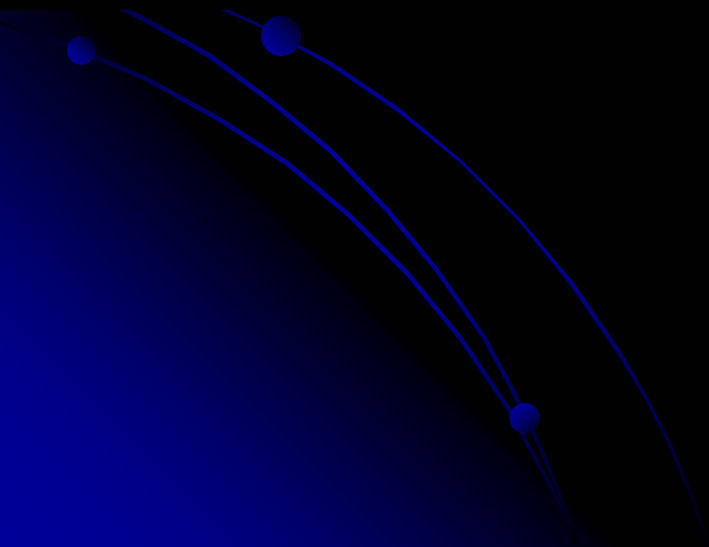


# インラインネットワーク計測に基づくTCPスループットの保証手法

大阪大学 山根木 果奈/長谷川 剛/村田 正幸



# 研究背景

- 一定の通信品質を要求するアプリケーション
    - リッチコンテンツの配信、定常的に発生するデータの配信
      - 一定スループット
    - 動画像のリアルタイム配信アプリケーション
      - 遅延ジッタ、伝送遅延
- ⇒ネットワークに対する通信品質の確保が求められている

- 従来のアプローチ
  - IP層における制御
    - ⇒スケーラビリティ、コスト面に問題
  - アプリケーション層における制御
    - ⇒アプリケーションごとの制御組み換えによる非効率性
    - ⇒異なるアプリケーションが競合した場合に、相互干渉や公平性の考慮が困難

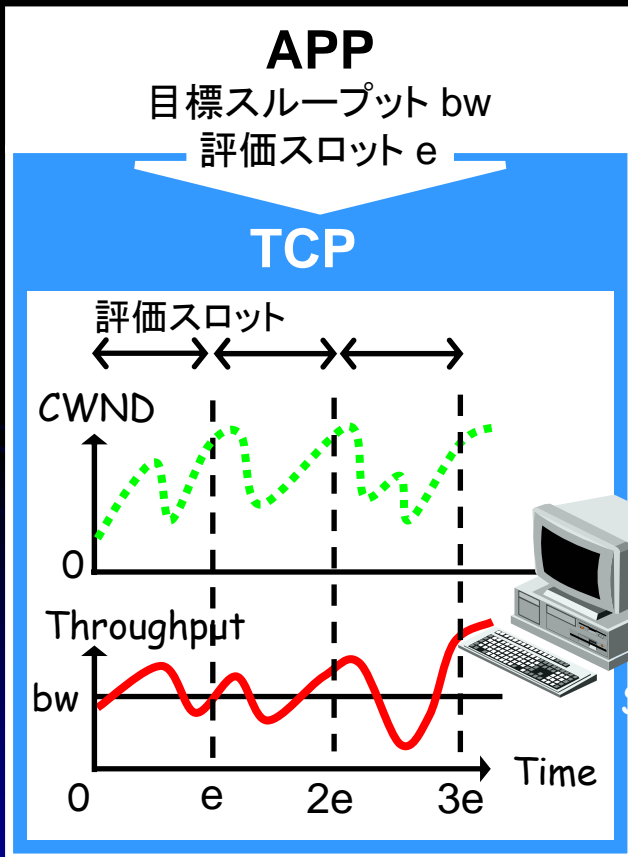
# TCP層において一定スループットの確保を目指す手法 [3]

- 送信側TCPを改良することで一定スループットの獲得を目指す
- 一定区間の平均スループットが一定値以上になるように制御

- ・ 一定区間 ⇒ 評価スロット
- ・ 一定値 ⇒ 目標スループット

アプリケーション層から指定され、TCPがこれら値に基づき輻輳ウィンドウサイズを制御

評価スロットごとのスループットが  
目標スループットになるようにパケットを送信する



[3] 山根木果奈, 長谷川剛, 村田正幸, “スループット保証を実現するTCP の輻輳制御方式の提案と評価,” 電子情報通信学会技術研究報告 (IN2005-189), Mar. 2006.

# 輻輳ウィンドウサイズの設定

ACKパケットを受け取るたびに

$$cwnd \leftarrow cwnd + \frac{k}{cwnd} \quad (1)$$

制御スロットにおける平均スループットが目標値に到達するために必要なパケット数を考慮し  $k$  を設定

- 評価スロットよりも細かいサイクル (制御スロット) をもとに設定する
  - 制御スロットごとにスループットの目標値を設定
  - 目標スループットおよび前制御スロットの獲得スループットを考慮

ただし

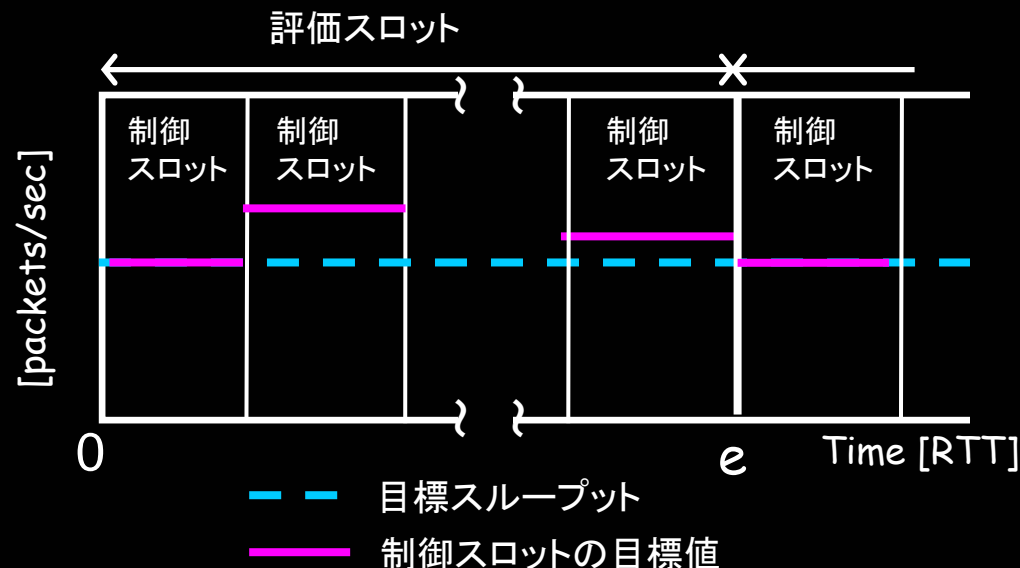
$$k_{\min} \leq k \leq k_{\max}$$

[ $k_{\min}$  : 1 (=TCP Reno)]

ネットワークが空いている場合にRenoと同等のスループットを得る

[ $k_{\max}$ ]

大き過ぎる $k$ が設定されてネットワークに悪影響を与えることを防ぐ



ネットワークの空き帯域に応じて動的に設定する

# Inline measurement TCP (ImTCP) [5]

- ネットワークの空き帯域を計測する手法
  - データ・ACKパケットのみを用いて計測を行う
- ImTCP の特徴
  - 少ない数のパケットで計測を行うことができる
  - 短い周期 (1 – 4 RTT) で継続的に計測を行うことができる
- ImTCP の問題点
  - 計測結果が不正確になる場合が存在する

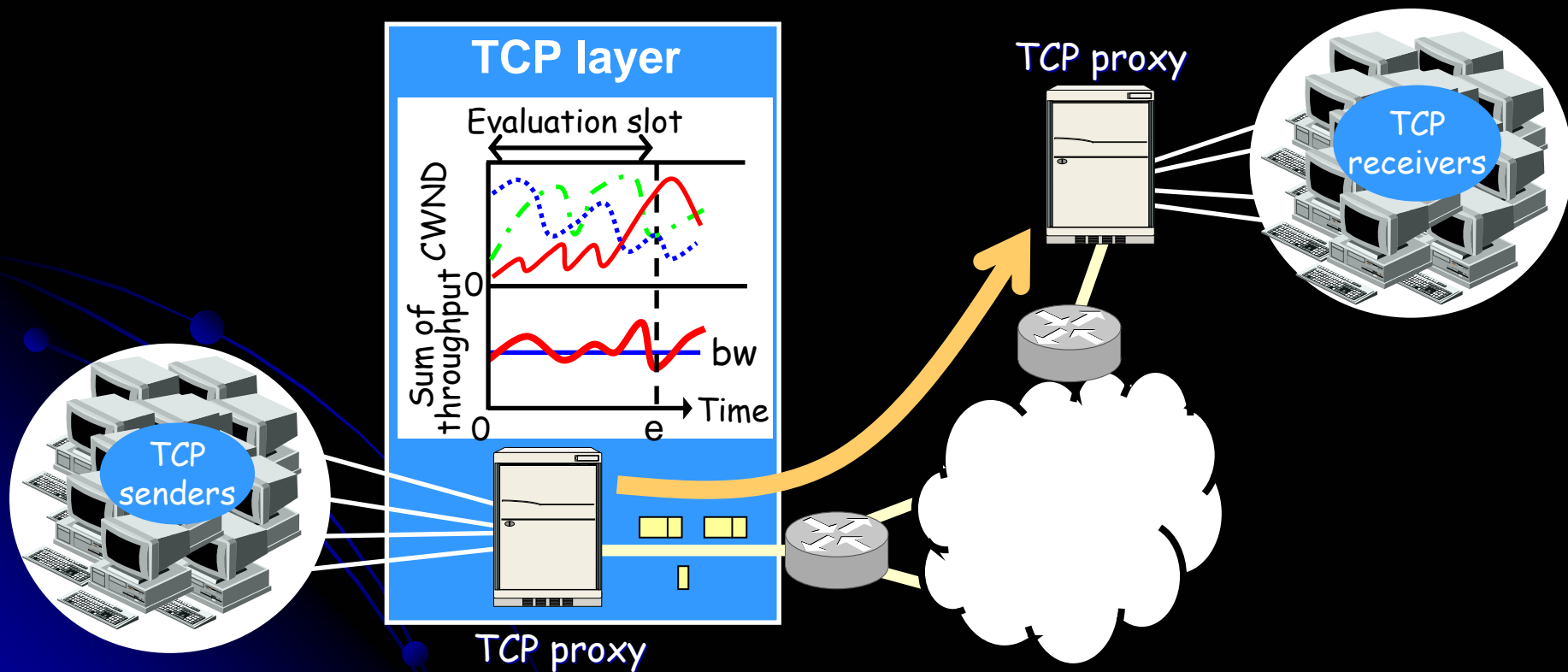
[5] L. T. M. Cao, G. Hasegawa, and M. Murata, "An Inline measurement method for capacity of end-to-end network path," in *Proceedings of IM 2005 E2EMON Workshop 2005*, May 2005.

# 研究の目的

- 今までの成果
  - 提案手法の本質的な性能を実証
    - 正確な利用可能帯域を取得可能であると仮定し、理想的な環境におけるシミュレーション
  - 1. 輻輳レベルに合わせて輻輳ウィンドウサイズの増加速度を変化させることによって一定のスループットを高い確率で獲得できる
  - 2. 提案手法を用いるコネクションが複数本競合している環境では、利用可能帯域に関する情報が重複し、性能が悪くなる
- 本研究の目的
  - ImTCP による利用可能帯域の計測値に基づいた制御を導入したシミュレーション
  - 複数のTCPコネクションの合計スループットをまとめて制御する手法の提案
    - 提案手法が複数本混在する場合に生じる問題を解決
    - 企業ネットワークにおいて2拠点間のトラフィックのスループット安定
    - TCP ストリーミングサーバに適用

# 複数本のTCPコネクションの合計スループットを管理する手法

- 送信ホスト、受信ホストがトランスポート層プロキシ(TCP proxy)に接続
- TCP proxy が複数のTCPコネクションの輻輳ウィンドウサイズを管理・制御
- TCP proxy 間の合計スループットが一定値以上になるように制御



# 輻輳ウィンドウサイズの制御

- 制御の対象となるコネクション数  $N_{pm}$  を決定
  - 輻輳回避フェーズにあるアクティブなコネクションの数
- ACK パケットが返ってきたときの  $k$  の計算方法の変更

1本の コネクション	$k = \frac{2\{(g_i \cdot srtt_i \cdot s - a_j) - (s - n_j - 1)cwnd_{n_j}\}}{(s - n_j - 1)(s - n_j)}$
複数 コネクション	$k = \frac{2\{(g_i \cdot srtt_i \cdot s - a_j^{sum}) / N_{pm} - (s - n_j - 1)cwnd_{n_j}\}}{(s - n_j - 1)(s - n_j)} \quad (2)$

目標スループット到達に必要な送信パケット数を制御対象のコネクション数  $N_{pm}$  で分割

ただし

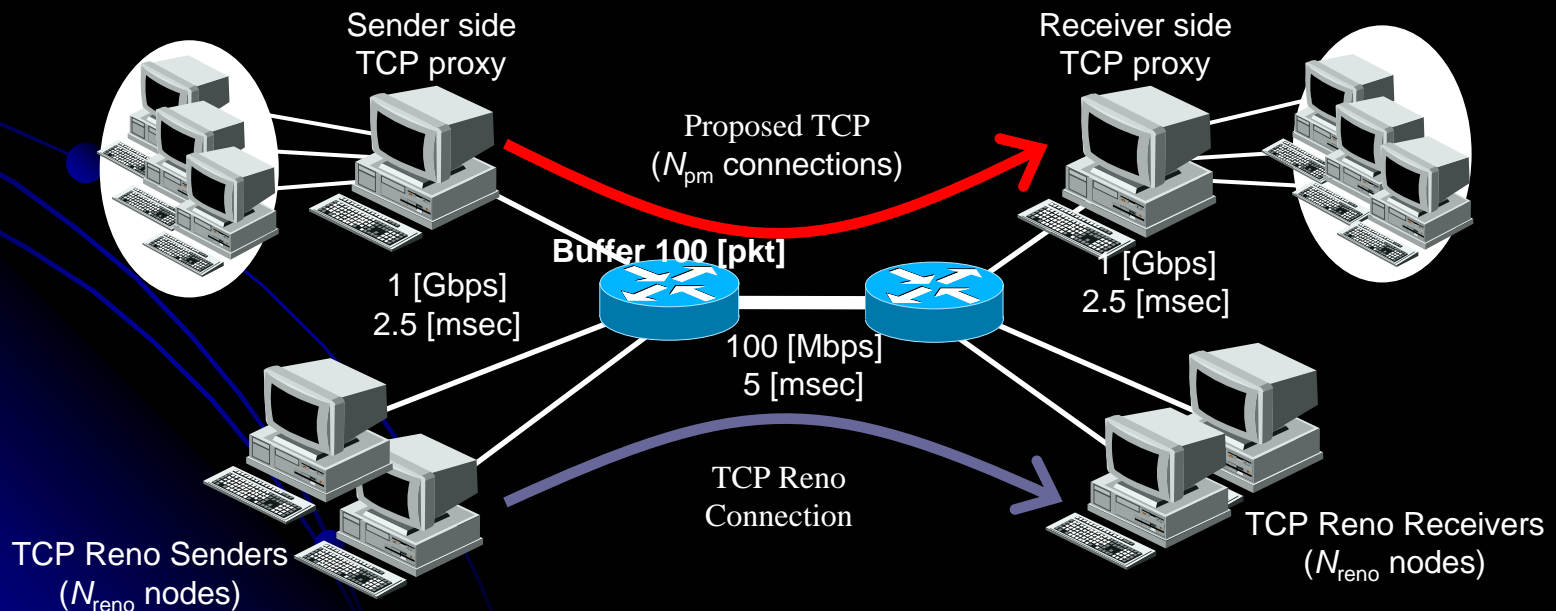
$$k_{min} \leq k \leq k_{max}$$

$g_i$  ( $i$  番目の制御スロットの目標値)     $srtt_i$  ( $i$  番目の制御スロットのSRTT)  
 $s$  (制御スロット長)     $a_j$  ( $j$  番目のACKを受け取った時点の packets 送信数)  
 $cwnd_{n_j}$  ( $j$  の  $cwnd$  番目のACKを受け取った時点)  
 $a_j^{sum}$  ( $j$  番目のACKを受け取った時点の packets 送信合計数)



# シミュレーションによる評価

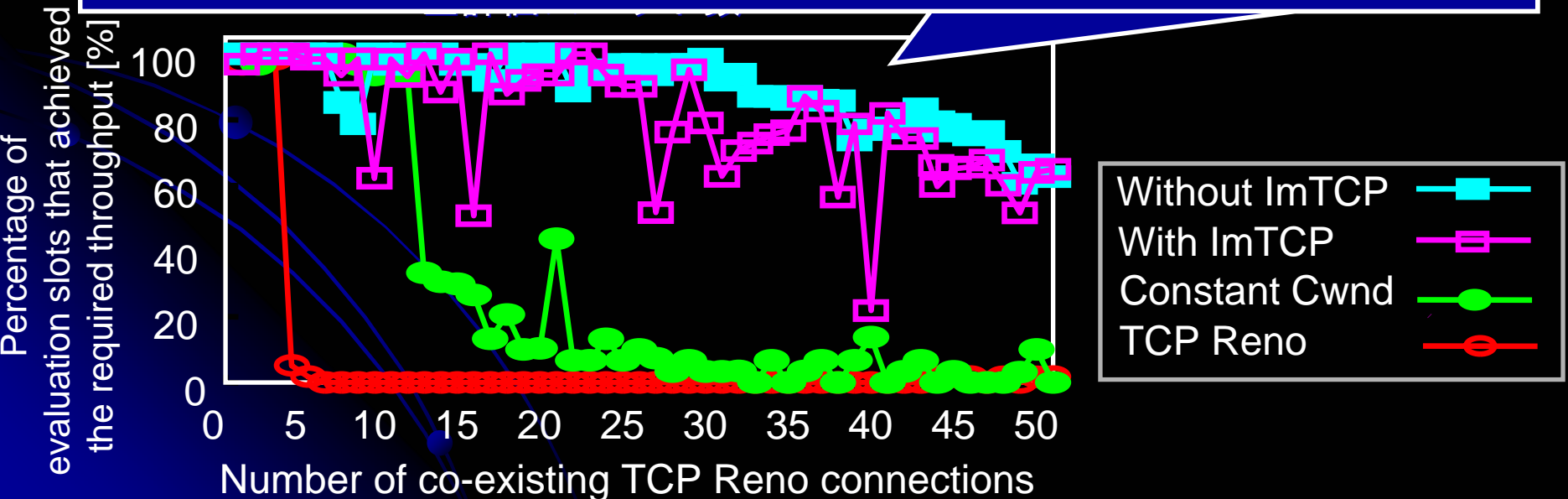
- ns-2 を用いたシミュレーション
  - ルータ: DropTail, バッファ100 packets
  - パケットサイズ: 1000 Bytes
- コネクション
  - 提案手法を用いるコネクション
  - 背景トラフィック (TCP Reno)



# ImTCPの導入による影響

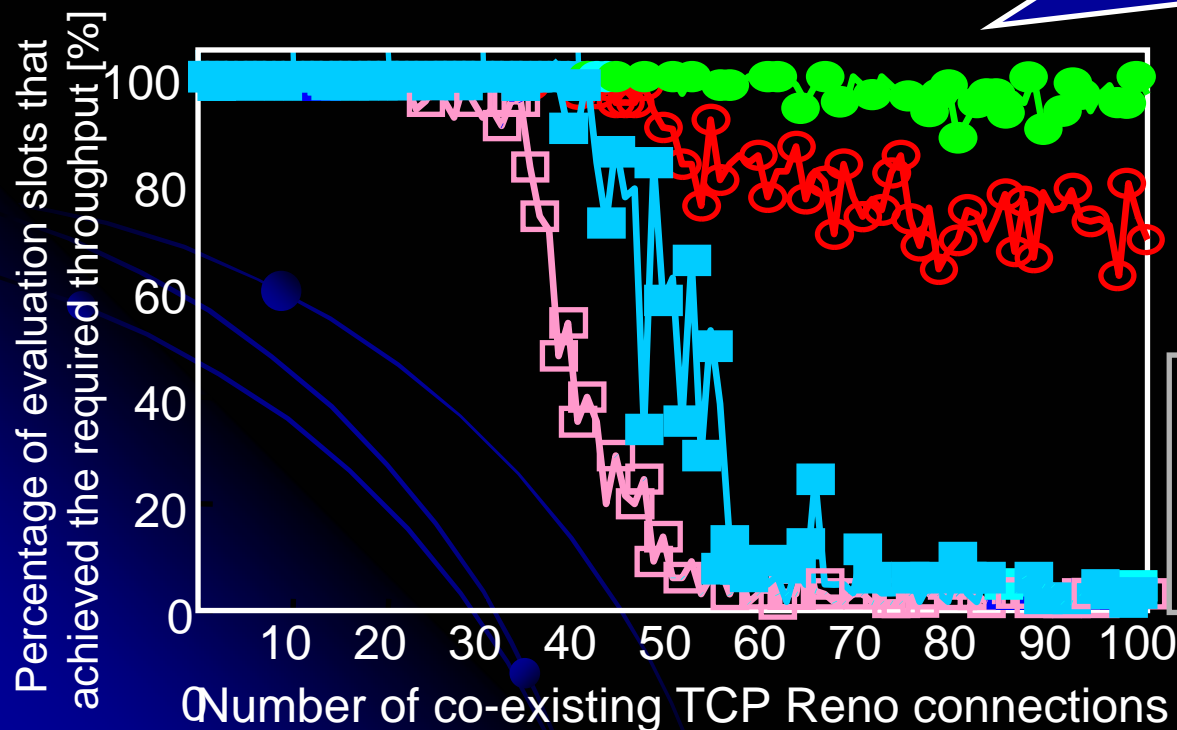
- 提案手法を用いるコネクション 1本

- 提案手法を用いた場合
  - ImTCP を用いた場合はスループットを獲得できた確率が不安定になるものの高い確率を保持している



# 複数本のTCPコネクションの合計スループットを管理する手法

- ・ 複数のコネクションを管理する手法
  - ・ ネットワークが混雑していても高い確率で目標スループットを獲得  
⇒ 利用可能帯域に関する情報を複数のコネクションで共有
  - ・ 制御対象であるコネクション数が多いほど性能が安定する



- ・ 複数本を制御 (5 connections) —●—
- ・ 複数本を制御 (10 connections) —●—
- ・ TCP Reno (10 connections) —□—
- ・ 1本を制御 (10 connections) —■—

# まとめ

- 一定のスループットをTCPの制御によってアプリケーションに提供する手法
  - ImTCP 導入による大きな性能劣化はない
- 複数のTCPコネクションの合計スループットをまとめて管理する手法を提案
  - 安定した確率で一定のスループットを獲得できる

## 今後の課題

- 提案手法の実機実装を用いた実験による性能評価

ご清聴ありがとうございました