

特別研究報告

題目

TCP コネクション間の公平性を改善するための
バックボーンルータにおける
RED の動的パラメータ設定方式の提案と評価

指導教官

村田 正幸 教授

報告者

板谷夏樹

平成 13 年 2 月 22 日

大阪大学 基礎工学部 情報科学科

TCP コネクション間の公平性を改善するための
バックボーンルータにおける RED の動的パラメータ設定方式の提案と評価

板谷夏樹

内容梗概

近年、インターネット利用者の急激な増加にともない、特にバックボーンルータにおける同時に通信を行なっているフロー数が増加している。その結果として、フロー数の多いポトルネックリンクでは、TCP のコネクション間の公平性が悪化するため、その点に着目した TCP の輻輳制御方式の改善に関するさまざまな研究が行われている。例えば、ルータのバッファにおいて RED (Random Early Detection) 方式を用いることで、TCP の不要なスループットの劣化を防ぎ、フロー間の公平性を向上することができるとされているが、フロー数が 1000 を越えるような場合の RED の評価はこれまで行われていない。

そこで本報告では、まずフロー数が非常に多いネットワーク環境における RED ルータと従来の TD (Tail Drop) ルータの性能を比較評価した。その結果、コネクション間の公平性は、その制御のためのパラメータ設定が適切な条件下では、RED 方式が TD 方式に対して優れているが、パラメータ設定が不適切な条件下では、RED 方式の性能が TD 方式よりもむしろ劣化する場合もあることが明らかとなった。すなわち、RED 方式の性能を十分に発揮させるためには、制御パラメータの設定を正しく行うことが重要である。しかし、ルータが収容するコネクション数が多い場合、アクティブフロー数、輻輳状況等が大きく変化するため、静的なパラメータ設定では、バックボーンルータの性能を十分に発揮させることは困難である。そこで本報告では、RED 方式のパラメータをルータの輻輳状況に応じて動的に変化させ、ネットワーク状況の変化に対応する動的パラメータ設定方式を提案し、性能評価を行った。その結果、TD 方式、および既存の RED 方式に比べて高い公平性を実現することができることが明らかとなった。

主な用語

インターネット, TCP (Transmission Control Protocol), RED (Random Early Detection),
Active Queue Management, 公平性, ルータ

目次

1	はじめに	8
2	ルータバッファにおけるパケット処理方式	11
2.1	TD (Tail Drop) 方式	11
2.2	RED (Random Early Detection) 方式	12
2.2.1	アルゴリズム	13
2.2.2	RED 方式の特徴	14
3	公平性の評価	16
3.1	関連研究	16
3.2	ネットワークトポロジー	17
3.3	公平性の評価基準	18
3.4	無限長ファイル転送の場合	18
3.5	Web トラフィックを考慮したドキュメント転送の場合	20
3.6	考察	28
4	RED の動的パラメータ設定方式	33
4.1	関連研究	33
4.2	提案方式	34
4.2.1	設計方針	34
4.3	アルゴリズム	35
5	提案方式の性能評価	38
5.1	無限長ファイルの転送の場合	38
5.2	Web トラフィックを考慮したドキュメント転送の場合	40
5.3	考察	44
6	おわりに	48

謝辭	49
参考文献	50

目次

1	RED 方式におけるパケット廃棄確率	14
2	ネットワークトポロジー	17
3	TD, RED 方式の公平性比較 (無限長ファイル転送, $BW = 1.5$ [Mbps], $\tau =$ 4 [msec])	21
4	TD, RED 方式の公平性比較 (無限長ファイル転送, $BW = 1.5$ [Mbps], $\tau =$ 400 [msec])	22
5	TD, RED 方式の公平性比較 (無限長ファイル転送, $BW = 150$ [Mbps], $\tau = 4$ [msec])	23
6	パラメータの違いによる RED 方式の公平性比較 (無限長ファイル転送, BW $= 1.5$ [Mbps], $\tau = 4$ [msec])	24
7	パラメータの違いによる RED 方式の公平性比較 (無限長ファイル転送, $BW =$ 1.5 [Mbps], $\tau = 400$ [msec])	25
8	パラメータの違いによる RED 方式の公平性比較 (無限長ファイル転送, BW $= 150$ [Mbps], $\tau = 4$ [msec])	26
9	TD, RED 方式の公平性比較 (Web ドキュメント転送, $BW = 1.5$ [Mbps], $\tau = 4$ [msec], $B = 100$ [packets])	29
10	TD, RED 方式の公平性比較 (Web ドキュメント転送, $BW = 1.5$ [Mbps], $\tau = 4$ [msec], $B = 1000$ [packets])	30
11	TD, RED 方式の公平性比較 (Web ドキュメント転送, $BW = 1.5$ [Mbps], $\tau = 4$ [msec], $B = 10000$ [packets])	31
12	TD 方式と RED 方式のパケット廃棄間隔	32
13	TCP コネクション数が動的な環境での dt-RED 方式の動作	39
14	TD, RED, dt-RED 方式の公平性比較 (無限長ファイル転送, $BW = 1.5$ [Mbps], $\tau = 4$ [msec])	41
15	TD, RED, dt-RED 方式の公平性比較 (無限長ファイル転送, $BW = 1.5$ [Mbps], $\tau = 400$ [msec])	42

16	TD, RED, dt-RED 方式の公平性比較 (無限長ファイル転送, $BW = 150$ [Mbps], $\tau = 4$ [msec])	43
17	TD, RED, dt-RED 方式の公平性比較 (Web ドキュメント転送, $BW = 1.5$ [Mbps], $\tau = 4$ [msec], $B = 100$ [packets])	45
18	TD, RED, dt-RED 方式の公平性比較 (Web ドキュメント転送, $BW = 1.5$ [Mbps], $\tau = 4$ [msec], $B = 1000$ [packets])	46
19	TD, RED, dt-RED 方式の公平性比較 (Web ドキュメント転送, $BW = 1.5$ [Mbps], $\tau = 4$ [msec], $B = 10000$ [packets])	47

表 目 次

1	RED 方式の制御パラメータ	18
2	dt-RED 方式の制御パラメータ	38

1 はじめに

インターネット利用者が急速に増加し、それとともにインターネットにおいて展開されるさまざまなサービスが社会生活に浸透しつつある今日、インターネットは誕生した当時には想定されていなかったような急激な発展をみせている。そのなかで、徐々に進むアクセス回線の高速化、さらに回線高速化にともなうアプリケーションの変化などに伴い、インターネット上を流れるトラフィックは質、量ともに急激な変化をみせている。特にバックボーンルータに収容される、同時に通信を行なっているフロー（以下、アクティブフロー）数が急速な増加をみせ、バックボーンリンクでの輻輳が大きな問題となっている。

これまでインターネットにおける輻輳の原因や輻輳制御に関してさまざまな研究が行われてきた。なかでも、現在のインターネットでのトラフィックの多くの部分を占める TCP (Transmission Control Protocol) [1] の向上に関する研究が盛んに行われてきている。特に上述の問題点を背景として、最近、アクティブフロー数の多い場合の TCP の動作についての研究も行なわれつつあり、例えば文献 [2] では、フロー数の多いボトルネックリンクでは、帯域を公平かつ効率的に共有するための TCP の輻輳制御機能が効果的に機能せず、TCP コネクション間での公平性が劣化する場合があることが指摘されている。そこで、このような公平性の劣化を改善するために、TCP の輻輳制御方式に関して、いくつかの研究が行われている [3-6]。

TCP の輻輳制御方式は、インターネットにおける基本的な概念であるエンド間制御を基本とし、ネットワーク内での制御は前提としない。しかし、ボトルネックリンクでの TCP フロー間の公平性を改善するためには、エンド間輻輳制御に加えてルータにおいても積極的な輻輳制御を行うことが必要である。現在、インターネット上のルータで一般的に実装されているバッファ制御方式は、パケット到着順にバッファに格納し、バッファが一杯になると到着パケットを廃棄する Tail Drop 方式である。Tail Drop 方式は構造が単純なため、実装が簡単であるが、パケットのスケジューリングに関しては特別な操作は行わないため、特にアクティブフロー数が多い場合に公平性の劣化やリンク利用率の低下といった問題が発生する。これは、輻輳時のバッファ溢れによる連続的なパケット損失の発生が原因で TCP のタイムアウトが発生するためである。

そこで、TCPの輻輳制御方式の改善に関する研究と並んで、ルータでのフロー間の公平性や、スループットを向上させるためのさまざまルータバッファ制御方式が研究されている。それらの制御方式の中で、1993年に文献[7]において提案されたRED (Random Early Detection)方式が、比較的オーバーヘッドが少ないまま、Tail Drop方式に比べて高いスループットを維持しつつ、公平性を改善できるといわれており、実際RED方式についてさまざまな研究が行われている [8-13]。それらの関連研究については3.1節で詳しく述べるが、ほとんどの研究では、ルータでの収容コネクション数が1~100本程度を想定しており、フロー数が1000を超えるようなバックボーンルータでRED方式を使用した場合の性能評価はこれまで行われていない。

そこで本報告ではまず、フロー数が非常に多いネットワーク環境において、REDルータとインターネット上で従来用いられているTail Dropルータとの性能比較評価を行う。評価に際しては、特にコネクション間の公平性に着目する。その結果、RED方式に必要な制御パラメータであるパケット廃棄確率やしきい値等の設定が適切に行われている場合、RED方式はTail Drop方式に対して優れた公平性を示すが、制御パラメータの設定が適切でない場合には、RED方式の性能はTail Drop方式のそれよりも劣化することを明らかにする。

RED方式のパラメータ設定に関する研究についても多く行われている [7, 12-14] が、そこで提案されているパラメータ群は、特定のネットワーク環境においては有効であるが、ネットワークの環境が変化するとRED方式の性能が低下してしまう。特に、本報告で対象とするようなルータが収容するコネクション数が多い場合には、アクティブフロー数、バッファ内パケット数(キュー長)、輻輳状況といったネットワークの環境が大きく変化するため、常に高い性能を示すようなRED方式のパラメータの設定を行うことは困難である。すなわち、バックボーンルータにRED方式を使用しても、公平性の改善に十分な性能を発揮できない。アクティブフロー数やそれぞれのフローからのパケット到着率などのネットワーク状況が変化するような環境において、RED方式のパラメータを動的に設定する方法は文献[15-17]などで提案されているが、それらの方式では、主にREDのパケット廃棄率を動的にコントロールしている。しかし、TCPフローのスループットはパケット損失に対して敏感で[18]、パケット損失率を変化させるとスループットやリンク利用率の低下を招くことが予測され

る。また、メモリ価格が低廉になる中で、以前に比べルータにおいてバッファを多く準備することが可能になっているにもかかわらず、パケット廃棄率を制御する方式では、豊富なバッファを十分に利用することができない。

そこで本報告では、RED 方式のパラメータをルータの輻輳状況に応じて動的に変化させ、ネットワーク状況の変化に対応する動的パラメータ方式を提案し、その評価を行う。提案方式においては、上記の考察に基づき、バッファの有効利用を図りながら平均キュー長を安定させるために、パケット廃棄率ではなくパケット廃棄の契機となるしきい値を動的にコントロールする。そして、提案方式によって、Tail Drop 方式、および既存の RED 方式に比べて高い公平性を実現し、またリンク利用率の低下も抑えることができることを、シミュレーション結果を通して明らかにする。

以下、2 章では、Tail Drop 方式、および RED 方式の説明、3 章では、Tail Drop 方式と RED 方式との性能評価比較の結果とその考察、4 章では、本報告で提案する RED 方式の動的パラメータ設定方式の設計方針、アルゴリズムを説明し、5 章でその性能評価と考察を行う。最後に 6 章で、本報告のまとめと今後の課題について述べる。

2 ルータバッファにおけるパケット処理方式

本章では、本報告で対象としているルータバッファでのパケット処理方式である TD (Tail Drop) 方式と RED (Random Early Detection) 方式について説明する。

2.1 TD (Tail Drop) 方式

TD (Tail Drop) 方式はパケットスケジューリングに複雑な処理を必要としないため実装が簡単であり、現在のインターネットで広くルータバッファの制御方式として採用されている。また、処理のオーバーヘッドも少ないため高速な処理が要求されるバックボーンルータなどにも適応できる。

TD 方式のルータバッファでは、パケット処理のスケジューリングは FCFS (First Come, First Service)、すなわち到着順で行われる。バッファの使用に関しても特別な動作は行わず、処理待ちのパケットが順にバッファに蓄積され、バッファが一杯になった時点で到着パケットはすべて廃棄される。この状態をバッファ溢れと呼ぶ。この状態では、到着パケットはすべて廃棄されるため、バースト的 (連続的) なパケット損失が発生しやすい。バースト的なパケット廃棄は、TCP コネクションのスループットを大きく低下させる原因となる。TCP の輻輳制御機構は、パケット損失が生じると TCP のウィンドウサイズを減少させて送信するデータ量を減らすことにより、輻輳を回避しようとする。問題は、バースト的なパケット損失によって特定のコネクションの送信パケットが連続して廃棄されると TCP の Fast Retransmit [1] による再送が有効に働かなくなり、タイムアウトが起こりやすくなる点にある [7]。

さらに、そのルータを経由している多くの TCP コネクションが同期して、バースト的なパケット損失によって送信データ量を減少させると、ルータでのリンク利用率が悪化する。これは Global Synchronization 問題と呼ばれている [7]。TD 方式は基本的にバースト的にパケットを送出するコネクションに対して不利に働く傾向を持つ。TCP によるデータ転送においては、通信の開始時やタイムアウト後などにバースト的にパケットを送信するが、TD 方式においてはそれらバースト性の高いコネクションのパケットは廃棄されやすいため、特

に双方向のデータ通信など、短い間隔でデータの送受信が発生するようなアプリケーションが不利になる傾向がある [7]。

また、リンクを共有する各コネクションの packets 到着レートに関係なく packets 廃棄が発生することや、上述のようにタイムアウトが多く発生することなどにより、TD 方式においてはコネクション間の公平性が低下する。

2.2 RED (Random Early Detection) 方式

TD 方式には 2.1 節で述べたような問題点が存在するため、それを改善するためにさまざまなルータバッファ管理方式が提案されてきている [19-21]。そのなかで、バッファ溢れによってバースト的な packets 損失が発生することで起こる、TCP のタイムアウトやスループットの低下を防止することを目的としたルータバッファ管理方式として、RED (Random Early Detection) [7] が提案されている。

RED 方式の基となっているのは、文献 [22, 23] などで研究された ERD (Early Random Drop) 方式である。これはルータバッファ内の処理待ち packets 数 (以下、キュー長と呼ぶ) があるしきい値を越えると、一定の確率で packets を廃棄する方式である。この方式の目的は、ルータバッファが一杯になる前に確率的に packets を廃棄することで、TCP の輻輳制御機構によって packets 送信レートが適切なレベルに抑えられ、バッファ溢れによるバースト的な packets 廃棄を回避することにある。

一方、RED 方式は、確率的な packets 廃棄を行うことでバースト的な packets 廃棄の発生を防止するという ERD 方式の考えを継承しながら、文献 [24] の平均キュー長を利用した輻輳検知方式を利用することで平均キュー長を積極的に制御することでより安定した輻輳制御を行うことを目的としている。RED 方式では ERD 方式と同様に、バッファ溢れで packets がバースト的に廃棄されることを防止するために、バッファが一杯になる前に確率的な packets 廃棄を開始する。この際、ローパスフィルタを用いて実際のキュー長から計算された平均キュー長を用いてバッファの状態を監視することで、バースト的なトラヒックが引き起こす短期間の輻輳などによるキュー長の変動を許す一方、平均キュー長の変化から長期的な輻輳を検知して輻輳制御を行い、バッファが一杯になることを防止する。

また、パケット廃棄率を平均キュー長の増加にしたがって線形に増加させることで、ネットワークの輻輳状況に応じた制御を実現し、平均キュー長を安定させている。これにより、必要以上にパケットを廃棄してルータのスループットが低下することなどの問題を回避している。

2.2.1 アルゴリズム

RED 方式ではパケット到着毎に、ローパスフィルタの一種である指数平均 (EWMA: Exponential Weighted Moving Average) を用いて現在のキュー長から平均キュー長 \bar{q} を計算する。すなわち、現在のキュー長を q とすると、RED ルータはパケット到着毎に、平均キュー長 \bar{q} の値を以下の式で更新する。式中の w_q は平均キュー長に対する実際のキュー長の重みを決定する RED 方式の制御パラメータである。

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

この平均キュー長 \bar{q} を 2 つのしきい値 minimum threshold (min_{th}) および、maximum threshold (max_{th}) と比較して到着パケットに対する処理を決定する。

平均キュー長を計算した後、その値に基づいて以下のようにパケット処理を決定する。

1. $\bar{q} \leq min_{th}$ ならば到着パケットをバッファへ
2. $min_{th} \leq \bar{q} \leq max_{th}$ ならば以下パケット廃棄確率 p_a を計算し、確率 p_a で到着パケットを廃棄し

$$p_b \leftarrow \frac{\bar{q} - min_{th}}{max_{th} - min_{th}} max_p \quad (2)$$

$$p_a \leftarrow \frac{p_b}{1 - count \cdot p_b} \quad (3)$$

とする

3. $max_{th} \leq \bar{q}$ ならば到着パケットを廃棄

図1は平均キュー長 \bar{q} とパケット廃棄率 p_b との関係を表している。図のように、 \bar{q} が min_{th} から max_{th} の間にあるときは、 \bar{q} に対してパケット廃棄率は線形的に大きくなり、 max_{th} を越えると廃棄率は1となり、すべての到着パケットを廃棄する。

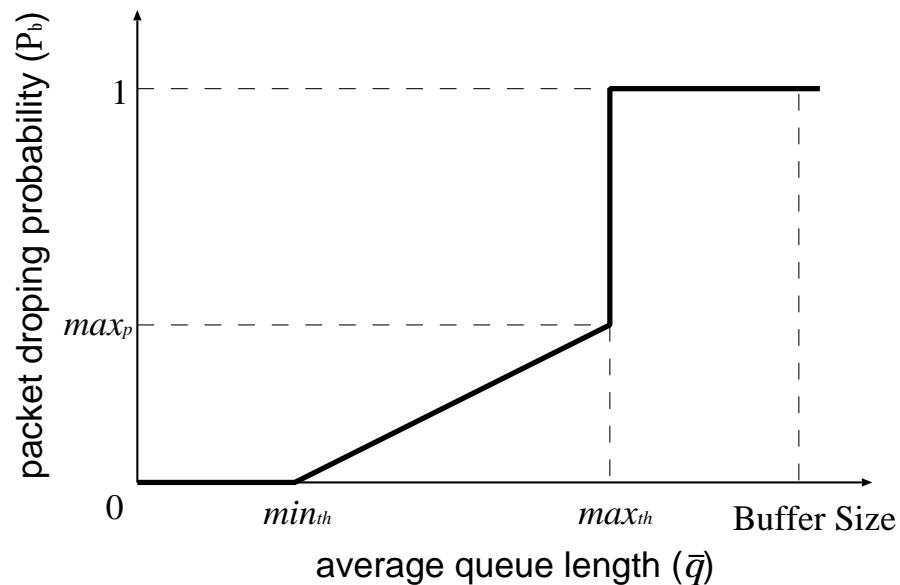


図 1: RED 方式におけるパケット廃棄確率

2.2.2 RED 方式の特徴

RED 方式の最大の特徴は、ルータにおけるバースト的なパケット廃棄を防止することである。バースト的なパケット廃棄を防止することで、TCP の Fast Retransmit 機能による再送を有効に利用し、ネットワーク全体のスループットが下がることを避けながら、効果的に輻輳を抑えることが可能となる。さらに、確率的なパケット廃棄がコネクションに関係なく一定に発生するため、各コネクションの到着レートに比例したパケット廃棄が行われ、コネクション間の公平な帯域割当が可能となると考えられる。

また、ERD 方式と異なり、ローパスフィルタを用いて計算された平均キュー長を基にパケット廃棄率を決定するため、バースト的にパケットを送出するコネクションによって引き起こされる、瞬間的なキュー長の増加に対してパケット廃棄率を過敏に増加させない。これ

により、TD方式にみられるバースト的なトラフィックに対する不公平性を改善することができる。

3 公平性の評価

本章では、ボトルネックルータが収容する TCP フロー数が多い環境でのフロー間の公平性に特に着目し、シミュレーション結果を通じて TD 方式と RED 方式の性能評価を行う。シミュレーションには、ネットワークシミュレータ ns [25] を用いる。

3.1 関連研究

RED 方式の性能評価に関してはこれまでもさまざまな研究が行われている [8-12]。例えば文献 [8, 11] はさまざまなネットワーク環境における RED 方式の性能を評価している。文献 [8] では、フロー数が多い環境での RED 方式の評価を行い、RED 方式の利点としてパケット損失率が小さくなる点が指摘されているが、想定しているフロー数が高々100 前後であり十分な評価とは言えない。文献 [11] では、FTP セッションと HTTP セッションの混在環境、TCP コネクションと UDP コネクションの混在環境、各 TCP コネクションの RTT が違う場合などさまざまな環境において RED 方式の性能が評価されているものの、どの場合も対象としているコネクション数が少ない。

文献 [10] は、RED 方式では各フローのパケット廃棄率はパケット送信レートに比例するが、そのことが必ずしもフロー間の公平性には寄与しないとして、RED 方式の公平性を改善する能力については効果が小さいとしている。また、文献 [9, 12] は RED 方式の有効性そのものに疑問を投げかけている。文献 [9] では、RED 方式はバースト的なトラヒックがルータに到着した際に、バースト的でないトラヒックのパケット廃棄率も上昇させる傾向があることや、Global Synchronization 問題を防ぐ効果が小さいこと、ルータバッファでの待ち時間の変動があることなどを指摘し、RED 方式の有効性を否定している。また文献 [12] でも RED 方式は TD 方式と比べて大きなスループットの改善がみられず、また制御パラメータの設定が困難であるため、RED 方式の有効性に疑問があるとしている。しかし、これらの評価ではフロー間の公平性の改善に関する評価はまったく行われておらず、またフロー数が非常に多い場合の評価は行っていない。

このように、RED 方式については多くの研究がなされているものの、フロー数が非常に

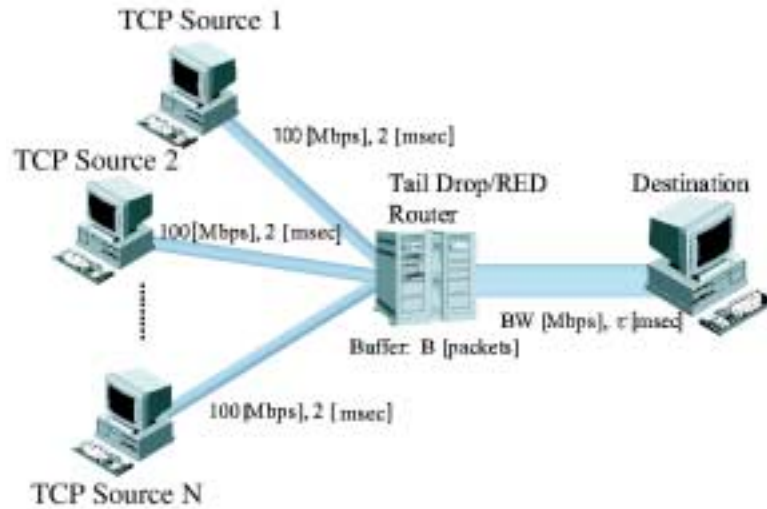


図 2: ネットワークトポロジー

多い環境での評価や、フロー間の公平性の観点から評価を行なった研究はほとんど行なわれていないのが現状である。

3.2 ネットワークトポロジー

図 2 にシミュレーションで用いるネットワークトポロジーを示す。ここでは、 N 台の送信側端末 (ルータが収容する TCP コネクション数に相当) が帯域、及び伝搬遅延時間が同一であるリンクで 1 台のルータに接続され、ルータから受信側端末へのリンクをボトルネックリンクとして共有しているネットワークを想定している。送信側端末とルータ間の各リンクの帯域は 100 [Mbps]、伝搬遅延時間を 2 [msec] とする。また、ボトルネックリンクの帯域を BW [Mbps]、伝搬遅延時間を τ [msec]、ルータのバッファサイズを B [packets] とし、これらの値を変化させることにより、さまざまな環境における評価を行う。ルータのバッファでは TD 方式あるいは RED 方式を用いる。RED 方式の初期パラメータを表 1 に示す。これは文献 [7] において推奨されている値である。

パラメータ	値
max_{th}	15 [packets]
min_{th}	5 [packets]
max_p	0.1
w_q	0.002

表 1: RED 方式の制御パラメータ

3.3 公平性の評価基準

本報告では、公平性の評価指標として Fairness Index 値 [26] を用いる。Fairness Index 値は以下のように定義される。 n をサンプル数、 x_i ($1 \leq i \leq n$) を n 個のサンプル値とすると、 n 個のサンプル値の公平性を表す指標 f を、

$$f = \frac{(\sum_{i=1}^n x_i)^2}{n \sum_{i=1}^n x_i^2} \quad (1 \leq i \leq n) \quad (4)$$

とする。 f は 0 以上 1 以下の値を取り、1 に近いほど公平性が高い。

3.4 無限長ファイル転送の場合

本節では、FTP (File Transfer Protocol) によるファイル転送等のアプリケーションを想定し、各送信側端末が無限長のファイルを転送する場合の公平性を評価する。公平性の指標としては式 (4) を用いるが、ここでは、 N 台の各送信側端末が受信側端末へ 1000 秒間のデータ転送を行った場合の各 TCP コネクションのスループットを 100 秒毎に計測し、サンプル値 x_i とすることによって、式 (4) による公平性の評価を行う。ただし、初期動作の影響を無視するため、シミュレーション開始から 100 秒間の結果は無視する。

図 3 は、図 2 のネットワークにおいて $BW = 1.5$ [Mbps]、 $\tau = 4$ [msec] とし、バッファサイズ $B = 100$ [packets] (図 3(a))、1000 [packets] (図 3(b))、及び 10000 [packets] (図 3(c)) とした場合の、コネクション数 N に対する Fairness Index 値の変化を示したものである。図 3(a)–3(c) をみると、コネクション数が多くなると RED 方式の公平性が急激に劣化し、TD 方

式と比較すると大きく劣っていることがわかる。これは、RED方式が $max_{th} = 15$ としているため、TD方式と比較して利用できるバッファサイズが小さくなり、TD方式よりもパケット損失の発生が多くなるためである。また、TD方式の場合、 $B = 100$ [packets] (図3(a)) の場合には、コネクション数が増えると公平性が悪化しているが、 $B = 1000$ [packets] (図3(b)) や 10000 [packets] (図3(c)) の場合では、公平性はコネクション数の増加とともに劣化すると同時に、コネクション数が少ない場合でも公平性が劣化している。これは、コネクション数に対してバッファサイズが大きすぎる場合、各コネクションのバッファ占有率に大きなばらつきが発生するため、コネクション間の公平性が劣化することを示している。

次に図4は、伝搬遅延時間 τ を 400 [msec] とした場合の結果を示している。ここでも、RED方式はコネクション数の増加にともなって急激に公平性を劣化させ、TD方式との比較でも大きく劣っている。TD方式では、コネクション数が少ない場合は、バッファサイズが変化しても公平性は大きくは変化しないが、図4(c)に示すように、コネクション数が多い場合には、バッファサイズを大きくすることによって公平性が改善していることがわかる。これは、TCPコネクションが 1000 本と非常に多い場合には、ある程度大きなバッファが必要であることを示している。図5は、 $BW = 150$ [Mbps]、 $\tau = 4$ [msec] とした場合の結果を示したものである。この図から、リンク帯域が大きくなることによって、特にRED方式における公平性が改善され、コネクション数が大きくなっても劣化していないことがわかる。これは、リンク帯域が十分であることにより、キュー長がそれほど大きくなりなため、RED方式においてバースト的なパケット廃棄がそれほど発生しないためである。

また、図3-5において、バッファサイズが大きくなってもRED方式の性能には大きな影響がないことがわかる。これは、 max_{th} が 15 と小さく、平均キュー長が 15 を越えることはないために、バッファサイズが変化してもバッファ利用率がそれほど変化しないことを示している。

以上の結果から、RED方式におけるパラメータに関して文献[7]に示されている推奨値では、特にルータが収容するフロー数が多い場合に性能が非常に劣化することがわかった。これは、パラメータ max_{th} が、バッファサイズ B に関係なく 15 に固定されているため、ルータのバッファが十分大きい場合にも、そのバッファを有効に利用できないためである。従っ

て、 max_{th} をバッファサイズ等に応じて変化させれば、RED 方式の性能が向上することが予測できる。

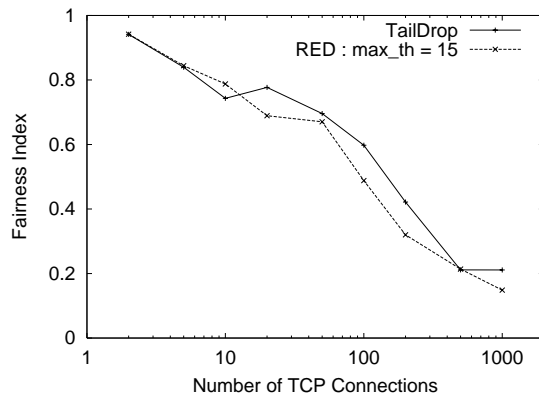
図 6-8 は、それぞれ図 3-5 に対応しており、RED 方式のパラメータ max_{th} をさまざまな値に変化させた場合の公平性の変化を示したものである。これらの結果から、 max_{th} をバッファサイズに合わせてある程度大きく設定することによって、RED 方式の公平性が大きく向上することがわかる。これは、 max_{th} を大きくすることによって、RED 方式における平均キュー長が max_{th} に達することが少なくなり、RED 方式による連続的なパケット廃棄を回避することができるためである。しかし、 max_{th} を大きくしても、TD 方式に比べて公平性が劣っている場合もある (例えば図 7(c)、8(b))。これは、これらの環境においては、TD 方式においてもバッファ溢れがほとんど発生せず、パケット廃棄が起こっていないためである。このような場合は、RED 方式によって強制的にパケット廃棄を発生させることによってフロー間の公平性が劣化する。

さらに、図 6-8 から、RED 方式におけるフロー間の公平性は max_{th} に大きく依存することがわかる。これは、RED 方式におけるパラメータ設定の困難さを示している。すなわち、どのようなネットワーク環境 (フロー数、バッファサイズ、ボトルネックリンクの帯域等) においても良い性能を示す RED 方式のパラメータは存在しないことを示している。

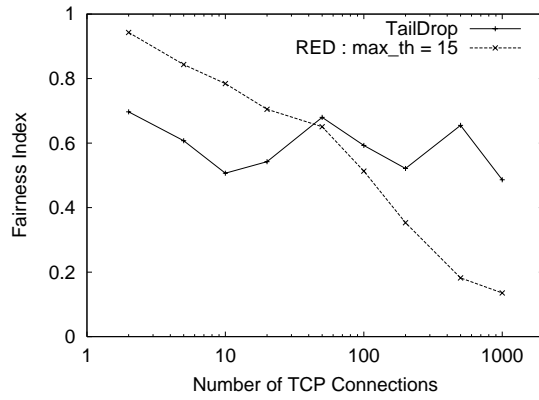
3.5 Web トラフィックを考慮したドキュメント転送の場合

次に、実際のインターネットにおける Web トラフィックを考慮したドキュメント転送を行った場合の、公平性の評価を行う。ここでは、文献 [27] において示されているインターネットにおける実際の Web アクセスの性質を考慮し、ドキュメントサイズ、リクエスト間隔等の分布を決定し、その分布に基づいてドキュメント転送のシミュレーションを行なった。公平性の評価には 3.4 節と同様に式 (4) を用いるが、ここでは同じサイズのドキュメントを転送したときの転送時間をサンプル値 x_i とし、それぞれのドキュメントサイズについて Fairness Index 値 f を求める。すなわち、ここでは同じサイズのドキュメント転送が等しい時間で終了することが公平であると定義する。

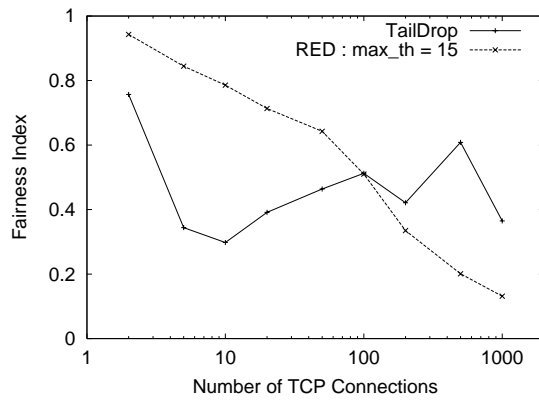
図 9-11 は $BW = 1.5$ [Mbps]、 $\tau = 4$ [msec]、 $B = 100-10000$ [packets]、及びコネクショ



(a) $B = 100$ [packets]

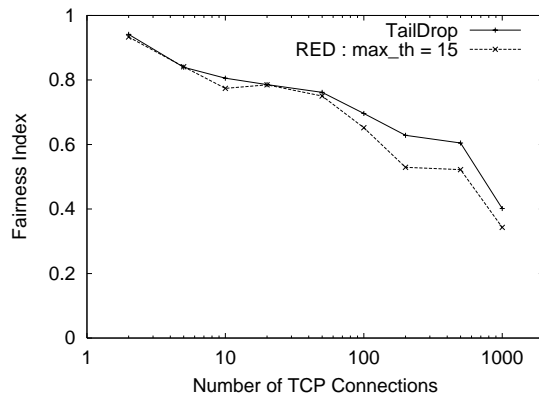


(b) $B = 1000$ [packets]

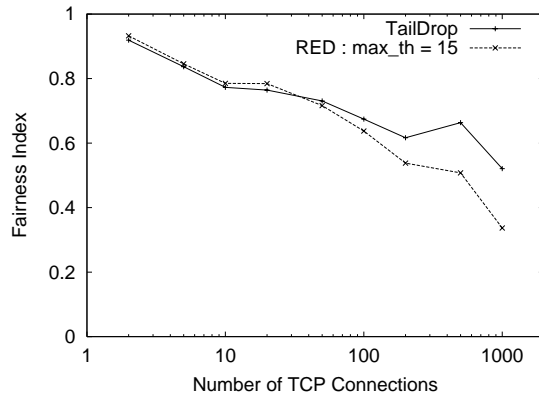


(c) $B = 10000$ [packets]

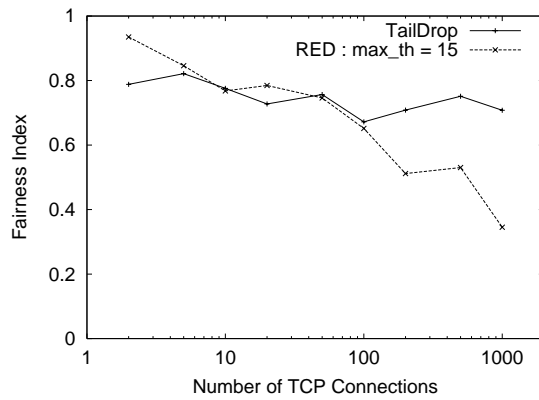
図 3: TD, RED 方式の公平性比較 (無限長ファイル転送, $BW = 1.5$ [Mbps], $\tau = 4$ [msec])



(a) $B = 100$ [packets]

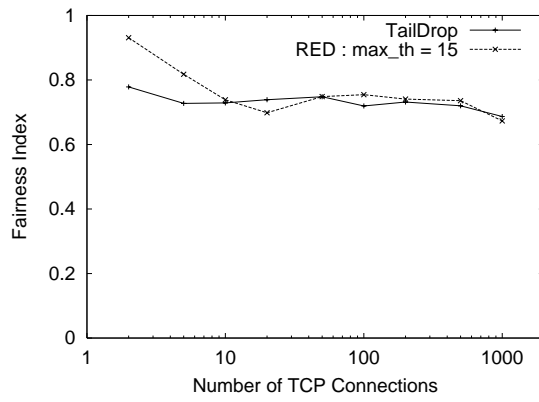


(b) $B = 1000$ [packets]

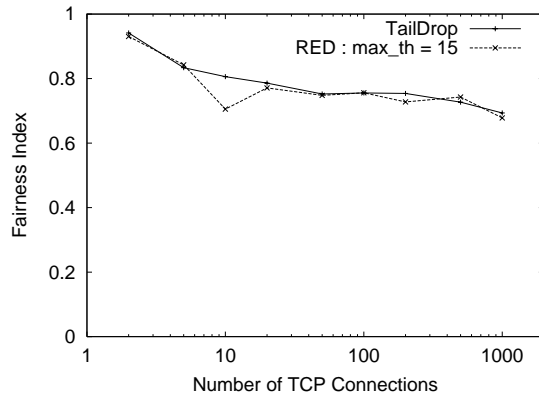


(c) $B = 10000$ [packets]

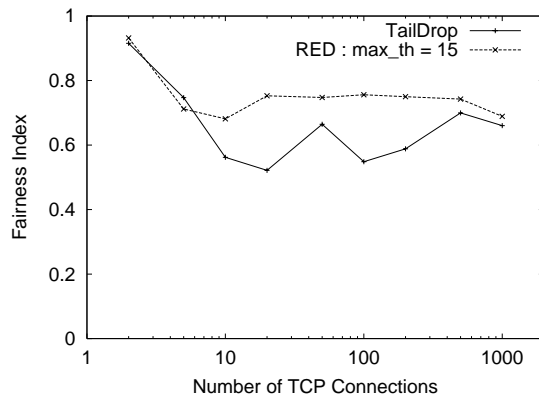
図 4: TD, RED 方式の公平性比較 (無限長ファイル転送, $BW = 1.5$ [Mbps], $\tau = 400$ [msec])



(a) $B = 100$ [packets]

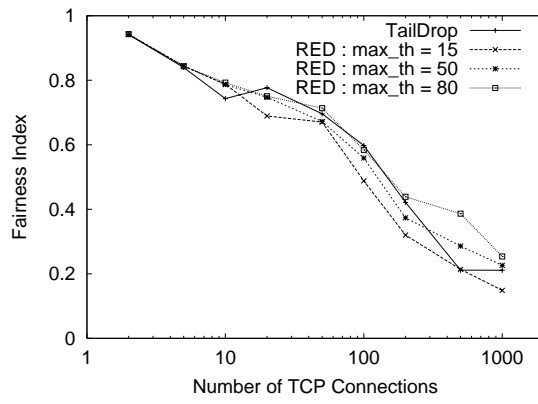


(b) $B = 1000$ [packets]

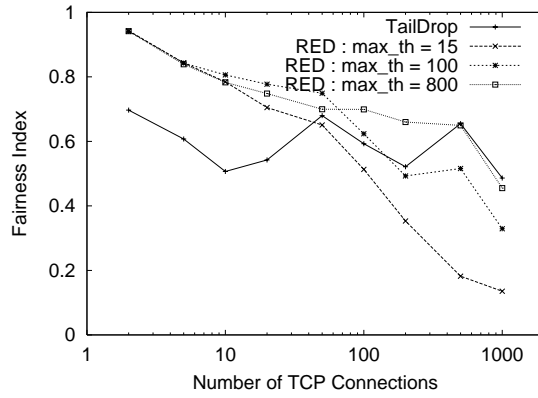


(c) $B = 10000$ [packets]

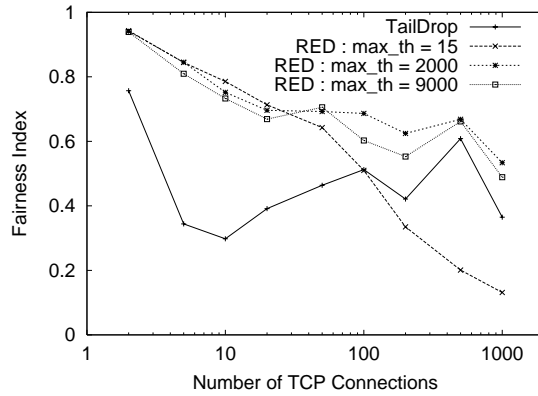
図 5: TD, RED 方式の公平性比較 (無限長ファイル転送, $BW = 150$ [Mbps], $\tau = 4$ [msec])



(a) $B = 100$ [packets]

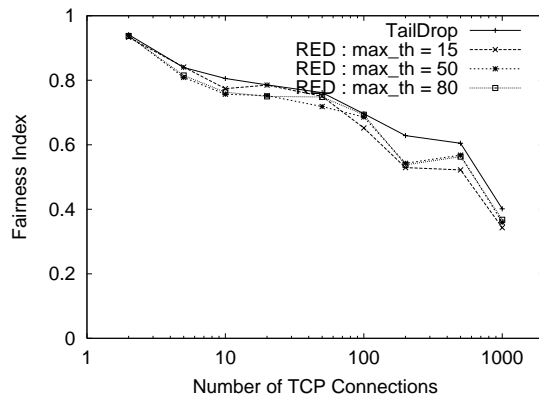


(b) $B = 1000$ [packets]

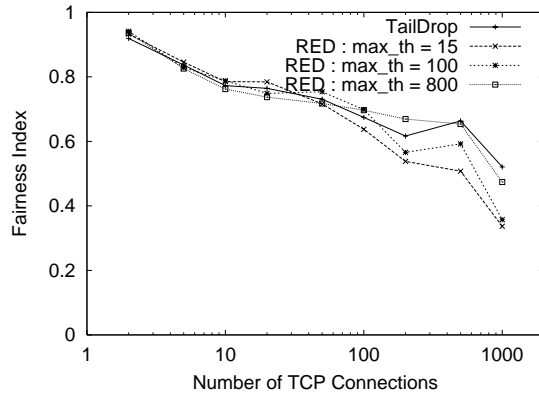


(c) $B = 10000$ [packets]

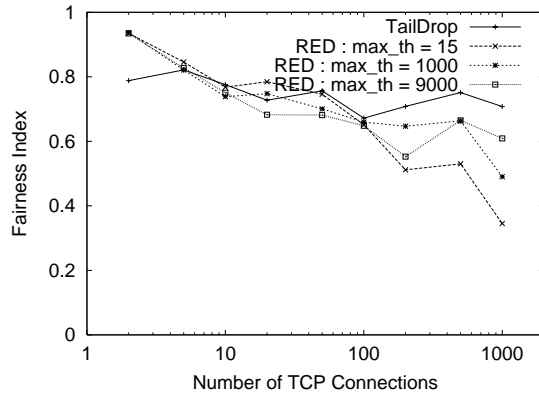
図 6: パラメータの違いによる RED 方式の公平性比較 (無限長ファイル転送, $BW = 1.5$ [Mbps], $\tau = 4$ [msec])



(a) $B = 100$ [packets]

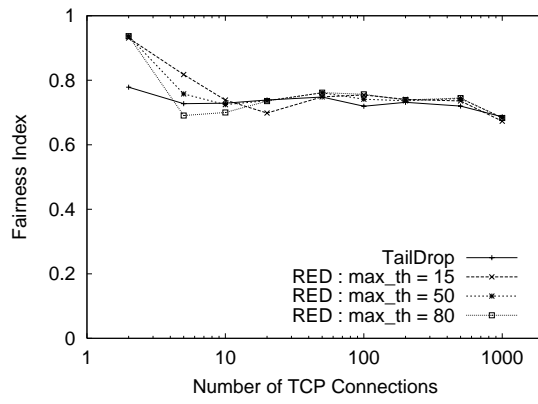


(b) $B = 1000$ [packets]

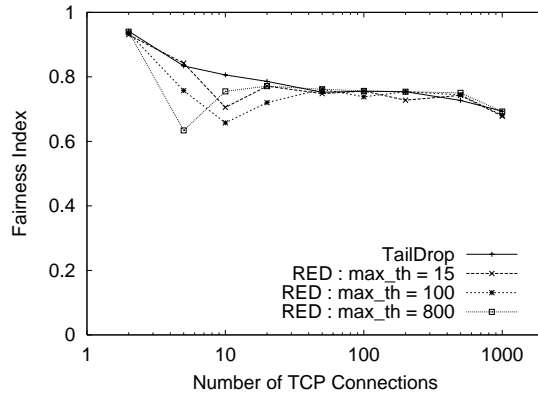


(c) $B = 10000$ [packets]

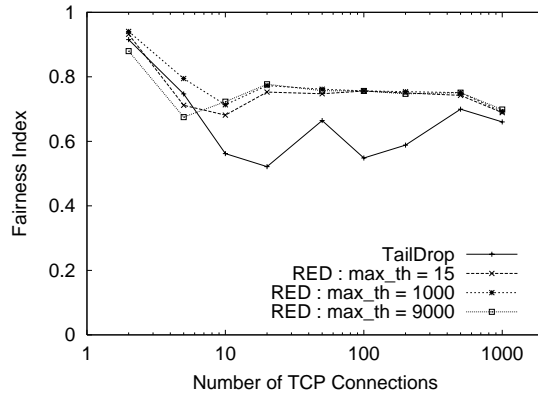
図 7: パラメータの違いによる RED 方式の公平性比較 (無限長ファイル転送, $BW = 1.5$ [Mbps], $\tau = 400$ [msec])



(a) $B = 100$ [packets]



(b) $B = 1000$ [packets]



(c) $B = 10000$ [packets]

図 8: パラメータの違いによる RED 方式の公平性比較 (無限長ファイル転送, $BW = 150$ [Mbps], $\tau = 4$ [msec])

ン数 N を 10、100、1000 本とした時の、ドキュメントサイズと Fairness Index 値の関係を示したものである。図から、TD 方式においては無限長ファイル転送の場合と同様、コネクション数が増えると公平性が劣化し、また、バッファサイズがコネクション数に対して大きくても小さくても公平性が劣化していることがわかる。また、ドキュメントサイズが小さい場合の公平性が劣化していることもわかる。これは、小さいドキュメントの転送では、転送に必要なパケットの個数が少ないため、TCP の Fast Retransmit 機構によるパケットの再送が行なわれず、タイムアウトが発生しやすくなるためである。

また、RED 方式の推奨値である $max_{th} = 15$ の場合は、特にコネクション数が多い場合 (例えば図 9(b)、9(c)) に公平性が大きく劣化している。さらに、バッファサイズが大きい場合 (図 10(c)、11(c)) においても、無限長ファイル転送の場合と同様に公平性が劣化している。このことは、[7] に示されている RED 方式の推奨パラメータである $max_{th} = 15$ では、バッファを有効に利用できず、公平性が維持できないことを示している。また、 max_{th} を大きくすることによって、RED 方式の公平性は大きく向上するが、特に小さいドキュメント転送の場合には、RED 方式の公平性が TD 方式よりも劣化していることがわかる。これは、RED 方式ではパケット廃棄の発生がほぼ定期的な発生するため、小さいドキュメントの転送においてもパケット損失が発生しやすく、その結果タイムアウトが発生するのに対し、TD 方式ではパケット廃棄がバースト的に発生するため、逆にパケット廃棄が発生しない時間も長く続くので、小さいドキュメントの転送がパケット廃棄が起こることなく終了することが多いためであると考えられる。このことは図 12 に示す、ルータバッファにおけるパケット廃棄の発生を示したグラフからも明らかである。逆に、ドキュメントサイズが大きくなると、RED 方式の公平性が TD 方式を上回っている。これは、ドキュメントサイズが大きくなると、TD 方式、RED 方式双方においてパケット廃棄が発生するため、バースト的なパケット廃棄を行なうため TCP のタイムアウトを起こしやすい TD 方式よりも、一度に 1 個ずつパケットを廃棄して Fast Retransmit 機構による再送を発生させやすくする RED 方式が有効であるためである。

さらに、コネクション数の変化が公平性に与える影響に着目すると、コネクション数 N が 10 本、100 本の場合では RED 方式が TD 方式の公平性を上回るが、コネクション数が

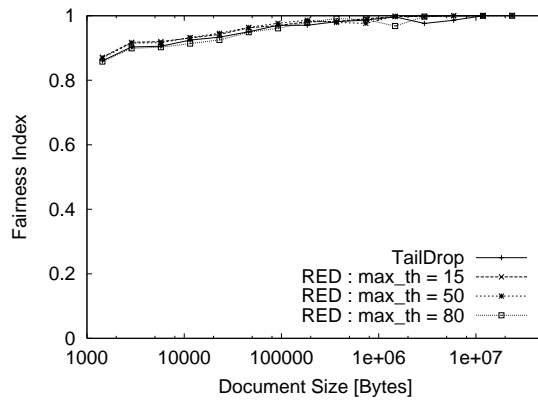
1000 本の場合では RED 方式の公平性が劣化している。これは、コネクション数が非常に多く、バッファサイズがコネクション数に対して絶対的に不足している場合、RED 方式のパラメータ max_{th} を調節することのみでは公平性を向上させることができず、バッファが一杯になる前にパケット廃棄を行なう RED 方式よりも、バッファサイズをすべて使い切ることが可能な TD 方式の方がパケット損失率が小さくなるためである。また、図 11 からバッファサイズが大きくなると、TCP コネクション数が 1000 本の場合に RED 方式の公平性が大きく改善していることがわかる。これは、バッファサイズが十分大きいと、RED 方式の確率的なパケット廃棄によるキュー長の制御が有効に機能しているためである。

また、Web ドキュメント転送の場合においても、無限長ファイル転送の場合と同様にパラメータ調整を行なうことで RED 方式での公平性を改善することができる。例えば図 9(b)、10(b)、11(b) では max_{th} の値をある程度大きくすることで TD 方式よりも高い公平性を実現している。しかし、図 9(c)、10(c) などからわかるように、どのような環境下でも、常に RED 方式で高い公平性を実現できるような 1 組のパラメータセットは存在しないことは明らかである。

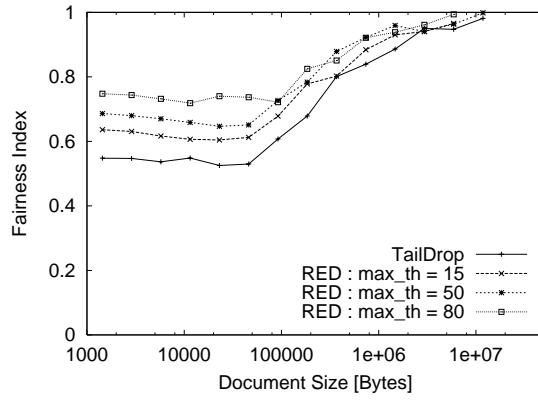
3.6 考察

3.4 節、3.5 節において、無限長ファイルの転送の場合と、Web ドキュメント転送の場合における TD 方式と RED 方式の公平性をそれぞれ評価した。その結果、文献 [7] において推奨されている RED 方式のパラメータを用いると、特にコネクション数が大きい場合に性能が非常に劣化することがわかった。また、パラメータ max_{th} をルータのバッファサイズに応じて変化させることによって、RED 方式の性能を大きく向上させることが可能であるが、適切なパラメータはネットワーク環境によって大きく変化することも明らかとなった。

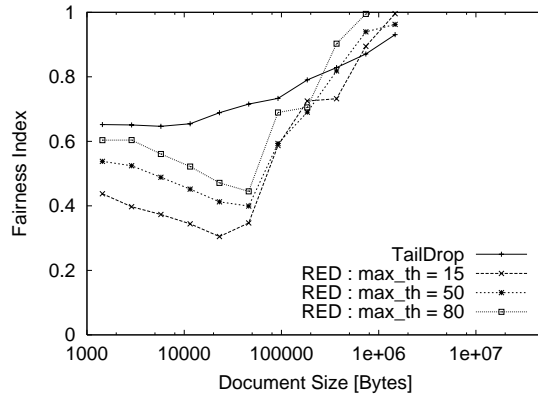
したがって、RED 方式のパラメータセットを 1 種類に固定し、アクティブなコネクション数、リンク帯域、バッファサイズ等が変化しても良好な性能を発揮させることは非常に困難であると言える。そこで、RED 方式のパラメータをネットワーク環境において動的に変化させることが有効であると考えられる。次章では、そのような方式の 1 つとして、RED 方式のパラメータのうち主にパケット廃棄のしきい値である min_{th} 及び max_{th} を、ルータ



(a) TCP Connection 10 本

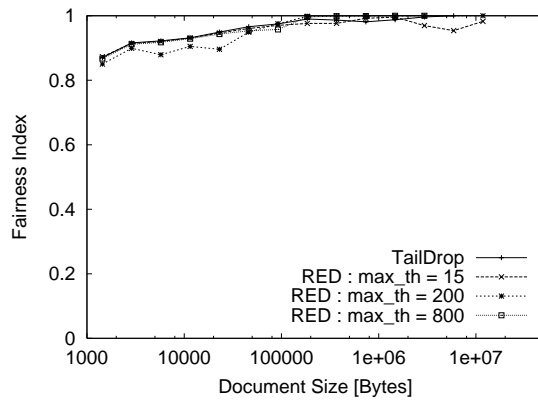


(b) TCP Connection 100 本

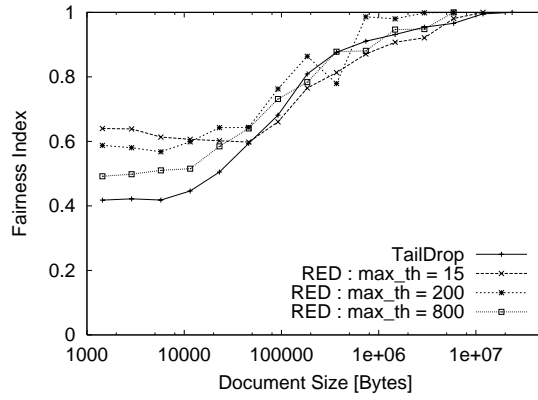


(c) TCP Connection 1000 本

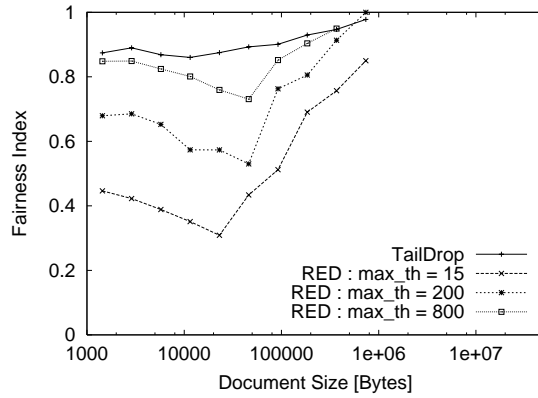
図 9: TD, RED 方式の公平性比較 (Web ドキュメント転送, $BW = 1.5$ [Mbps], $\tau = 4$ [msec], $B = 100$ [packets])



(a) TCP Connection 10 本

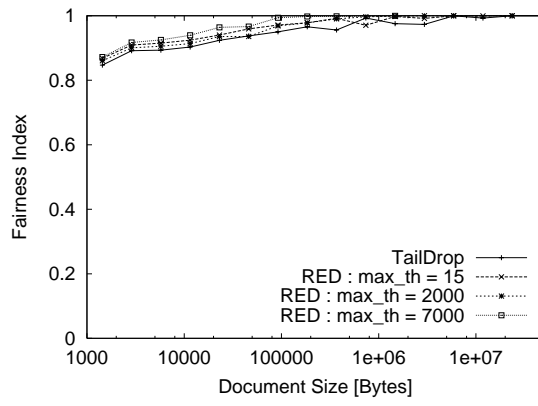


(b) TCP Connection 100 本

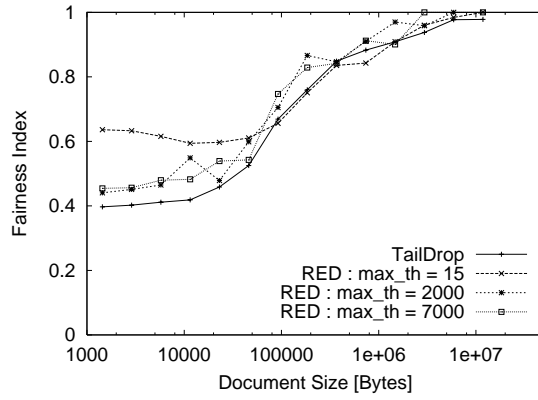


(c) TCP Connection 1000 本

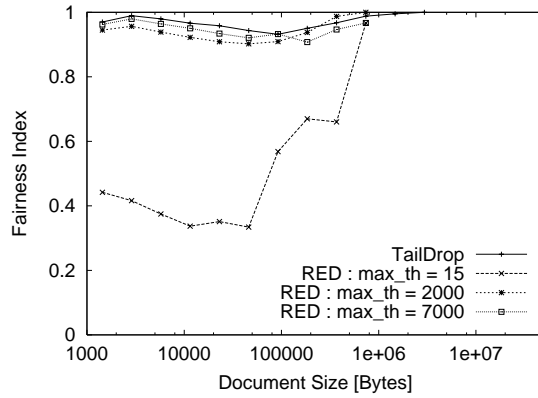
図 10: TD, RED 方式の公平性比較 (Web ドキュメント転送, $BW = 1.5$ [Mbps], $\tau = 4$ [msec], $B = 1000$ [packets])



(a) TCP Connection 10 本



(b) TCP Connection 100 本



(c) TCP Connection 1000 本

図 11: TD, RED 方式の公平性比較 (Web ドキュメント転送, $BW = 1.5$ [Mbps], $\tau = 4$ [msec], $B = 10000$ [packets])

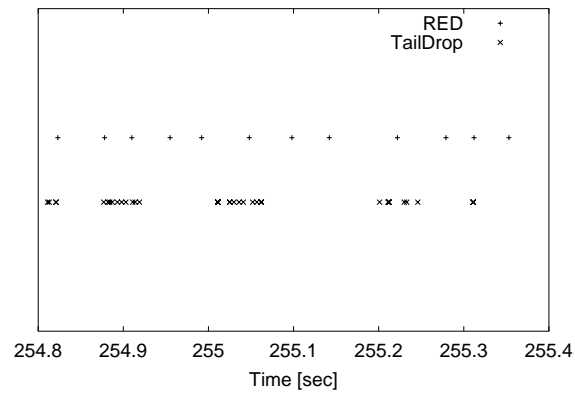


図 12: TD 方式と RED 方式の packet 廃棄間隔

バッファのキュー長に応じて動的に変化させる方式を提案する。

4 REDの動的パラメータ設定方式

3章の結果から、RED方式が効率的かつ公平に動作するには制御パラメータの調整が必要であることがわかった。REDの制御パラメータの設定に関してはこれまでも研究が行われている [7, 12-14]。しかし、これらの研究で提案されている制御パラメータは特定のネットワーク環境においては有効であるものの、異なった環境では必ずしも有効ではない。このことは、3章の結果からも明らかである。つまり、ルータに収容されるフロー数、バッファ内パケット数(キュー長)や輻輳状況といったネットワーク環境が大きく変化することが考えられるバックボーンルータを考えた場合、常に高い性能を示すようなRED方式の制御パラメータ群の設定を静的に行うことが困難であることは明らかである。そこで、本章ではキュー長の変動を基に動的にしきい値を制御するような方式である、dt-RED (RED with dynamic threshold control) 方式を提案する。

4.1 関連研究

RED方式のパラメータを動的に設定するための方式は文献 [15-17] 等でも提案されている。例えば、文献 [15] では、バースト的なコネクションが不利になりにくいというREDの特徴を有効に利用し、かつ平均キュー長が max_{th} に近付いた場合に発生するバースト的なパケット廃棄を防ぐため、RED方式の制御 w_q を平均キュー長 \bar{q} と実際のキュー長 q の差の大きさにしたがって変化させる、refined RED方式を提案している。また、平均キュー長の値にしたがって段階的にパケット廃棄率の最大値を決定するパラメータ max_p を増加させ、平均キュー長の不要な増大を抑えている。これによって、平均キュー長をしきい値 max_{th} と min_{th} の間で安定させ、バースト的なパケット廃棄の発生を抑えることを実現している。文献 [16] では、平均キュー長 \bar{q} が max_{th} を越えるとパケット廃棄率 max_p を大きく、逆に min_{th} を下回るとパケット廃棄率 max_p を小さくすることにより、到着パケットレートに応じた適切なパケット廃棄率を設定してキュー長の振動を抑え、低いパケット損失率と高いスループットを実現している。文献 [17] で提案されているBLUE方式は、パケット廃棄の発生とリンク利用率を監視して、パケット廃棄率を制御している。具体的には、バッファ溢れに

よるパケット廃棄が連続して発生するか、実際のキュー長があるしきい値を越えた場合に、パケット廃棄率 max_p を大きくする。またキュー長が0になるなどによって、リンク利用率が低下するとパケット廃棄率 max_p を小さくする。これにより、ネットワークの負荷が高い場合にはバースト的なパケット廃棄の発生を抑え、負荷が低いときにはリンク利用率の低下を抑えることを実現している。

これらの方式では、ネットワークの輻輳状況に応じて、主にパケット廃棄率を動的に制御している。しかし、TCP アプリケーションのスループットはパケット損失に対して敏感であり [18]、パケット廃棄率を急激に変化させることによってスループットやリンク利用率の低下といった問題を引き起こすことが考えられる。そこで本報告では、 max_p ではなく、しきい値 max_{th} 、 min_{th} の値を動的に制御する方式を提案する。

4.2 提案方式

4.2.1 設計方針

4.1 節で紹介した RED 方式の動的パラメータ設定方式は、主に RED 方式のパケット廃棄率を動的に制御するものであった。しかし、TCP を使用したアプリケーションでは、パケット損失率の変化に対するスループットの変動が大きいため、パケット廃棄率を制御する方法では、スループットの低下やリンク利用率の低下を招く恐れがある。さらに、これらの提案方式では公平性については考慮されておらず、同様にパケット廃棄率の制御では対応しきれない程の多くのフロー数を収容することは考慮されていない。また今後、技術革新にともないメモリの高性能化、低価格化が進行し、ルータバッファのために用意されるメモリ容量も増加することが考えられる。しかし、これまでの RED 方式の動的パラメータ設定方式では、しきい値 (max_{th} , min_{th}) を動的に制御しないので、ルータにおけるアクティブフロー数の増減に応じてルータバッファを有効に活用することができない。そこで、本報告では、主に RED 方式の制御パラメータ max_{th} , min_{th} を平均キュー長 \bar{q} や実際のキュー長 q の変化にしたがって制御し、最大パケット廃棄率 max_p の制御は副次的な手段とする、dt-RED (RED with dynamic threshold control) 方式を提案する。

3章において示したように、RED方式においては、平均キュー長が頻繁にしきい値 max_{th} に達するような輻輳状態においては、十分な性能を発揮できない。従って、dt-RED方式では、平均キュー長 \bar{q} を監視し、それが max_{th} に接近した場合に、バッファに余裕があれば max_{th} を大きくする。これにより、平均キュー長が max_{th} に達することによって発生するバースト的なパケット廃棄を防止する。また逆に、平均キュー長が max_{th} に比べて十分小さい場合には、 max_{th} を段階的に小さくする。これは、 max_{th} を小さくすることによって、全体のパケット廃棄率を大きくし、平均キュー長を短くすることが目的である。この方式によって、dt-RED方式においても間接的にパケット廃棄率が変動する。しかし、 max_p を直接調節する他の方式に比べ、パケット廃棄率の変動は緩やかであるため、TCPアプリケーションのスループットに与える影響は小さいと考えられる。

また、実際のキュー長が0になり、ルータの利用率が低下する場合には、しきい値 min_{th} を一時的に大きくすることによって廃棄するパケット数を減少させ、ルータの利用率を向上させる。その後、ルータへのパケット到着レートが増加し、実際のキュー長が0より大きくなれば、 min_{th} を再び大きくする。さらに、しきい値 max_{th} , min_{th} の動的な設定によって平均キュー長の制御を行うことができない場合、つまり、 max_{th} をバッファサイズ付近まで大きくしても平均キュー長が max_{th} に達する場合や、 min_{th} を大きくしてもルータの利用率の低下を防止できない場合には、RED方式のパケット廃棄率を決定する max_p を変化させる。つまり、平均キュー長の増加を防止できない場合には max_p を大きく、ルータの利用率の低下を防止できない場合には max_p を小さく変更する。

具体的なアルゴリズムについては次節で述べる。

4.3 アルゴリズム

dt-RED方式は、平均キュー長を監視することによってしきい値の制御を行うが、パケット到着毎にキュー長 \bar{q} の状態を調べて、動的にパラメータを制御すると、各パラメータの変動が不安定になるため、本方式においては、パケットが N 個到着する期間中(以下、観測期間と呼ぶ)のキュー長の状態を調べ、動的なパラメータの制御を行う。以下、順に max_{th} , min_{th} , max_p の動的制御アルゴリズムを示す。

- max_{th} の制御

観測期間内に平均キュー長 \bar{q} が $X_{max} \cdot max_{th}$ に 1 回以上達した場合、 max_{th} の値を式 (5) によって大きくする。

$$max_{th} \leftarrow max_{th} + \alpha_{max} \cdot B \quad (5)$$

また、観測期間中常に $(\bar{q} - min_{th}) < Y_{max} \cdot (max_{th} - min_{th})$ であった場合、 max_{th} の値を式 (6) によって小さくする。

$$max_{th} \leftarrow max_{th} - \beta_{max} \cdot (max_{th} - min_{th}) \quad (6)$$

ここで、 α_{max} , β_{max} はそれぞれ、 max_{th} の増加率と減少率とを決定する、dt-RED 方式の制御パラメータである。また、 X_{max} は max_{th} の値が現在の平均キュー長に対して小さいことを判断するためのしきい値であり、 Y_{max} は max_{th} の値が現在の平均キュー長に対して大きすぎることを判断するためのしきい値である。

- min_{th} の制御

観測期間内に実際のキュー長が 1 回以上 0 になる場合、リンク利用率が低下していると判断し、 min_{th} の値を式 (7) によって大きくする。

$$min_{th} \leftarrow min_{th} + \alpha_{min} \cdot (max_{th} - min_{th}) \quad (7)$$

逆に、実際のキュー長が 1 回も 0 にならない、かつ平均キュー長 \bar{q} が min_{th} よりも大きい場合には、リンク利用率が十分であると判断し、輻輳を回避するために min_{th} の値を式 (8) で更新する。

$$min_{th} \leftarrow min_{th} - \beta_{min} \cdot (max_{th} - min_{th}) \quad (8)$$

ここで、 α_{min} , β_{min} はそれぞれ、 min_{th} の増加率、減少率を決定する、dt-RED 方式の制御パラメータである。

- max_p の制御

上述の max_{th} 、 min_{th} の制御ではリンク利用率の低下やバースト的なパケット廃棄を避けられない場合に制御を行なう。まず、観測期間内に平均キュー長 \bar{q} が 1 回以上 max_{th} に達

すると同時に $max_{th} \geq B \cdot X_{buf}$ となっている場合、 max_{th} を大きくすることで平均キュー長を安定化させることが望めないため、 max_p の値を式 (9) によって大きくする。

$$max_p \leftarrow \alpha_p \cdot max_p \quad (9)$$

また、式 (7) によって min_{th} が増加し続け、観測期間中常に $min_{th} \geq \frac{1}{2}B$ である場合、パケット廃棄率が高すぎると判断し、リンク利用率を上げるために、 max_p の値を式 (10) によって小さくする。

$$max_p \leftarrow \beta_p \cdot max_p \quad (10)$$

max_{th}, min_{th} の場合と同様に、 α_p, β_p は max_p の増加率、および減少率を決定する dt-RED 方式のコントロールパラメータであり、 X_{buf} は平均キュー長の増大を抑えられず、 max_p を大きくする必要があることを判断するためのしきい値である。

dt-RED 方式は、 min_{th} 、 max_{th} 、 max_p を動的に制御するために 8 つの制御パラメータを導入している。しかし、RED 方式と違い、これらのパラメータの値は dt-RED 方式の性能に与える影響は小さい。つまり、一組のパラメータセットを用いることで、さまざまなネットワーク環境に対応することができる。このことは 5 章において、シミュレーション結果を通じて明らかにする。

パラメータ	値
max_{th} の初期値	$\frac{1}{2}B$ [packets]
min_{th} の初期値	5 [packets]
観測期間 N	100 [packets]
α_{max}	0.05
α_{min}	0.05
β_{max}	0.05
β_{min}	0.05
X_{max}	0.9
Y_{max}	0.8
X_{buf}	0.8

表 2: dt-RED 方式の制御パラメータ

5 提案方式の性能評価

本章では本報告で提案する dt-RED (RED with dynamic threshold control) 方式を、従来の TD 方式、及び既存の RED 方式と公平性の面から比較して性能を評価する。シミュレーションで用いるネットワークポロジは 3.2 節で示したものを、ボトルネックルータに dt-RED 方式を用いる。dt-RED 方式のパラメータには表 2 の値を用いる。また、 max_p 、 w_q の値は表 1 に示した RED 方式のものを用いる。またここでは 4.3 節で説明したアルゴリズムのうち、 max_p のコントロールは用いず、 max_{th} 、 min_{th} の制御のみを行なっている。

5.1 無限長ファイルの転送の場合

本節では、3.4 節で得られた結果と対応する結果として、各送信側端末が無限長のファイル転送を行なう場合における公平性の評価を行なう。公平性の指標には 3.4 節と同様に、N 台の送信側端末が受信側端末に 1000 秒間のデータ転送を行ない、100 秒毎のスループットをサンプル値 x_i として式 (4) で計算した Fairness Index 値を用いる。

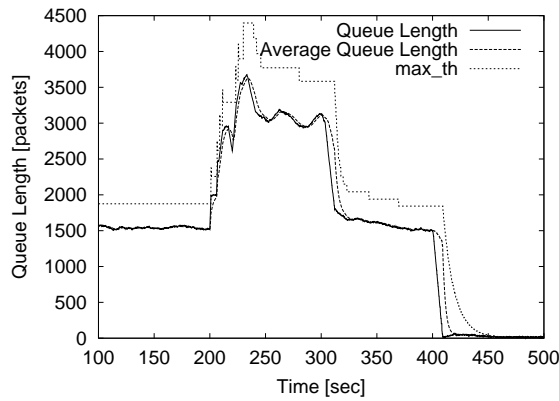


図 13: TCP コネクション数が動的な環境での dt-RED 方式の動作

図 13 は dt-RED 方式が、ネットワーク環境の動的な変化に合わせて max_{th} の値を変化させている様子を示したものである。ここでは $BW = 1.5$ [Mbps]、 $\tau = 4$ [msec]、 $B = 10000$ [packets] とし、TCP コネクション数 N をシミュレーション開始から 200 [sec] までは 500、200 [sec]–300 [sec] は 1000、300 [sec]–400 [sec] は 500、400 [sec]–500 [sec] は 10 に変化させている。図からわかるように、平均キュー長がコネクション数の増減にしたがって変化するのに対応して、 max_{th} が変化していることがわかる。このような動作をすることで dt-RED 方式ではバースト的なパケット廃棄を極力避け、高い公平性を実現する。

図 14 は、 $BW = 1.5$ [Mbps]、 $\tau = 4$ [msec]、 $B = 100$ [packets] (図 14(a))、1000 [packets] (図 14(b))、及び 10000 [packets] (図 14(c)) とした場合の、コネクション数 N に対する Fairness Index 値の変化を示したものである。ここでは、dt-RED 方式の結果とともに、TD 方式、文献 [7] で推奨されているパラメータの値を用いた RED 方式、及び 3 章で得られた最適なパラメータ調整を行なった RED 方式の結果を示している。図から、dt-RED 方式はさまざまなネットワーク環境において、最適なパラメータ調整を行なった RED 方式と同等か、あるいはそれ以上の性能を示している。

図 15 は $BW = 1.5$ [Mbps]、 $\tau = 400$ [msec]、図 16 は $BW = 150$ [Mbps]、 $\tau = 4$ [msec] の場合の結果を示している。この場合も図 14 と同様、RED 方式の最適パラメータ調整を行なった場合と同じか、それを上回る公平性を実現している。さらに、最適なパラメータ調整を行なった RED 方式の公平性が TD 方式に比べて劣化している場合でも、公平性を劣化

させていない(図 15(b))。これはコネクション数が異なる環境において max_{th} を固定して RED 方式を用いると、コネクション数によってはその max_{th} が最適でない場合があるのに対し、dt-RED 方式は動的にパラメータ調整を行なうので、コネクション数に応じて常に各パラメータを最適な値にすることができるからである。しかし、図 15(c) では dt-RED 方式を用いても公平性が改善していない。これは、dt-RED 方式は RED 方式と同様にパケット廃棄を意図的に発生させるため、TD 方式においてパケット廃棄が発生しないようなネットワーク環境においては、TD 方式よりも性能が劣化することは避けられないことを示している。このような状況における dt-RED 方式の改善方式については今後の課題としたい。

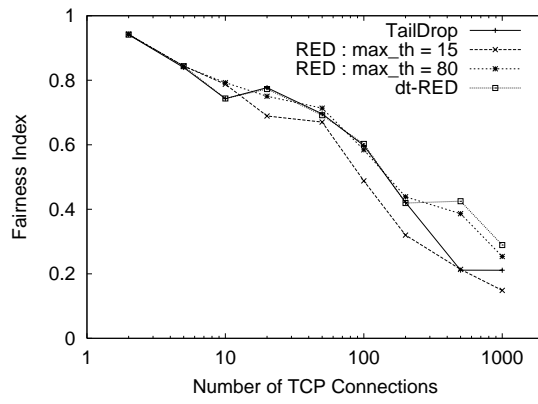
ここまでの結果から、dt-RED 方式は TD 方式、および既存の RED 方式に対して同程度か、それを上回る公平性を実現していることが明らかになった。さらに、RED 方式は個々ネットワーク環境においてパラメータ調整が必要であったが、dt-RED 方式では表 2 に示す 1 つのパラメータセットで優れた公平性を実現している。このことから、dt-RED 方式によって、RED 方式の持つ欠点であるパラメータの設定に対して敏感であるという点が解消されていると言える。

5.2 Web トラフィックを考慮したドキュメント転送の場合

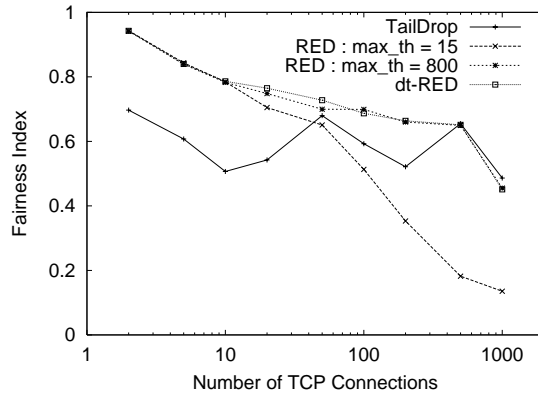
次に、3.5 節で得られた結果に対応する結果として、実際の Web トラフィックを考慮したドキュメント転送を行なった際の dt-RED 方式の性能を評価する。

図 17-19 は $BW = 1.5$ [Mbps]、 $\tau = 4$ [msec]、 $B = 100-10000$ [packets]、及びコネクション数 10、100、1000 本としたときのドキュメントサイズと Fairness Index 値の関係を TD 方式、[7] で推奨されているパラメータの値を用いた RED 方式、3 章で得られた最適なパラメータチューニングを行なった RED 方式、及び dt-RED 方式について示したものである。図から、無限長ファイル転送の場合と同様、dt-RED 方式が最適なパラメータチューニングを行なった RED 方式とほぼ同じか、それを上回る公平性を常に実現していることがわかる。例えば、図 18(b)、19(b) では、特に転送ドキュメントサイズが小さい場合に、最適なチューニングを施した RED 方式の公平性を上回る公平性を実現している。

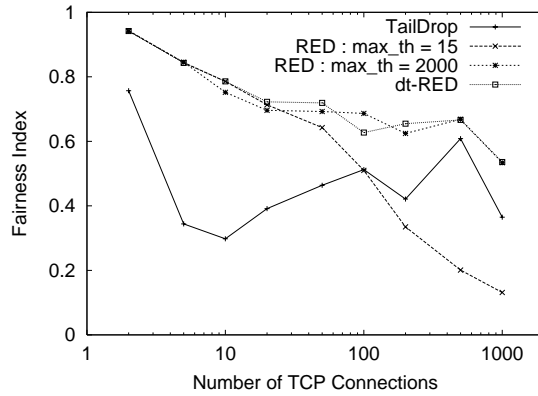
また、図 17(c) を見ると、dt-RED 方式が TD 方式よりも低い公平性を示しているが、こ



(a) $B = 100$ [packets]

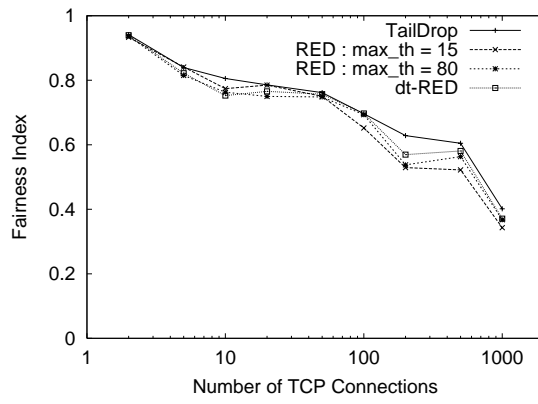


(b) $B = 1000$ [packets]

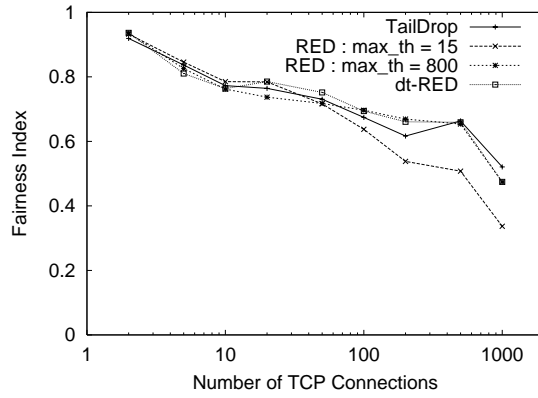


(c) $B = 10000$ [packets]

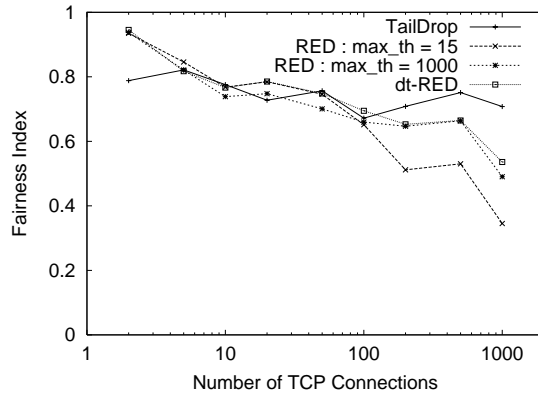
図 14: TD, RED, dt-RED 方式の公平性比較 (無限長ファイル転送, $BW = 1.5$ [Mbps], $\tau = 4$ [msec])



(a) $B = 100$ [packets]

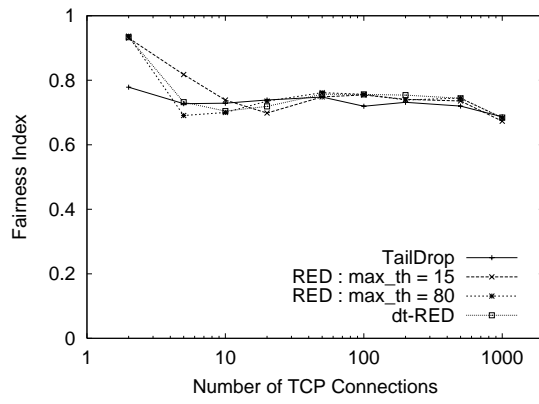


(b) $B = 1000$ [packets]

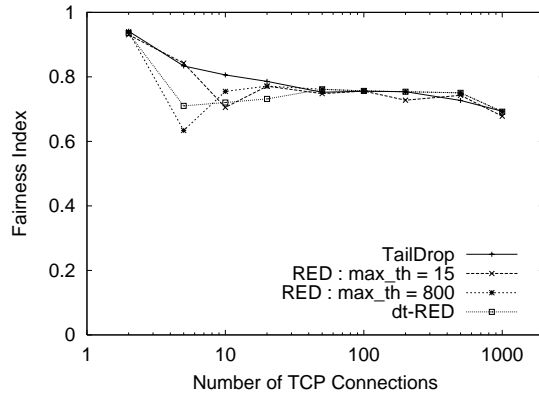


(c) $B = 10000$ [packets]

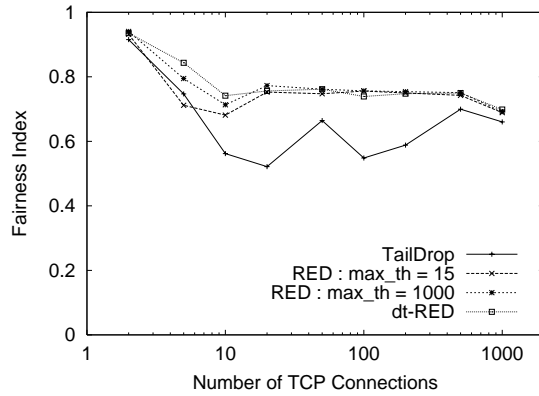
図 15: TD, RED, dt-RED 方式の公平性比較 (無限長ファイル転送, $BW = 1.5$ [Mbps], $\tau = 400$ [msec])



(a) $B = 100$ [packets]



(b) $B = 1000$ [packets]



(c) $B = 10000$ [packets]

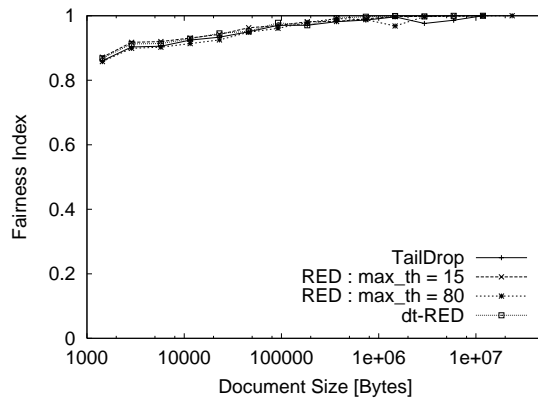
図 16: TD, RED, dt-RED 方式の公平性比較 (無限長ファイル転送, $BW = 150$ [Mbps], $\tau = 4$ [msec])

これは、コネクション数が1000本に対してバッファサイズが100 [packets] と絶対的に不足しているため、dt-RED方式によるパラメータ制御を行ってもキュー長が安定せず、TD方式よりもバースト的なパケット廃棄が多く発生していることが原因である。このような場合は max_{th} 、 min_{th} の制御に加えて、パケット廃棄確率である max_p の制御が重要となるが、それは今後の課題としたい。さらに、図19(c)において、TD方式よりもdt-RED方式の方が公平性が若干劣化しているが、これは図16(c)と同様、バッファサイズが十分に大きく、TD方式でパケット損失が発生しない場合であるため、強制的にパケットを廃棄するRED方式をベースとしたdt-RED方式では公平性が劣化するためである。

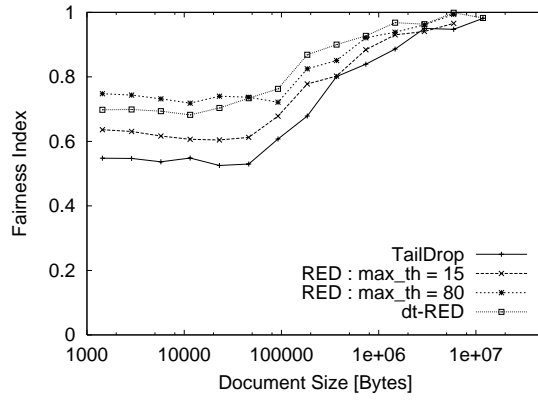
5.3 考察

5.1節、5.2節において、無限長ファイル転送の場合とWebトラフィックを考慮したドキュメント転送の場合について、dt-RED方式の性能を評価した。その結果、dt-RED方式ではRED方式のパラメータをネットワークの状況に合わせて設定した場合と同程度か、それを上回る公平性を実現できることが明らかとなった。また、dt-RED方式では一組のパラメータセットを用いて、異なった環境でそれぞれ最適なパラメータチューニングを行なったRED方式を上回る公平性を実現できることを示した。これによって、dt-RED方式を用いることでRED方式の欠点であるパラメータ設定の困難さを解消できることがわかった。

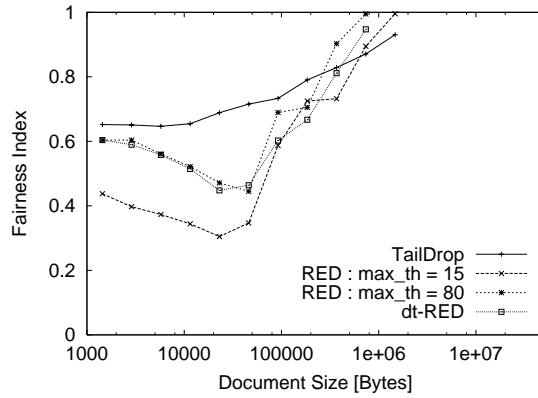
dt-RED方式はRED方式を基に、そのパラメータを動的に制御する方式であるため、最適なパラメータチューニングを施したRED方式に対して、著しい公平性の改善を示すわけではない。しかし、RED方式ではネットワーク環境の変化によって制御パラメータをその都度慎重に設定しなければ十分な性能を発揮できないのに対して、dt-RED方式では制御パラメータを、一種類に固定したままでRED方式における最良の結果と同程度の結果を実現できる。このことは、コネクション数、輻輳状況などが大きく変化する、実際のネットワーク上のルータに適用する上で非常に重要な意味を持つといえる。



(a) TCP Connection 10 本

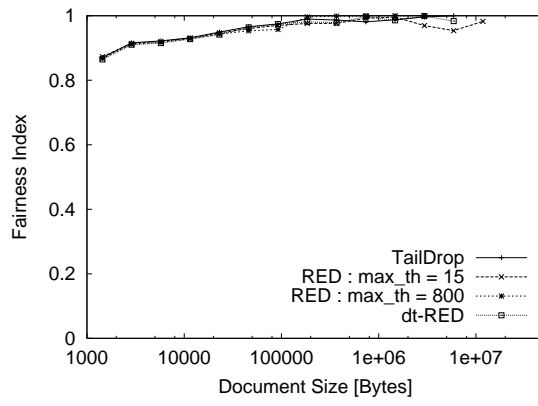


(b) TCP Connection 100 本

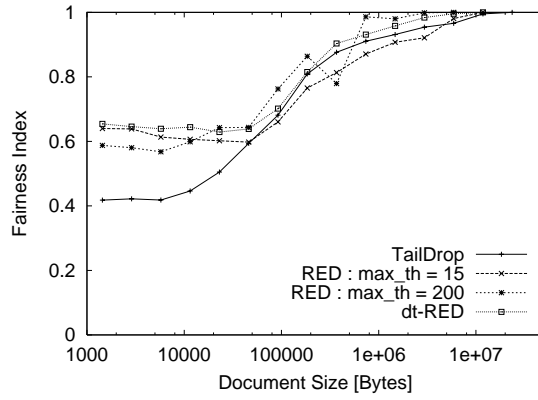


(c) TCP Connection 1000 本

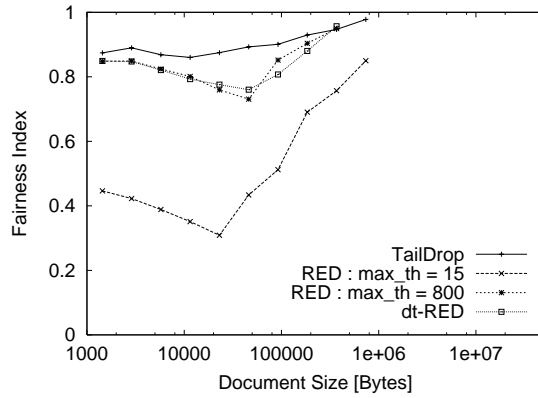
図 17: TD, RED, dt-RED 方式の公平性比較 (Web ドキュメント転送, $BW = 1.5$ [Mbps], $\tau = 4$ [msec], $B = 100$ [packets])



(a) TCP Connection 10 本

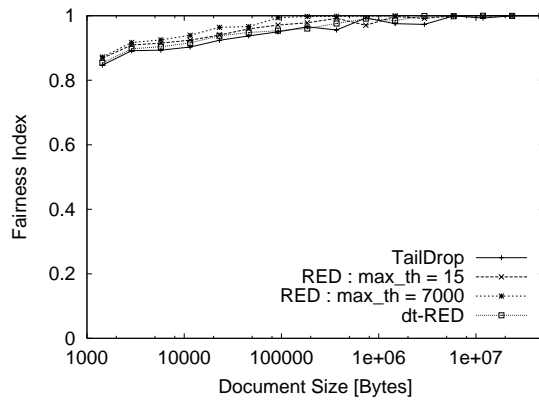


(b) TCP Connection 100 本

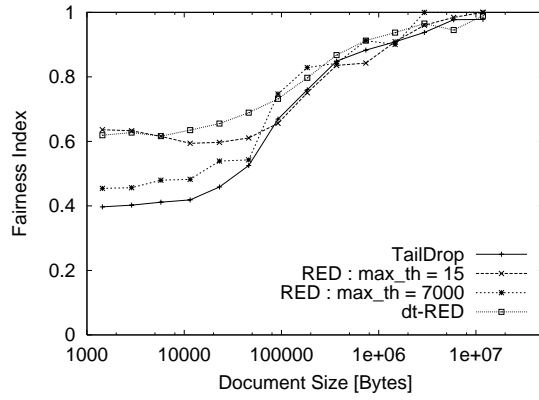


(c) TCP Connection 1000 本

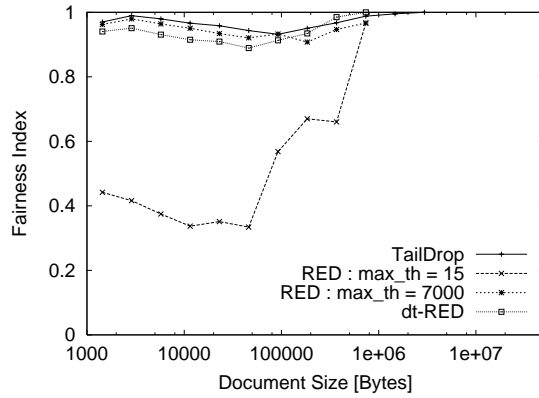
図 18: TD, RED, dt-RED 方式の公平性比較 (Web ドキュメント転送, $BW = 1.5$ [Mbps], $\tau = 4$ [msec], $B = 1000$ [packets])



(a) TCP Connection 10 本



(b) TCP Connection 100 本



(c) TCP Connection 1000 本

図 19: TD, RED, dt-RED 方式の公平性比較 (Web ドキュメント転送, $BW = 1.5$ [Mbps], $\tau = 4$ [msec], $B = 10000$ [packets])

6 おわりに

本報告では、特にフロー数が非常に多い環境での TCP コネクション間の公平性に着目した RED (Random Early Detection) 方式の性能評価、及び公平性を改善するための RED 方式の動的パラメータ設定方式の提案と評価を行った。まず、TCP コネクションを用いて無限長ファイル転送、及び Web のトラフィック分布を考慮したドキュメント転送を行う場合におけるコネクション間の公平性に着目して RED 方式と TD 方式との性能比較を行った。その結果、RED 方式は TD 方式と比較して高い公平性を実現することが可能であるが、RED 方式がその性能を十分に発揮するためには、ルータに収容されるコネクション数などのネットワークの環境に応じてパラメータの設定を慎重に行う必要があり、その設定が適切でない場合には、RED 方式の性能は TD 方式よりも劣化することが明らかとなった。また、RED 方式が十分な性能を発揮するパラメータはフロー数、バッファ内パケット数 (キュー長) といったネットワークの環境によって大きく変化するため、静的なパラメータ設定で常に高い公平性を実現することは困難であることも明らかとなった。

以上の結果から、本報告ではさらに RED 方式のパラメータ設定を動的に行う dt-RED (RED with dynamic threshold control) 方式を提案し、性能を評価した。その結果、従来の TD 方式および、既存の RED 方式よりも高い公平性を示すことが明らかとなった。

今後の課題として、提案方式に加えてさらに最大パケット廃棄確率 max_p の制御を行った際の性能や、TCP と UDP が混在する環境での提案方式の評価を行う必要がある。また、文献 [28-30] などで提案されている RED 方式のアルゴリズム改善案や、その他の動的パラメータ設定方式と提案方式との比較評価を行う必要がある。

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村田正幸教授に深く感謝致します。また、本報告において熱心に直接御指導頂いた長谷川剛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、大阪府立看護大学医療技術短期大学の菅野正嗣助教授、サイバーメディアセンターの馬場健一助教授、宮原研究室の若宮直紀講師、サイバーメディアセンターの大崎博之助手、大阪市立大学の阿多信吾助手、経済学部の荒川伸一助手に心から感謝致します。最後に、御協力頂いた村田研究室および宮原研究室の皆様にお礼申し上げます。

参考文献

- [1] W. R. Stevens, *TCP/IP Illustrated, Volume 1: The Protocols*. Reading, Massachusetts: Addison-Wesley, 1994.
- [2] Robert Morris, “TCP behavior with many flows,” in *Proceedings of IEEE*, October 1997.
- [3] L. S. Brakmo, S. W.O’Malley, and L. L. Peterson, “TCP Vegas: New techniques for congestion detection and avoidance,” in *Proceedings of ACM SIGCOMM’94*, pp. 24–35, October 1994.
- [4] S. Floyd and T. Henderson, “The NewReno modification to TCP’s fast recovery algorithm,” *Request for Comments 2582*, April 1999.
- [5] M. Mathis, J. Mahdavi, S. Floyd, and A. Romanow, “TCP selective acknowledgement options,” *Request for Comments 2018*, October 1996.
- [6] Go Hasegawa and Kenji Kurata and Masayuki Murata, “TCP Vegas+ for fairness improvement between TCP Reno and TCP Vegas,” *Technical Report of IEICE (SSE2000-31)* (in Japanese), pp. 1–6, May 2000.
- [7] S. Floyd and V. Jacobson, “Random early detection gateways for congestion avoidance,” *IEEE/ACM Transactions on Networking*, vol. 1, pp. 397–413, August 1993.
- [8] Carlos M.Pazos, Juan C.Sanchez Agrelo, Mrio Gerla, “Using back-pressure to improve TCP performance with many flows,” in *Proceedings of IEEE INFOCOM’99*, March 1999.
- [9] Thomas Bonald, Martin May Jean-Chrysostome Bolot, “Analytic evaluation of RED performance,” in *Proceedings of IEEE INFOCOM 2000*, March 2000.
- [10] D. Lin and R. Morris, “Dynamics of random early detection,” in *Proceedings of SIGCOMM ’97*, 1997.

- [11] Yin Zhang, Lili Qiu, “Understanding the end-to-end performance impact of RED in a heterogeneous environment,” Tech. Rep. Cornell CS Technical Report 2000-1802, July 2000.
- [12] Martin May, Jean Bolot, Christophe Diot, and Bryan Lyles, “Reasons not to deploy RED,” in *Proceedings of IWQoS’99*, June 1999.
- [13] Mikkel Christiansen, Kevin Jeffay, David Ott, F. Donelson Smith, “Tuning RED for web traffic,” in *Proceedings of ACM SIGCOMM 2000*, August 2000.
- [14] S. Floyd, “RED: Discussions of setting parameters,” available at <http://www.aciri.org/floyd/REDparameters.txt>, 1997 November.
- [15] Haining Wang and Kang G. Shin, “Refined design of random early detection gateways,” in *Proceedings of Globecom’99*, pp. 769–775, December 1999.
- [16] Wu-chang Feng, Dilip D. Kandlur, Debanjan Saha, Kang G. Shin, “A self-configuring RED gateway,” in *Proceedings of IEEE INFOCOM’99*, March 1999.
- [17] Wu-chang Feng, Dilip D. Kandlur, Debanjan Saha, Kang G. Shin, “BLUE: A new class of active queue management algorithms,” Tech. Rep. CSE-TR-387-99, U. Michigan, March 1999.
- [18] J. Padhye, V. Firoiu, D. Towsley, and J. Kurose, “Modeling TCP throughput: a simple model and its empirical validation,” in *Proceedings of ACM SIGCOMM’98*, pp. 303–314, August 1998.
- [19] M. Shreedhar and G. Varghese, “Efficient fair queuing using deficit round robin,” *IEEE/ACM Transactions on Networking*, vol. 4, pp. 375–385, June 1996.
- [20] B. Suter, T.V. Lakshmanan, D. Stiliadis, and A.K. Choudhary, “Design considerations for supporting TCP with per-flow queueing,” in *Proceedings of IEEE INFOCOM ’98*, April 1998.

- [21] M. Parris, K. Jaffay, and F. Fonelson Smith, "Lightweight active router-queue management for multimedia networking," in *Proceedings of SPIE: Multimedia Computing and Networking 1999*, pp. 162–174, January 1999.
- [22] E. Hashem, "Analysis of random drop for gateway congestion control," Tech. Rep. MIT-LCS-TR-465, Laboratory for Computer Science, MIT, Cambridge, MA, 1989.
- [23] L. Zhang, "A new architecture for packet switching network protocols," Tech. Rep. MIT-LCS-TR-455, Laboratory of Computer Science, MIT, August 1989.
- [24] V. Jacobson, "Congestion avoidance and control," in *Proceedings of SIGCOMM '88*, pp. 314–329, August 1988.
- [25] Network Simulator - ns (version 2). available from <http://www.isi.edu/nsnam/ns/>.
- [26] R. Jain, "Throughput fairness index: An explanation," *ATM Forum Contribution 99-0045*, February 1999.
- [27] Paul Barford, Mark Crovella, "Generateing representative web workloads for network and server performance evaluation," in *Proceedings of the 1998 ACM SIGMETRICS International Conference on Measurement and Modeling of Computer Systems*, pp. 151–160, July 1998.
- [28] S. Floyd, "Recommendation on using the "gentle_" variant of RED," available at <http://www.aciri.org/floyd/red/gentle.html>, March 2000.
- [29] W. Feng, D. Kandlur, D. Saha, K. Shin, "Techniques for eliminating packet loss in congested TCP/IP networks," Tech. Rep. CSE-TR-349-97, U. Michigan, November 1997.
- [30] D. D. Clark, and W. Fang, "Explicit allocation of best effort packet delivery service," *IEEE/ACM Transaction on Networking*, vol. 6, pp. 362–373, August 1998.