また、いくつかの数値例により、RED ルータの制御パラメータを変更することにより、その特性がどのように変化するかを明らかにした。さらに、本解析で得られた解析結果とシミュレーション実験により得られた結果を比較することにより、本近似解析の妥当性を示した。これらの結果、エンド-エンド間の伝搬遅延時間に影響を与えるパケット待ち行列長の平衡点は、RED ルータの最大パケット棄却確率に大きく依存することがわかった。また、TCP のコネクション数もしくは  $(\text{RED ルータの処理能力}) \times (\text{TCP の往復伝搬遅延時間})$  が大きくなるにつれ、RED ルータの制御パラメータの安定領域が大きくなることがわかった。さらに、ネットワーク全体の過渡特性を最適化するには、RED ルータの最小のしきい値を慎重に決定する必要があることがわかった。また、以上の解析結果にもとづき、RED ルータの制御パラメータをどのように決定すればよいかを提案した。



## 謝辞

本報告を終えるにあたり、御指導、御教授を頂いた村田正幸教授に深く感謝致します。また、本報告において終始直接御指導頂いた情報処理教育センターの大崎博之助手に深く感謝致します。並びに適切な助言を頂いた宮原秀夫教授、宮原研究室の若宮直紀講師、経済学部の長谷川剛助手に心から感謝致します。最後に、御協力を頂いた村田研究室および宮原研究室の皆様心からお礼申し上げます。

## 参考文献

- [1] V. Jacobson, “Congestion avoidance and control,” in *Proceedings of SIGCOMM '88*, pp. 314–329, August 1988.
- [2] W. R. Stevens, *TCP/IP Illustrated, Volume 1: The Protocols*. New York: Addison-Wesley, 1994.
- [3] E. Hashem, “Analysis of random drop for gateway congestion control,” *Technical Report MIT-LCS-TR-465*, 1989.
- [4] S. Floyd and V. Jacobson, “Random early detection gateways for congestion avoidance,” *IEEE/ACM Transactions on Networking*, vol. 1, pp. 397–413, August 1993.
- [5] B. Barden et al., “Recommendations on queue management and congestion avoidance in the Internet,” *Request for Comments (RFC) 2309*, April 1998.
- [6] D. Lin and R. Morris, “Dynamics of random early detection,” in *Proceedings of ACM SIGCOMM '97*, September 1997.
- [7] M. May, J. Bolot, C. Diot, and B. Lyles, “Reasons not to deploy RED,” in *Proceedings of IWQoS '99*, pp. 260–262, March 1999.
- [8] W. chang Feng, D. D. Kandlur, D. Saha, and K. G. Shin, “A self-configuring RED gateway,” in *Proceedings of IEEE INFOCOM '99*, March 1999.

- [9] T. J. Ott, T. V. Lakshman, and L. Wong, “SRED: Stabilized RED,” in *Proceedings of IEEE INFOCOM '99*, March 1999.
- [10] M. May, T. Bonald, and J.-C. Bolot, “Analytic evaluation of RED performance,” to be presented at *IEEE INFOCOM 2000*, March 2000.
- [11] V. Sharma, J. Virtamo, and P. Lassila, “Performance analysis of the random early detection algorithm,” available at <http://keskus.tct.hut.fi/tutkimus/com2/publ/redanalysis.ps>, September 1999.
- [12] W. R. Stevens, “TCP slow start, congestion avoidance, fast retransmit, and fast recovery algorithms,” *Request for Comments (RFC) 2001*, January 1997.
- [13] 平井 一正, 池田 雅夫, 非線形制御システムの解析. オーム社, 1986.
- [14] 美多 勉, デジタル制御理論. 昭晃堂, 1984.
- [15] “LBNL network simulator (ns).” available at <http://www-nrg.ee.lbl.gov/ns/>.