

## ECN を用いた TCP Vegas にもとづく ウィンドウ型フロー制御方式

大崎 博之

大阪大学 大学院基礎工学研究科 情報数理系専攻  
oosaki@ics.es.osaka-u.ac.jp  
www-miya.ics.es.osaka-u.ac.jp/~oosaki/

SSE 99-3

Hiroyuki Ohsaki

3

## 発表の内容

- ◆ 研究の背景
- ◆ TCP Vegas にもとづくウィンドウ型フロー制御
- ◆ 安定性解析
- ◆ シミュレーション
- ◆ ECN による問題点の改善
- ◆ まとめ

SSE 99-3

Hiroyuki Ohsaki

2

## TCP (Transmission Control Protocol)

- ◆ **パケット再送機構**
  - 棄却されたパケットを受信側ホストへ再送
- ◆ **輻輳回避機構**
  - ウィンドウサイズによって送出パケット数を調整
- ◆ **複数のバージョンの TCP が存在**
  - TCP Tahoe
  - TCP Reno
  - TCP Vegas

SSE 99-3

Hiroyuki Ohsaki

3

## TCP Reno

- ◆ BSD UNIX に実装され、現在も広く利用
- ◆ **パケット棄却の発生を利用**
  1. 送信側ホストは連続的にウィンドウサイズを増加
  2. ルータにおいてパケット廃棄が発生
  3. タイムアウト等によりパケット廃棄を検出
  4. ウィンドウサイズを減少
- ◆ **パケット棄却が起こるまで輻輳を検出できない**

SSE 99-3

Hiroyuki Ohsaki

4

## TCP Vegas

- ◆ TCP Reno の問題点を解消
  - 新しいタイムアウト機構
  - バッファ占有量を制御する輻輳回避機構
  - 改良されたスロースタート機構
- ◆ **パケットのラウンドトリップ時間を観測して利用**
  1. ACK を受信するまでの時間を測定
  2. バッファ内に蓄えられているパケット数を推測
  3. この値が一定となるようにウィンドウサイズを調整
- ◆ **パケット棄却を待たずに輻輳の検出が可能**

SSE 99-3

Hiroyuki Ohsaki

5

## TCP Vegas の輻輳回避機構

- ◆ 送信側ホストは往復伝播遅延時間  $\tau$  を観測
- ◆ ACK パケットにより RTT  $r(k)$  を測定

$$d(k) = \frac{w_n(k)}{\tau} - \frac{w_n(k)}{r(k)}$$

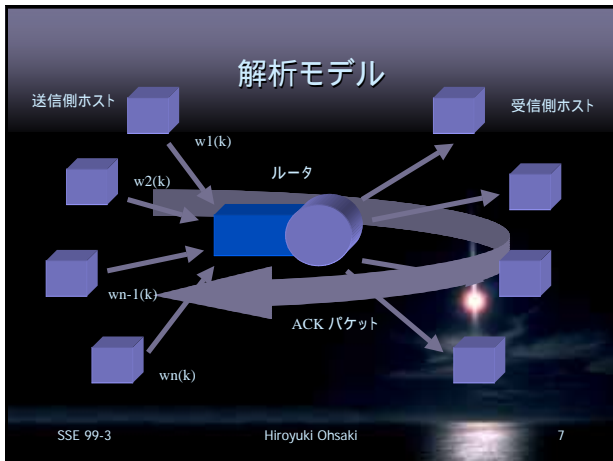
- ◆  $d(k)$  の値に応じてウィンドウサイズを変更

$$w_n(k+1) = \begin{cases} w_n(k)+1 & \text{if } d(k) < \alpha \\ w_n(k)-1 & \text{if } \beta < d(k) \\ w_n(k) & \text{otherwise} \end{cases}$$

SSE 99-3

Hiroyuki Ohsaki

6



### システムの状態方程式

- ◆ ウィンドウサイズの変化  
 $\delta$ : ウィンドウサイズの変化量を決める制御パラメータ  

$$w_n(k+1) = \max(w_n(k) + \delta(\gamma - d(k)), 0)$$
- ◆ ルータのバッファ内パケット数の変化  

$$q(k+1) = \min(\max(Nw(k) - Bw(k), 0), L)$$

SSE 99-3 Hiroyuki Ohsaki 8

### 安定性解析

- ◆ システムの平衡点を導出
- ◆ 平衡点の近傍で線形化
- ◆ 平衡点が局所漸近安定となる条件を導出

$$\frac{\delta}{(B + \gamma N)\tau} + 2 > 0$$

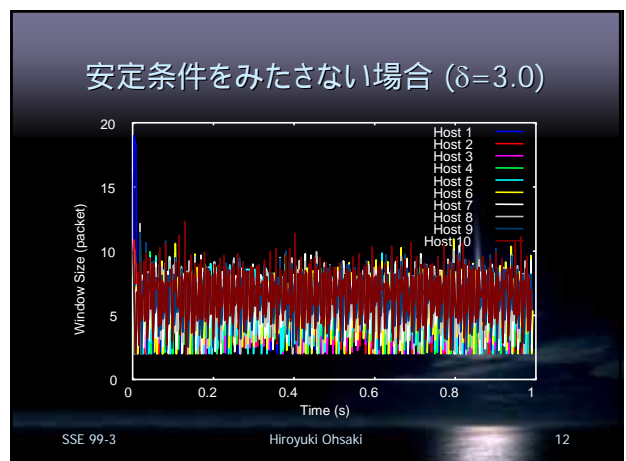
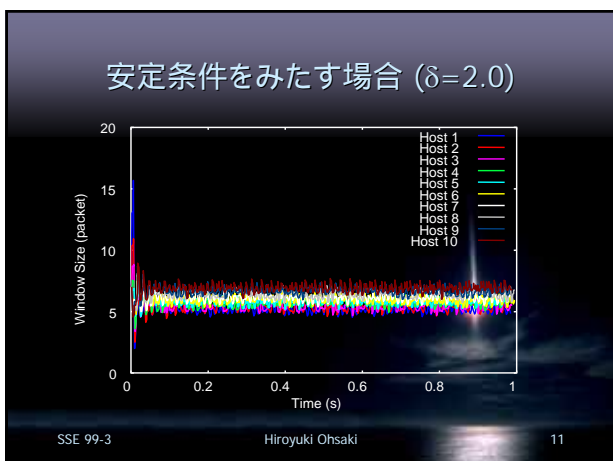
$$\frac{B\delta}{(B + \gamma N)} < 1$$

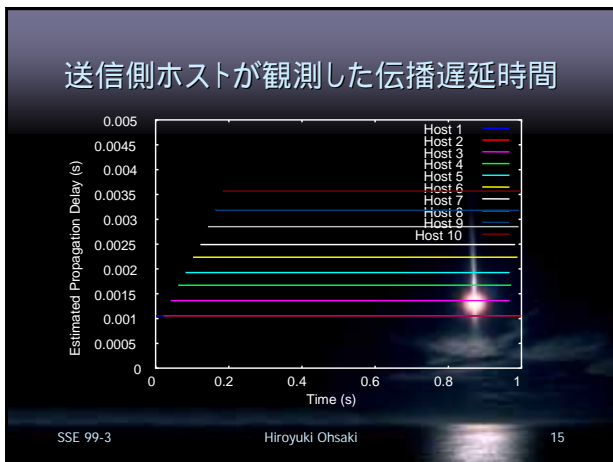
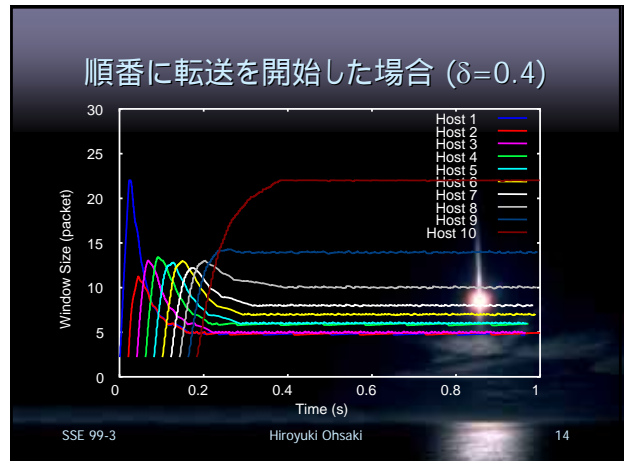
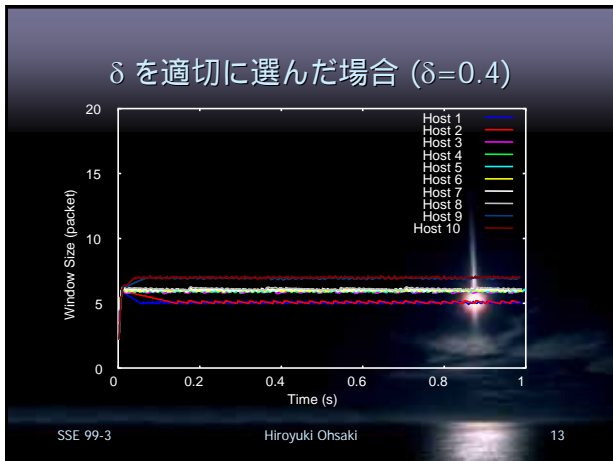
SSE 99-3 Hiroyuki Ohsaki 9

### シミュレーションパラメータ

- ◆ パケット長 1000バイト
- ◆ コネクション数 10本
- ◆ ボトルネック帯域 20パケット/ms (155Mbps)
- ◆ 伝播遅延時間 1ms
- ◆ 制御パラメータ  $\gamma$  3 パケット
  - コネクションあたりのバッファ占有量
- ◆ 制御パラメータ  $\delta$  0.4, 2.0, 3.0
  - 1 RTT あたりのウィンドウサイズ変化量
  - 0.4 (最適), 2.0 (安定), 3.0 (不安定)

SSE 99-3 Hiroyuki Ohsaki 10





### TCP Vegas の問題点

- ◆ 伝播遅延時間を正確に観測できない
  - 全てのルータがコネクションごとにバッファを持つ
  - 全てのルータのバッファが空
  - ウィンドウサイズは伝播遅延時間に比例
- ◆ コネクション数に関するスケーラビリティがない
  - バッファ内パケット数はコネクション数に比例

$$w^* = \tau \left( \frac{B + \gamma N}{N} \right) \quad q^* = \gamma N \tau$$

SSE 99-3 Hiroyuki Ohsaki 16

### ECNによる問題点の回避

- ◆ ECN (Explicit Congestion Notification)
  - ルータから送信側ホストへ輻輳の発生を通知
- ◆ 2種類の実装方法
  - ICMP Source Quench パケット
  - IP パケットの TOS (Type of Service) フィールド
- ◆ 送信側ホストが ECN を受信した場合
  - TCP Vegas の輻輳制御が正しく動作していない
  - ECN を受信しなくなるまでウィンドウサイズを減少

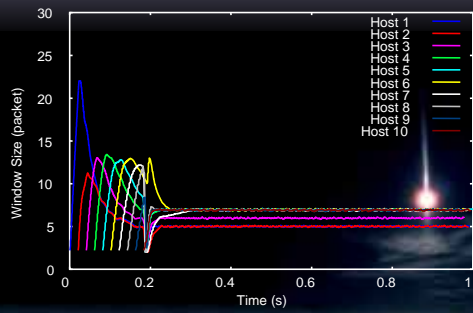
SSE 99-3 Hiroyuki Ohsaki 17

### 提案方式

- ◆ IP パケットの TOS フィールドを利用
- ◆ 受信した ACKパケットのうち ECN の比率を計算
 
$$e(k) = \frac{N_e}{N_a}$$
- ◆  $e(k)$  が 0 になるようにウィンドウサイズを制御
 
$$w_n(k+1) = \max(w_n(k) + \delta(\gamma - d(k)) - w_n(k)e(k), 0)$$
- ◆ IETF で議論されている ECN とは異なる

SSE 99-3 Hiroyuki Ohsaki 18

## 提案方式の場合 ( $\delta=0.4$ )



SSE 99-3

Hiroyuki Ohsaki

19

## まとめ

- ◆ TCP Vegas にもとづくウィンドウ型フロー制御
- ◆ 安定性解析
- ◆ シミュレーション
- ◆ TCP Vegas の問題点を指摘
  - 伝播遅延時間を正確に観測できない
  - コネクション数に関するスケーラビリティがない
- ◆ ECNを用いた問題点の改善
  - コネクション間の公平性が大幅に改善

SSE 99-3

Hiroyuki Ohsaki

20