

# 改造TCP がネットワークに与える影響に関する一検討

○丸山 純一†, 長谷川 剛‡, 村田 正幸†  
 †大阪大学大学院情報科学研究科  
 ‡大阪大学サイバーメディアセンター

## 発表内容

- 研究背景
- 研究目的
- tampered-TCP の定義
- スループット解析
- シミュレーション評価
- まとめと今後の課題

## 研究背景

- 現在のインターネットには改造 TCP が存在
  - カーネル内のソースコードを少し変えれば改造可能
  - ex. 輻輳ウィンドウサイズの増減幅の変更
    - ⇒ 輻輳ウィンドウサイズの上げ幅・下げ幅を決定するパラメータを変更するだけでよい
- tampered-TCP が増加すると公平性が保たれない
  - ex. 輻輳ウィンドウサイズの増減幅の変更
    - ⇒ tampered-TCP コネクションが多くの帯域を占める可能性
    - ⇒ 既存の TCP Reno コネクションのスループットが低下
- しかし, tampered-TCP の効果が出ない可能性もある
  - ex. 輻輳ウィンドウサイズの増減幅の変更
    - ⇒ 輻輳ウィンドウサイズが急激に大きくなる
    - ⇒ パケット廃棄数が増加し, タイムアウトが増加

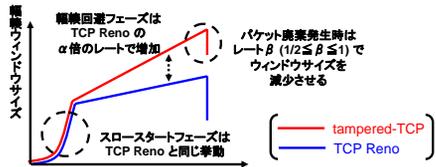
## 研究目的

- tampered-TCP がネットワークに与える影響を明らかにする
  - 数学的解析とシミュレーションによる評価により, 次の3点を導く
    - 上げ幅が3以上の場合, タイムアウトが発生してスループットが低下すること
    - 下げ幅を小さくすることでスループットが増大すること
    - 下げ幅を小さくすることの効果より, 上げ幅の増加によるスループット低下の影響の方が大きいこと

tampered-TCP の有効領域が狭いことを示し, tampered-TCP の減少をねらう

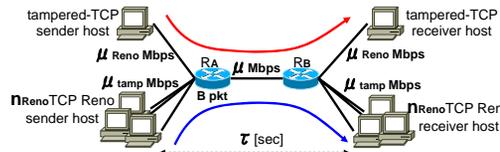
## tampered-TCP の定義

- 輻輳ウィンドウサイズの上げ幅・下げ幅を変更した tampered-TCP に着目
  - 改造者の視点: 改造が容易かつ効果が期待できる
  - 研究者の視点: 数学的解析手法による性能解析が可能



## 評価方法

- 数学的解析とシミュレーションを用いた評価
  - 評価モデル

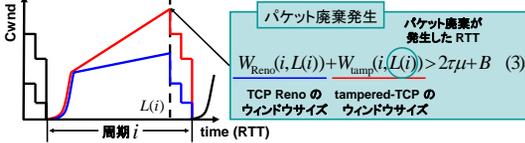


- 評価指標

$$\text{スループット比} = \frac{(\text{tampered-TCP のスループット})}{(\text{TCP Reno のスループット})}$$

### 数学的解析手法による評価 (1/3)

- パケット廃棄をきっかけとした周期的なウィンドウサイズの変化をモデル化 [7]



$$W_{\text{Reno}}(i, L(i)) + W_{\text{tamp}}(i, L(i)) > 2\tau\mu + B \quad (3)$$

TCP Reno のウィンドウサイズ  
tampered-TCP のウィンドウサイズ

- TCP の特性を考慮
  - 1 ウィンドウで 3 個以上のパケット廃棄 ⇒ タイムアウト [8]
- tampered-TCP の特性を考慮
  - 1 ウィンドウの廃棄パケット数 ⇒  $\alpha$  個

[7] K. Tokuda, G. Hasegawa, and M. Murata "Performance analysis of HighSpeed TCP and its improvement for high throughput and fairness against TCP Reno connections," in Proceedings of IEEE High Speed Network Workshop 2003 (HSN'03), (San Francisco), Mar. 2003.  
[8] K. Fall and S. Floyd, "Simulation-based Comparisons of Tahoe, Reno, and SACK TCP," Computer Communication Review, vol. 26, pp. 5-21, July 1996.

### 数学的解析手法による評価 (2/3)

- タイムアウトが発生しない場合
  - $\alpha \leq 2$  の場合
  - 周期  $i$  開始時のウィンドウサイズ  $W_{\text{tamp}}(i, 1) = \beta^\alpha W_{\text{tamp}}(i-1, L(i-1))$
  - ⇒  $j$  RTT 目のウィンドウサイズ  $W_{\text{tamp}}(i, j) = \beta^\alpha W_{\text{tamp}}(i-1, L(i-1)) + \alpha \cdot j$
- タイムアウトが発生する場合
  - $\alpha > 2$  の場合
  - スロースタートフェーズも考慮 ⇒  $j$  RTT 目のウィンドウサイズ  $W_{\text{tamp}}(i, j) = \beta^\alpha W_{\text{tamp}}(i-1, L(i-1)) + \alpha [j - \log_2 \{\beta^2 W_{\text{tamp}}(i-1, L(i-1))\}]$  (スロースタートフェーズにかかった RTT 数)

tampered-TCP のウィンドウサイズ

$$W_{\text{tamp}}(i, j) = \begin{cases} \beta^\alpha W_{\text{tamp}}(i-1, L(i-1)) + \alpha \cdot j & (\text{if } \alpha \leq 2) \\ \beta^\alpha W_{\text{tamp}}(i-1, L(i-1)) + \alpha [j - \log_2 \{\beta^2 W_{\text{tamp}}(i-1, L(i-1))\}] & (\text{if } \alpha > 2) \end{cases} \quad (6)$$

### 数学的解析手法による評価 (3/3)

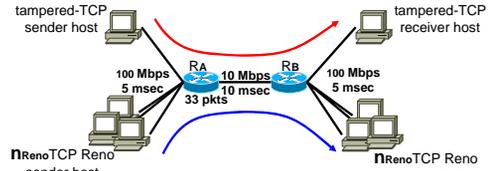
- 各 TCP コネクションの平均スループット  
- (1 RTT 毎のウィンドウサイズの合計) / (RTT の合計)

$$\rho_{\text{tamp}} = \frac{\sum_{i=1}^{\infty} \sum_{j=1}^{L(i)} W_{\text{tamp}}(i, j)}{\sum_{i=1}^{\infty} \sum_{j=1}^{L(i)} (Q(i, j) + 2\tau)}$$

$$\rho_{\text{Reno}} = \frac{\sum_{i=1}^{\infty} \sum_{j=1}^{L(i)} W_{\text{Reno}}(i, j)}{\sum_{i=1}^{\infty} \sum_{j=1}^{L(i)} (Q(i, j) + 2\tau)}$$

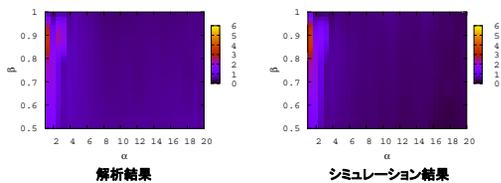
$$Q(i, j) = \frac{\max(W_{\text{Reno}}(i, j) + W_{\text{tamp}}(i, j) - 2\tau\mu, 0)}{\mu}$$

### シミュレーション環境



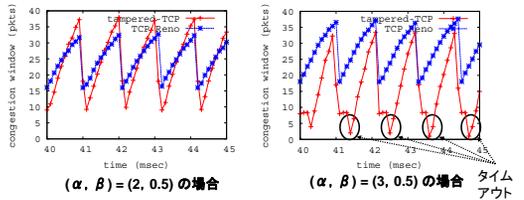
- パケットサイズは 1500 バイト
- シミュレーション時間は 60 秒
- tampered-TCP コネクションのパラメータを変化させる
  - $\alpha$  を 1~20,  $\beta$  を 0.5~1 で変化させる

### スループット比の変化 ( $n_{\text{reno}}=1$ )



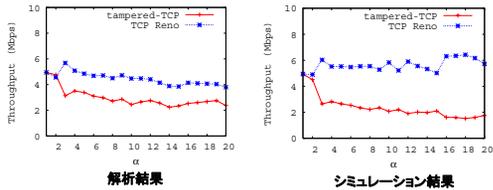
- 解析結果とシミュレーション結果がほぼ一致
- ほとんどのパラメータ領域で tampered-TCP が効果を発揮していない

### ウィンドウサイズの変動



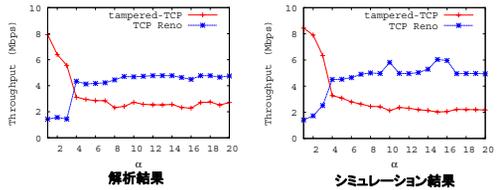
- $\alpha=2$  では、タイムアウトが発生していない
- $\alpha=3$  では、タイムアウトが頻発している
  - $\alpha$  が大きくなることで、急激にウィンドウサイズが増加 ⇒ パースト的なパケットロスが発生してタイムアウトが起こる

## 各 TCP コネクションのスループットの変動 ( $\beta=0.5$ )



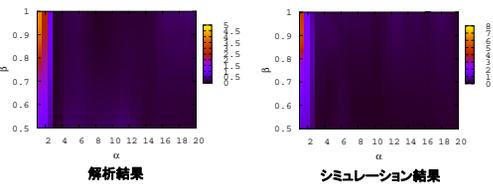
- $\alpha=3$  を境に tampered-TCP のスループットが激減している
  - 廃棄パケット数が 3 個を超え、タイムアウトが発生するため
- $\alpha$  の増加にしたがって tampered-TCP のスループットが低下
  - パケット廃棄の発生頻度が高くなり、タイムアウトも増加するため

## 各 TCP コネクションのスループットの変動 ( $\beta=0.9$ )



- $\alpha < 3$  の場合にのみ tampered-TCP が効果を発揮
  - タイムアウトが発生しないので、 $\beta$  を大きくした効果が出る
- $\alpha \geq 3$  の場合は  $\beta$  が大きくても tampered-TCP の効果は無い
  - タイムアウトによるスループット低下の影響の方が大きい

## スループット比の変化 ( $n_{c.o.}=5$ )



- 解析結果とシミュレーション結果がほぼ一致
- ほとんどのパラメータ領域で tampered-TCP が効果を発揮しない
- tampered-TCP コネクションの効果は  $n_{\text{Reno}}=1$  の場合より低下
  - パケット廃棄・タイムアウト発生までの RTT 数が減少するため

$$L(i) = \frac{(2\tau\mu + B) - n_{\text{Reno}} W_{\text{Reno}}(i-1) + W_{\text{tamp}}(i-1)}{\alpha + n} \quad (7)$$

## まとめと今後の課題

- tampered-TCP の有効領域が狭いことを示した
  - $\alpha \geq 3$  になるとタイムアウトを起こしてスループットが激減
  - $\alpha \geq 3$  になると  $\alpha$  の増加にしたがってパケット廃棄の発生頻度が高くなり、スループットが低下
  - $\alpha < 3$  かつ  $\beta$  が大きい場合のみ有効
  - $\beta$  が大きいことの効果より、 $\alpha$  の増加によるスループット低下の影響の方が強い
  - TCP Reno コネクションが複数本存在する環境でも同様
- 今後の課題
  - tampered-TCP の SACK オプションを ON にした評価
  - 他の種類の tampered-TCP の評価

## ご清聴ありがとうございました。

• ご清聴ありがとうございました。