3X-1

CUBIC TCP の RTT 公平性の改善

神津　智樹†　　大浦　亮‡　　秋山　友理愛†　　山口　実靖†

†工学院大学工学部情報通信工学科　　‡工学院大学大学院工学研究科電気・電子専攻

1. はじめに

TCP は現在のインターネットにおいて標準的に用いられているトランスポート層プロトコルである．高遅延環境下の通信性能には，OS の TCP 実装の動作が大きな影響を与える．多くの OS に実装されている典型的な TCP である TCP Reno では高遅延・広帯域ネットワーク環境においてネットワーク帯域を十分に使い切れないという問題が指摘されている[1]．TCP Reno に変わる高遅延環境で高い通信性能を提供できる TCP の輻輳制御アルゴリズムとして CUBIC TCP[2]が提案されている．

しかし，この TCP アルゴリズムでは RTT による性能の公平性は達成されていない．本研究では，CUBIC TCP の RTT による公平性に着目し，RTT 公平性の改善手法を提案し，その有効性の検証を行う．

2. 輻輳制御アルゴリズム

TCP はネットワークの輻輳状況に応じて輻輳ウィンドウサイズを動的に増減させ，ネットワークへ送出するデータ量を調節することにより輻輳制御を行う[3]．輻輳制御手法には，複数のアルゴリズムがあり，多くのサーバ OS で利用されている Linux では CUBIC TCP[2]と呼ばれる手法が用いられている．

3. CUBIC TCP

CUBIC TCP は，BIC TCP のスケーラビリティを維持しながら，TCP Fairness と RTT Fairness，制御手法の複雑さを改善した高速 TCP である[4]．

CUBIC TCP では，BIC TCP のバイナリーサーチを用いて，利用可能帯域を検索するアルゴリズムを次の式のような 3 次関数を用いた制御によって実現している．

$$cwnd = c(t-K)^3 + W_{max} \qquad (1)$$

$$\mathrm{K} = \sqrt[3]{\frac{W_{max}\beta}{C}} \qquad (2)$$

ここで，$cwnd$ は輻輳ウィンドウサイズ，t はパケットロス検出時からの経過時間，W_{max}はパケットロス検出時の輻輳ウィンドウサイズ，C は増加幅を決めるパラメータ，βはパケットロス検出時のウィンドウサイズ減少幅を表している．一般に Cは 0.4，βには 0.2 が用いられる．K は，式(1)における 3 次関数の変曲点までの時間であり，t=K において，輻輳ウィンドウサイズが前回のパケットロス検出時の値と等しくなる．よって K が小さいほど短い時間で輻輳ウィンドウサイズが回復することとなり，高い性能が得られやすくなる．一般に RTT が大きい通信ほど，輻輳ウィンドウサイズの値が性能に影響を与えより大きな輻輳ウィンドウサイズが必要となる．しかし，式(2)より，前回のパケットロス次の輻輳ウィンドウサイズが大きいほど K の値が大きくなり，高い性能が得られづらいことが分かる．よって，RTT が大きい通信ほど輻輳ウィンドウサイズが大きくなり，結果 K の値が大きくなり通信性能が低くなりやすいと予想される．これは，RTT 公平性を低下させる原因になると考えられる．

上記のようにパケットロス検出時からの経過時間を用いて，ウィンドウサイズの増加から RTT の影響を排することで RTT Fairness の向上を目指しており，BIC TCP の低遅延環境で輻輳ウィンドウサイズを急速に成長させすぎる問題を解決している．

4. RTT を考慮した K の決定

本章で RTT を考慮して K を調整する手法を提案する．式(5)に示すように，K の決定式を改善する．x(RTT)は RTT の関数であり，RTT が大きいほど小さくなる．RTT が大きい通信の K の値を小さくすると，それら通信の性能が向上し，結果として公平性が向上すると期待される．

$$\mathrm{K} = \sqrt[3]{\frac{W_{max}\beta}{C}} \times \mathrm{x(RTT)} \qquad (5)$$

5. 性能測定

5.1　実験環境

図 1 のようなネットワーク環境で実験 1 および実験 2 を行い，通信性能の測定を行った．

"RTT Fairness Improvement of CUBIC TCP"

†Tomoki Kozu, Yuria Akiyama, Saneyasu Yamaguchi, Department of information and Communications Engineering, Kogakuin University

‡Ryo Oura, Electrical Engineering and Electronics, Kogakuin University Graduate School

Dummynet はネットワーク遅延をエミュレーションする装置である．PC1，PC2，PC3 では Linux OS を用い，TCP は CUBIC TCP を実用いた．通信は PC1-PC3 間と PC2-PC3 間で同時に行い，PC1 と PC2 を送信者，PC3 を受信者とした．それぞれの受信ウィンドウサイズは 10MB に設定した．

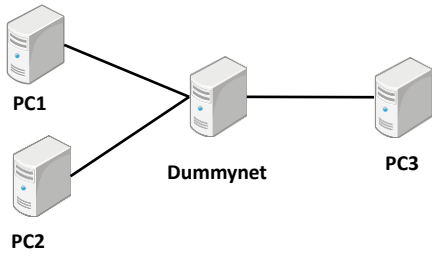

図 1　実験のネットワーク図

5.2　初期状態の K による性能測定

実験 1 では，PC1-PC3 間と PC2-PC3 間で netperf を用いて同時に通信を行い，それぞれの通信性能を測定した．PC1-PC3 間の RTT を 2ms に固定（RTT 固定）し，PC2-PC3 間の RTT は 2ms から 64ms と変動（RTT 変動）させて測定を行った．この時，K は式(2)の初期状態で測定を行った．

実験結果を図 2 に示す．横軸が RTT 固定と RTT 変動の RTT 比であり，縦軸が得られたスループットである．この結果から RTT 比が高いほど，公平性が低くなることが確認できた．これは RTT が大きいほど輻輳ウィンドウサイズが上昇するまでにかかる時間が長く，公平性が低くなったからであると考えられる．

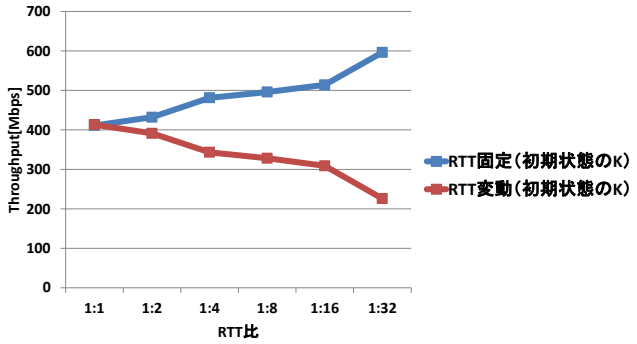

図 2　RTT と Throughput(初期状態の K)

5.3　RTT を考慮した K による性能測定

実験 2 では，PC1-PC3 間の PC2-PC3 間で netperf を用いて同時に通信を行い，それぞれの通信性能を測定した．PC1-PC3 間および PC2-PC3 間の RTT は実験 1 と同様である．この時，K は式(5)の RTT を考慮した K を用いて測定を行った．関数 x(RTT)は図 3 に示す通りである．

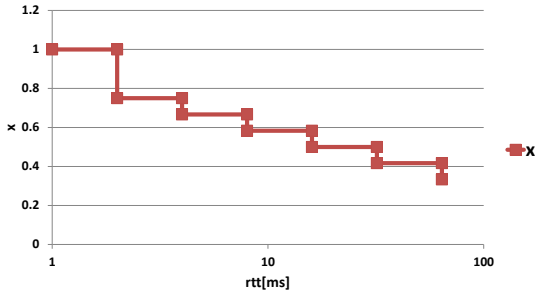

図 3　RTT における関数 x

実験結果を図 4 に示す．初期状態の K より RTT を考慮した K のほうが公平性が高く，提案手法が有効であることが確認できた．

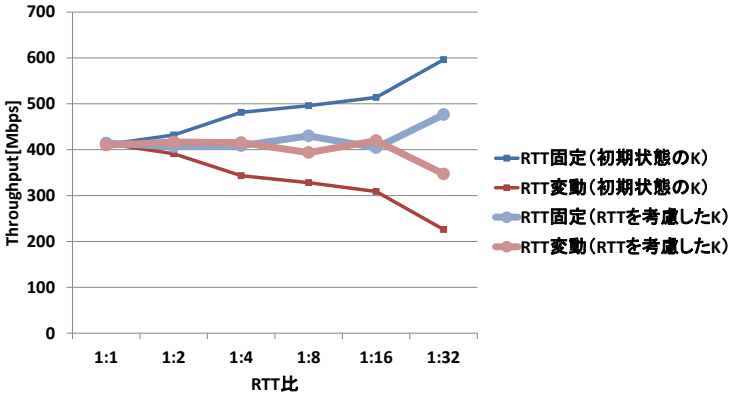

図 4　RTT と Throughput（RTT を考慮した K）

6. おわりに

本研究では CUBIC TCP の RTT Fairness 公平性改善手法として，RTT を考慮した K の決定法を提案し，性能測定を行った．結果，RTT を考慮した K が有効であることを確認した．

今後は，他の TCP アルゴリズムとの間の RTT 公平性について考察していく予定である．

謝辞

本研究は JSPS 科研費 22700039，24300034 の助成を受けたものである。

参考文献

[1] D.Katabi，M.Handley，and C.Rohrs，“Congestion control for high bandwidth-delay product networks” ,in Proceedings of ACM SIGCOMM 2002,Aug.2002.
[2] Injong Rhee, and Lisong Xu “CUBIC: A New TCP-Friendly
High-Speed TCP Variant.” In Proc. Workshop on Protocols for
Fast Long Distance Networks, 2005, 2005.
[3]大浦亮，山口実靖，“実機を用いた高速 TCP の公平性の評価” 第 10 回情報科学技術フォーラム(FIT2011) RL-003.
[4] 鈴川 竜司，安達 直世 “CUBIC-TCP におけるフロー間帯域の公平性に対する改善手法の提案” 電子情報通信学会信学技報 NS2008-108