

## TCPプロキシ機構の 実ネットワーク上での実装評価

大阪大学  
山根木 果奈

## 発表内容

- 研究の背景
- 実ネットワーク上における実験結果
  - 実験環境
  - TCP プロキシの効果
  - 高速 TCP の効果
- まとめと今後の課題

2005/02/17

2

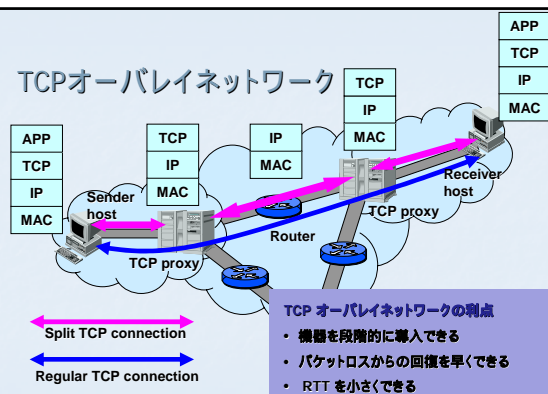
## 研究の背景

- 広帯域アクセス網技術の進展
- ユーザ数の爆発的な増加  
多種多様で高度なサービス要求
  - ベストエフォート型のネットワークでは困難
- TCP オーバレイネットワーク
  - トランスポート層における品質制御

2005/02/17

3

## TCPオーバレイネットワーク

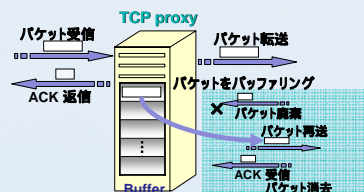


2005/02/17

4

## TCP コネクション分割機構

- TCP コネクションをネットワーク内のノード (TCPプロキシ) で分割
- TCP プロキシ
  - 擬似的な ACK を返送
  - 受信パケットをバッファリング
    - ➡ パケット廃棄時に TCP プロキシから再送
- TCP コネクションのフィードバックループが小さくなり、データ転送効率が向上する



2005/02/17

5

## 研究の目的

- TCP プロキシ機構の実ネットワークでの実装評価
  - 東京大阪間の公衆インターネット回線を利用
  - 実ネットワークでの有効性
- 高速 TCP を TCP プロキシ間に用いた場合の効果の評価
  - HSTCP (HighSpeed TCP)
  - gHSTCP (gentle HSTCP)

2005/02/17

6

### 実験環境

- 送受信エンド端末 4 台
- TCP プロキシ 2 台
- ネットワークエミュレータ
  - 東京 - 大阪間相当 (遅延なし)
  - 東京 - 沖縄間相当 (遅延: 25 msec)

大阪大学 (大阪府豊中市)    NEC (神奈川県川崎市)

osaka A    TCP proxy A    TCP proxy B    tokyo A

Internet    Network Emulator    tokyo B

Case2:
 

- TCP プロキシを経由
- 3つのTCPコネクションに分割

Case1:
 

- 通常のTCPコネクション
- 送受信ホスト間に設定

### 実験方法

- 帯域測定ツール iperf を用いてトラフィックを送出
- スループット
  - 受信側ホストの単位時間あたりの受信データ量
  - 受信側エンド端末で計測
- RTT, cwnd
  - Case1: 送信側エンド端末で計測
  - Case2: 送信側エンド端末に近い TCP プロキシで計測

大阪大学 (送信側)    NEC (受信側)

osaka A    TCP proxy A    TCP proxy B    tokyo A

Internet    Network Emulator    tokyo B

RTT, cwnd 測定    スループット測定

2005/02/17    8

### 実ネットワーク環境調査

- スループット
  - コネクション数を 5 本以上にしてもスループットは向上しない
- RTT と cwnd
  - cwnd の増減にともない RTT が増減

ボトルネックリンクの帯域: 約 70 Mbps

往復伝播遅延時間: 約 18 msec

最大 RTT: 約 30 msec

ボトルネックリンクの出力バッファ: 約 12msec (約 100 KBytes)

2005/02/17    9

### TCPプロキシの効果

Case1: 通常のTCP    Case2: TCPプロキシあり

Throughput (Mbps)

大阪 大阪 東京 沖縄 東京 沖縄 東京

64K 2M 64K 2M 64K 2M 64K 2M

• 送受信エンドホストの TCP バッファの大きさが性能に関係

- TCP バッファが小さい場合も高いスループットが得られる
- 遅延が大きい場合は帯域を有効利用できていない
- リンク帯域約 70 Mbps に対し、スループット約 30Mbps
- TCPプロキシ間のTCPコネクションに高速TCPを入れることで性能改善を図る

2005/02/17    10

### 関連研究: 高速TCP

- TCP Reno
  - 輻輳ウィンドウ (cwnd) の増加: 1/バケット/RTT
  - パケット廃棄が起こると: cwnd を半減
  - 帯域の利用率が悪い
- HSTCP (High Speed TCP)
  - cwnd の増加:  $a(w)/RTT$
  - パケット廃棄が起こると:  $cwnd \times b(w)$
  - 帯域を専有し、共存するコネクションとの公平性を保つことができない

Window Size    Congestion Window (pkts)

HighSpeed TCP    TCP Reno

Decrease less than Reno    Halve when packet loss occurs

Increase faster than Reno    1 pkt/RTT increase

$a(w)$      $b(w)$

2005/02/17    11

### 関連研究: gHSTCP (gentle High Speed TCP)

- cwnd の増加方法に 2 つのモード
- RTT の増加傾向を計測し、輻輳を検知
  - 輻輳時: Reno モード
  - それ以外: HSTCP モード
- gHSTCP:
  - パースト的なパケット廃棄を回避
  - 共存する TCP Reno との公平性を保つ

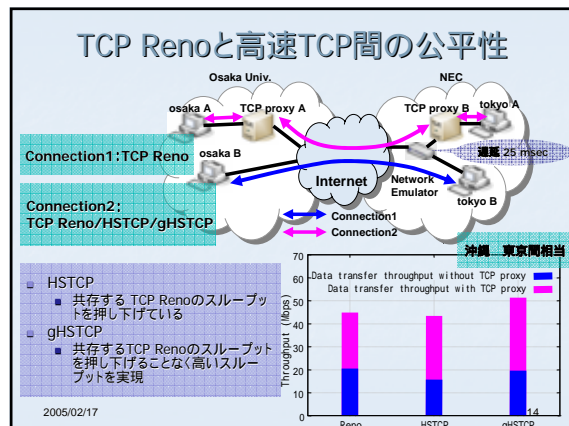
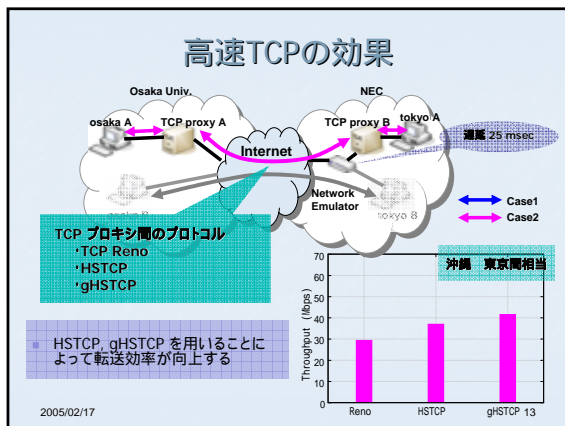
Link Bandwidth    Time

HighSpeed TCP    Gentle HighSpeed TCP

Reno mode/HSTCP mode

HSTCP mode

2005/02/17    12



- ### まとめと今後の課題
- TCP プロキシ機構を用いることで、データ転送効率が向上することが実証された。
    - エンドホストの特別な設定は不要
    - TCP プロキシ間に高速 TCP を用いることで、さらに転送効率が向上
  - 今後の課題
    - HSTCP, gHSTCP のパラメータセッティング
    - 大規模公衆網を用いた実験ネットワークにおける検証
    - TCP プロキシ機構の高速ネットワークにおける性能評価
- 2005/02/17 15