



帯域計測に基づく TCPの輻輳制御方式の 提案と評価

大阪大学 大学院情報科学研究科
情報ネットワーク学専攻
井口 智仁

t-iguti@ist.osaka-u.ac.jp



TCP の輻輳制御機構

- 主な目的

- 輻輳の発生を避けながらリンク帯域を使い切る
 - 複数コネクション間の公平性を実現できるように

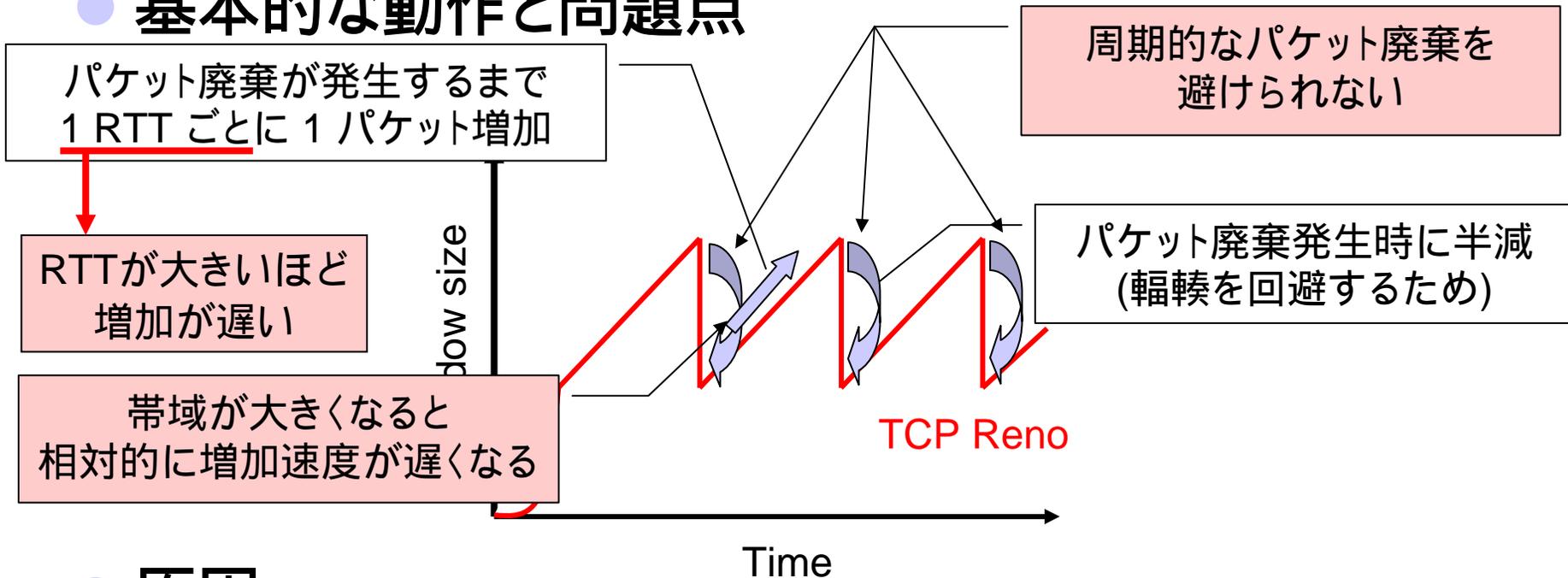
- ウィンドウサイズを増減させることで実現

- ウィンドウサイズ:
確認応答無しにネットワーク内へ送出可能なデータ量



TCP Reno のウィンドウサイズ制御

● 基本的な動作と問題点



● 原因

- 利用可能帯域に関する情報を持たない
- RTT の大きさに依存する制御を行っている

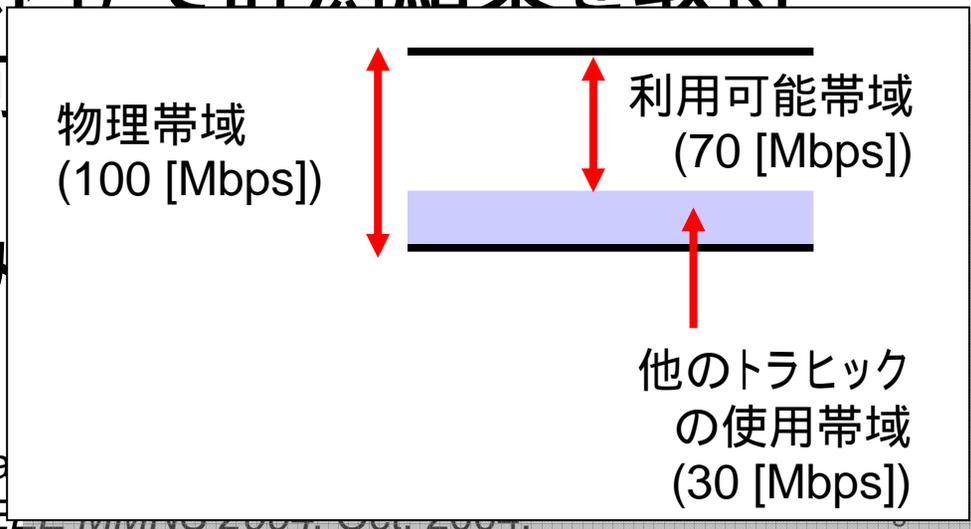
研究の目的

- TCP Reno が持つ問題を解決した新しい TCP の輻輳制御方式の提案
 - 帯域情報を利用したウィンドウサイズ制御アルゴリズム
 - インライン計測によって帯域情報を取得
 - 数理生態学における数学モデルを利用したウィンドウサイズ制御を使用



ImTCP: インラインネットワーク計測手法

- TCPのデータ転送パケットのみを使用して計測
 - TCP のデータ転送と併用可能
- エンド端末間のボトルネックリンクの物理帯域, 利用可能帯域を計測可能
- 高い頻度 (4 RTT以内) で計測結果を取得
 - ネットワークの利用可素早く追隨可能
- 任意の TCP に組み



参考文献:

M. L. T. Cao, G. Hasegawa, and M. Murata
TCP connection," in *Proceedings of IFIP/IEEE*



ロジスティック増殖モデル

● 1種の生物の個体数変化を表す

$$\frac{d}{dt} N(t) = \varepsilon \left(1 - \frac{N(t)}{K} \right) N(t)$$

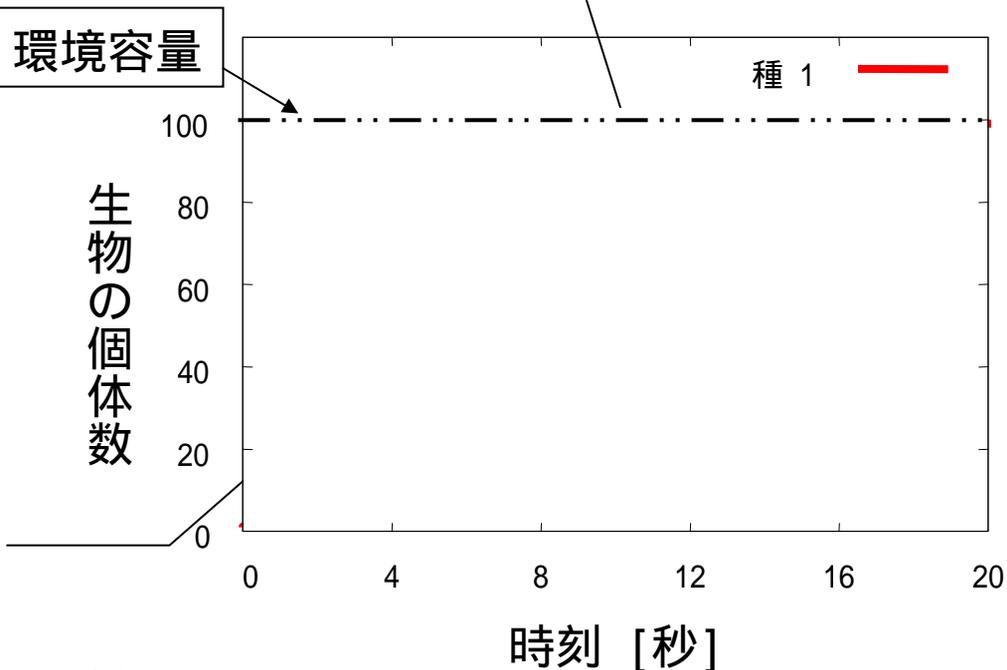
N : 個体数
 K : 環境容量
 ε : 種内の自然増加率

環境容量

生物の個体数

個体数は収束する

増加速度は環境容量に応じて調節される





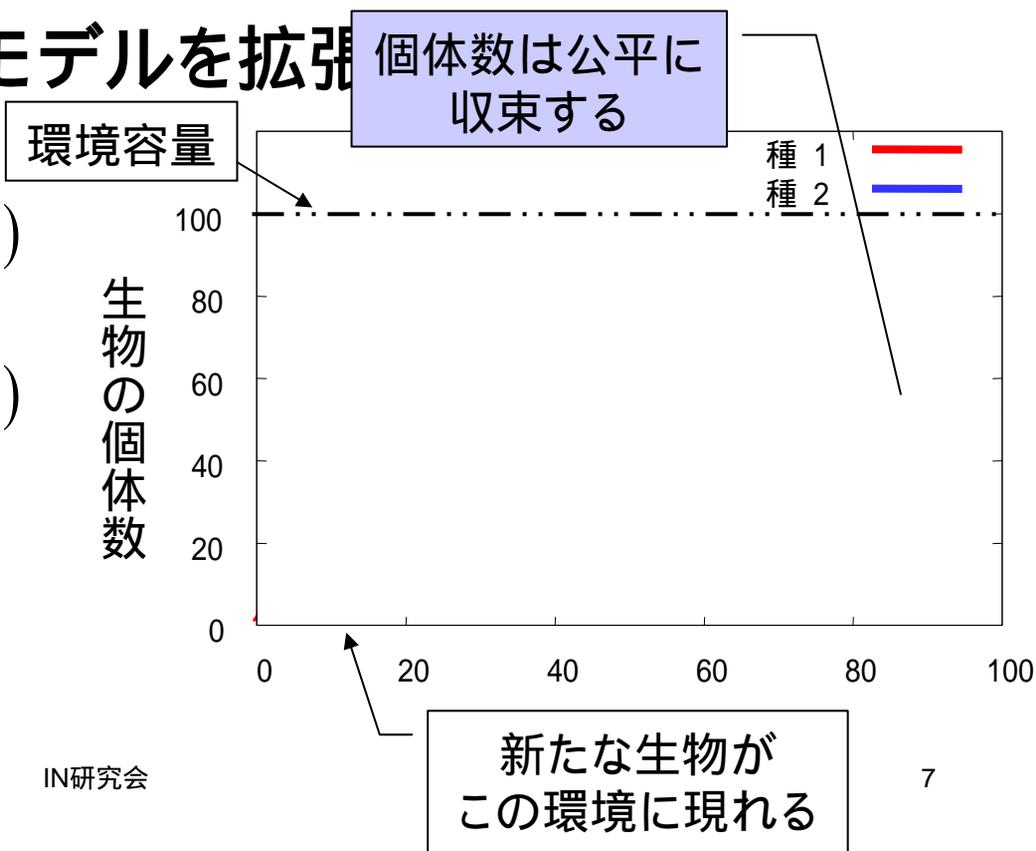
ロトカ・ヴォルテラ競争モデル

- 互いに競争関係にある2種以上の生物の個体数変化を表す

- ロジスティック増殖モデルを拡張

$$\frac{d}{dt} N_1(t) = \varepsilon \left(1 - \frac{N_1(t) + \gamma \cdot N_2(t)}{K} \right) N_1(t)$$
$$\frac{d}{dt} N_2(t) = \varepsilon \left(1 - \frac{N_2(t) + \gamma \cdot N_1(t)}{K} \right) N_2(t)$$

N : 個体数
 K : 環境容量
 ε : 種内の自然増加率
 γ : 種間競争による増加率の低下



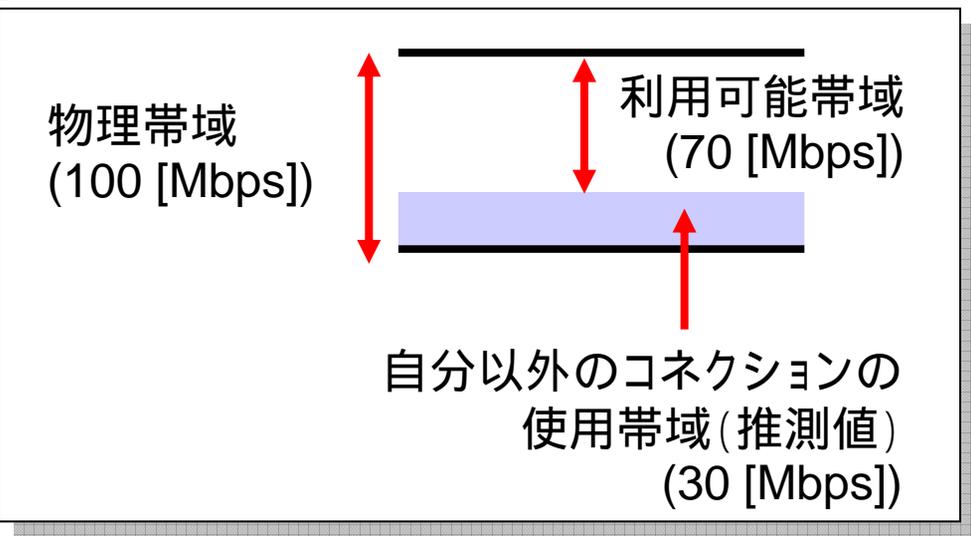


提案方式の輻輳制御への適用

- TCP のデータ転送速度の制御へ変換
 - 個体数 データ転送速度
 - 環境容量 物理帯域
 - 自分以外の生物の個体数
 自分以外のコネクションの使用帯域
 (物理帯域 - 利用可能帯域) として推測する

● ウィンドウサイズ制御式

$$w_i(t) = \frac{w_i(0) \cdot e^{\varepsilon \cdot t \left\{ 1 - \gamma \left(1 - \frac{A_i}{K} \right) \right\}}}{w_i(0) \cdot \left(e^{\varepsilon \cdot t \left\{ 1 - \gamma \left(1 - \frac{A_i}{K} \right) \right\}} - 1 \right)}$$





シミュレーションによる評価

- ネットワークシミュレータ：NS-2

- 比較対象

- TCP Reno

- HighSpeed TCP (HSTCP)

- Scalable TCP (STCP)

- FAST TCP

輻輳の検出にパケット廃棄ではなく
RTT を用いている
(TCP Vegas の改善)

帯域遅延積の大きい
ネットワークに対応できる方式

UDP Traffic

r_{udp} [Mbps]

BW [Mbps]

τ [msec]

bw_i [Mbps]

5 [msec]

TCP Traffic

バッファサイズ
= TCP コネクションの帯域遅延積



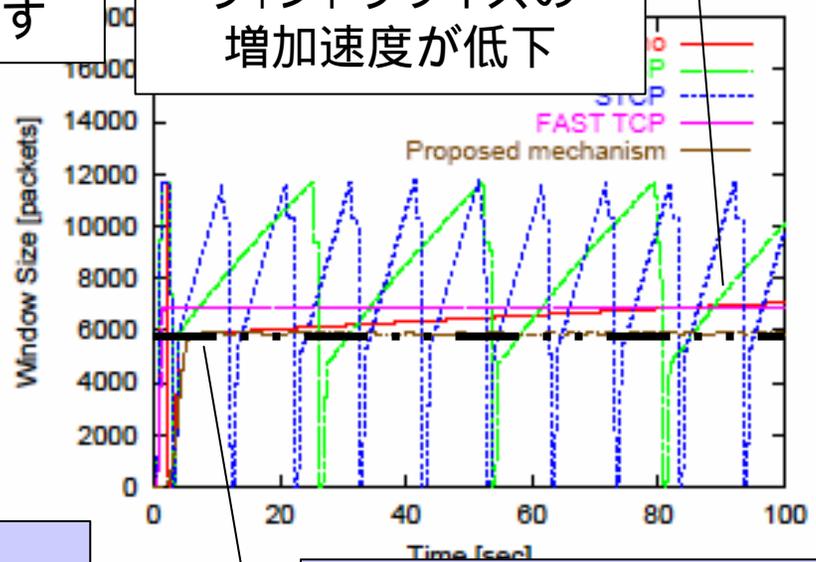
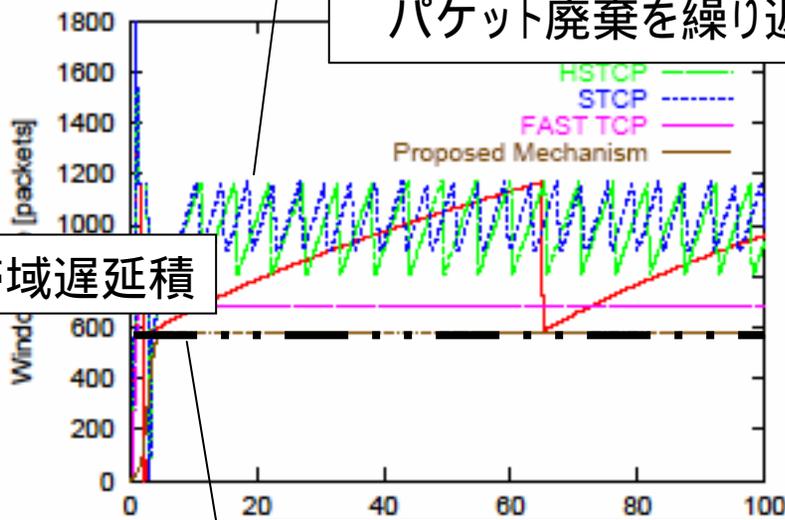
結果: ウィンドウサイズの変化

● 帯域 BW

Reno, HSTCP, STCP は
パケット廃棄を繰り返す

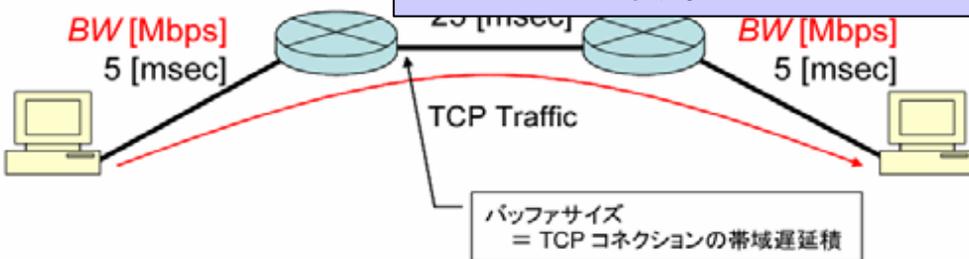
Reno, HSTCPは
ウィンドウサイズの
増加速度が低下

帯域遅延積



FAST, 提案方式は
パケット廃棄を発生させず,
収束する

STCP, FAST, 提案方式は
増加速度が
ほとんど変化しない



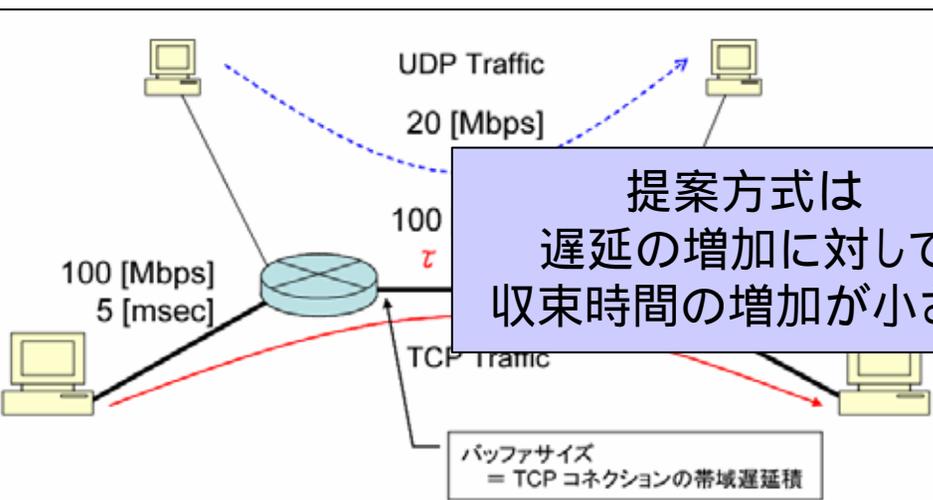
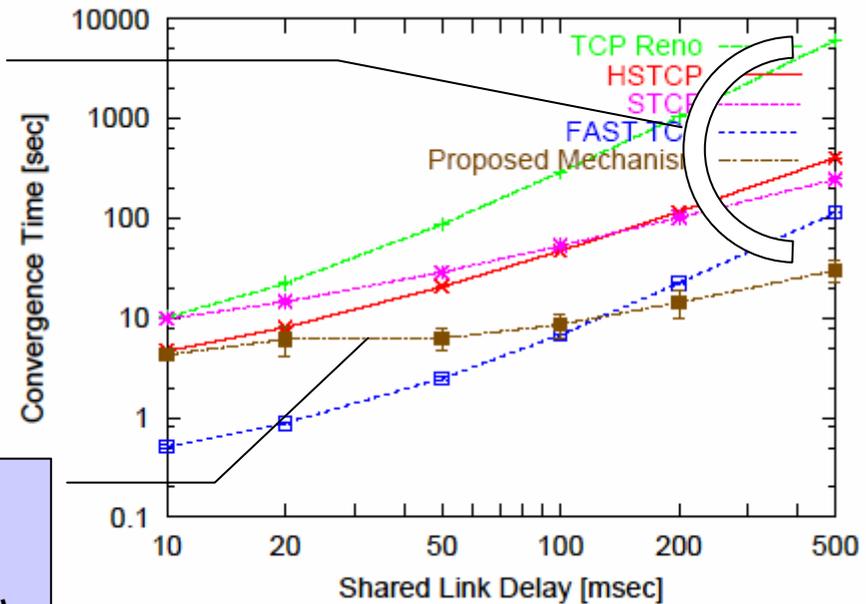
結果: ボトルネックリンク遅延の変化に対する収束時間



- 遅延の変化幅は 10 ~ 500 [msec]
 - 収束時間: 帯域の 99% を使い切るまでの時間

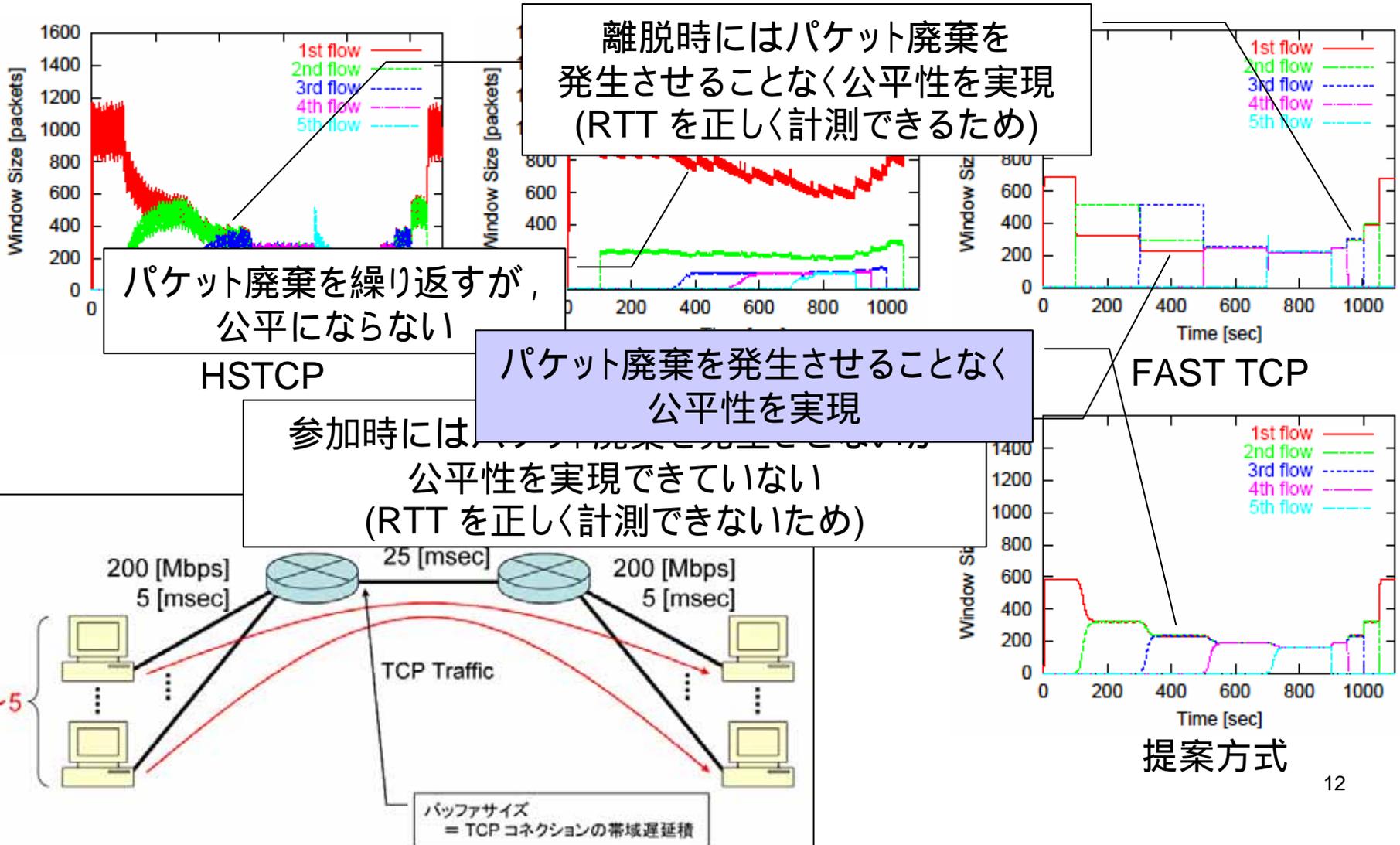
提案方式以外の方式は遅延の増加に対して収束時間の増加が大きい

提案方式は遅延の増加に対して収束時間の増加が小さい





結果：コネクション数の変化による影響



まとめと今後の課題



●まとめ

- 帯域計測に基づくTCPの輻輳制御を提案
- 解析とシミュレーションで次の性質の実現を確認
 - 帯域, 遅延の変化に対応可能
 - コネクション間でウィンドウサイズを公平に収束させる

●今後の課題

- 異なる遅延を持つコネクション間の公平性評価
- 帯域計測結果の誤差を考慮した制御
- 実ネットワーク上での実装実験による性能評価