

# 無線LAN環境における 移動端末間の公平性を考慮した輻輳制御手法の提案と評価

飯尾 明日香<sup>†</sup> 前野 誉<sup>††</sup> 高井 峰生<sup>†††</sup> 小口 正人<sup>†</sup>

<sup>†</sup>お茶の水女子大学 〒112-8610 東京都文京区大塚 2-1-1

<sup>††</sup>株式会社スペースタイムエンジニアリング 〒101-0025 東京都千代田区神田佐久間町 3-27-3 ガーデンパークビル 7F

<sup>†††</sup>UCLA Computer Science Department 3803 Boelter Hall, Los Angeles, CA 90095-1596, USA

E-mail: <sup>†</sup>asuka-i@ogl.is.ocha.ac.jp, <sup>††</sup>tmaeno@spacetime-eng.com, <sup>†††</sup>mineo@cs.ucla.edu,

<sup>††††</sup>oguchi@computer.org

あらまし 現在、センシングデバイスはスマートフォンや自動車のようなモバイル機器には不可欠な存在となっている。ユーザが、より便利、快適に使用するため、高精度なセンサが無数に搭載された移動端末は、センサ情報を収集することで、より正確で精密な周囲の環境情報を取得できる。この環境情報をコンテキストと呼ぶが、端末は、通常、取得したコンテキストを、スマートフォンの場合には、操作性の向上や高度なアプリケーションの実現等、また、自動車の場合には、車両安定制御や道路情報の把握等に利用している。本研究では、このコンテキストが、無線通信においても利用可能であると考え、移動端末のリソースを有効に活用した通信制御手法の検討を行う。本論文では、通信の上位層である TCP の輻輳制御に着目した。通常、TCP において、輻輳ウィンドウサイズは有線指向で輻輳制御アルゴリズムによりエンド・エンドで制御されているため、必ずしも周囲の状況に適した通信を行っているとは言えない。そこで、周囲の端末数に基づいて既存の TCP の輻輳ウィンドウサイズを制御する手法を提案し、提案手法を用いて改良した TCP を使用した場合のスループットおよび端末間の公平性をシミュレーションにより評価し、既存の TCP と通信性能の比較を行う。

キーワード コンテキスト, TCP, 輻輳制御方式, WLAN, モバイル通信, 公平性

## A Evaluation of TCP Congestion Window Setup Technique Considering the Fairness Based on Environmental Information in Wireless LAN.

Asuka IIO<sup>†</sup>, Taka MAENO<sup>††</sup>, Mineo TAKAI<sup>†††</sup>, and Masato OGUCHI<sup>†</sup>

<sup>†</sup> Ochanomizu University 2-1-1 Otsuka, Bunkyo, Tokyo, 112-8610, JAPAN

<sup>††</sup> Space-Time Engineering Japan, Inc. 3-27-3 Kanda sakumacho, Chiyoda, Tokyo, 101-0025, JAPAN

<sup>†††</sup> UCLA Computer Science Department 3803 Boelter Hall, Los Angeles, CA 90095-1596, USA

E-mail: <sup>†</sup>asuka-i@ogl.is.ocha.ac.jp, <sup>††</sup>tmaeno@spacetime-eng.com, <sup>†††</sup>mineo@cs.ucla.edu,

<sup>††††</sup>oguchi@computer.org

### 1. はじめに

現在、加速度センサやイメージセンサなどのセンシングデバイスは、スマートフォンや自動車のようなモバイル機器には不可欠な存在となっている。そのため、これらのモバイル機器は、センサ情報を収集することで、正確で精密な周囲の環境情報を

取得できる。この環境情報をコンテキストと呼ぶが、通常、コンテキストは、スマートフォンの場合には、ユーザの操作性の向上や高度なアプリケーションの実現等、また、自動車の場合には、車両安定制御や道路情報の把握等に利用されている。

一方、近年 WLAN ( Wireless Local Area Network ) による通信が一般化しているが、WLAN において通信に用いられる通信

パラメータは各々が独自のアルゴリズムに従って設定されており、その場の環境に応じた通信設定は行われない。したがって、コンテキストを取得可能な端末であってもリソースを効率的に利用した通信が行われておらず、必ずしも最適ではない状態で通信を行っているという問題がある。そこで、本研究では、コンテキストを無線通信パラメータ設定に利用し、各端末が周囲の状況に応じた最適な設定を行うことで、通信効率を向上させる手法の検討を行う。

本研究では、コンテキストの利用先として上位層の TCP (Transmission Control Protocol) に着目した。TCP は有線指向で研究・開発されており、無線通信においては、無線区間のフェージングやノイズによるパケットロスと輻輳を明確に見分ける方法がないため、最適な制御とは限らない。また、TCP において、確認応答 (ACK) を受けずに送信できるパケットの最大数である輻輳ウィンドウサイズ (cwnd) は輻輳制御アルゴリズムによりエンド・エンドで制御されているため、各端末は必ずしも周囲の状況に合った通信を行っているとは言えない。そこで、コンテキストとして周辺端末数および端末の移動状態を利用し、各端末が最適な cwnd を用いて通信を行うことで通信性能の向上を目指す。

本論文では、第 2 章で、TCP 輻輳制御について述べ、第 3 章で、コンテキストとして周辺端末数を利用することで、端末間で帯域を均等に分けて通信を行う輻輳制御手法を提案する。第 4 章では、提案する制御手法を用いて通信を行った場合と、輻輳制御アルゴリズムにより cwnd が制御される一般的な TCP を用いて通信を行った場合において、それぞれの基礎性能をシミュレーション評価し、第 5 章では、数種類の TCP が混在する環境での提案する制御手法の振舞いの評価を行う。また、第 6 章では、既存のロスベース方式の TCP の輻輳制御へコンテキストを利用する手法の提案を行い、第 7 章では、提案手法の評価を行う。最後に、第 8 章で本研究の結論を述べ、第 9 章で本研究の課題を述べる。

## 2. TCP 輻輳制御

### 2.1 輻輳制御の概要

輻輳とは、通信トラフィックが帯域やネットワーク機器の許容量を上回ることによって、パケット損失率が高くなる状態である。パケットロスが起これば、損失したパケットを再送するため、通信スループットは低下してしまう。TCP における輻輳制御とは、データ送信側が帯域の混雑具合を予測し、輻輳を未然に防ぐための制御である。TCP では、輻輳ウィンドウという、転送先からの ACK を受信することなく一度に連続して送り出せる最大のセグメント数を示すパラメータの大きさを調節することで、伝送レートを下げ、輻輳が起きないように制御している [1]。

特に高遅延環境においては、通信スループットへの影響が大きく、cwnd が大きくなれば、連続して送り出せるセグメント数も多くなり、通信スループットは高くなる。つまり、より賢い制御を行い輻輳が起きない範囲で cwnd を大きく保つことができれば、スループットは向上する。

### 2.2 現行の輻輳制御アルゴリズム

これまでに研究開発が行われ実装された TCP の輻輳制御アルゴリズムには、TCP Tahoe [2] [3] を修正した TCP Reno [4] や TCP CUBIC [5] [6] に代表されるロスベース方式、TCP Vegas [7] に代表される遅延ベース方式があり、近年では、それらを組み合わせ合わせたハイブリッド方式の提案も盛んに行われている。

ロスベース方式とは、正常な確認応答を受信したら cwnd を増加させ、パケットロスを検出すると輻輳とみなして cwnd を減少させる制御である。有線通信においては、通信エラーは一般に経路の混雑によるルータのバッファ溢れを示すため、この手法は効果的であるが、無線通信のように輻輳していなくてもパケットロスが発生する環境下においては、ノイズによる通信不良も帯域の混雑とみなしてしまうため、帯域を大きく余らせてしまうことがある。

遅延ベース方式とは、データ転送中に計測された RTT の実測値と理論値を比較し、輻輳が起こる前に適切な cwnd に調節する制御手法であり、ロスベースよりもパケットロスが少なく、帯域を有効に使用することが可能である。しかし、高遅延環境で、ロスベース方式と遅延ベース方式が混在する環境下では、パケットロスを検知するまで cwnd を増加させバッファを使い切ることができるロスベース方式に対し、遅延ベース方式は cwnd をほとんど増加できないため、スループットが低下してしまう [8] [9]。

## 3. 提案方式

### 3.1 コンテキストを利用した通信制御方式

本稿では、コンテキストを利用して輻輳制御を行う TCP を提案する。

コンテキストとは「実体を特徴づけることのできるあらゆる情報」と定義されている [10]。ここでの実体とは、ユーザとアプリケーションの間のやり取りに関連していると考えられる人、場所、物のことを指しており、これにはユーザやアプリケーション自体も含まれている。コンテキストをこのように定義することで、アプリケーションを開発する際に必要となる、実体のその時の状況がわかるような情報の列挙がより容易になる。そして、コンテキストは、さまざまなアプリケーションで利用することができる [11]。本研究では、コンテキストが通信の性能改善に有効であると考え、無線通信のパラメータ設定において利用する。

図 1 は、本稿で提案する制御方式の概要で、センサ情報の取得から無線通信へのコンテキスト利用の流れを表している。

図 1 のように、無線通信を行う端末が、自身に搭載された無数のセンサから多くの周辺情報を取得する。そして、取得した情報からコンテキストを生成し、そのコンテキストを無線通信の上位層である TCP のパラメータ設定に利用することで、各端末が周囲の状況に応じた通信設定を行う。

### 3.2 コンテキストに基づく通信可能帯域の均等振り分け

本稿では、コンテキストとして周辺端末数を利用することで、式 (1) のように、帯域幅と往復遅延時間 (RTT: Round Trip Time) の積である帯域幅遅延積 [12] を、同じ AP (Access Point)

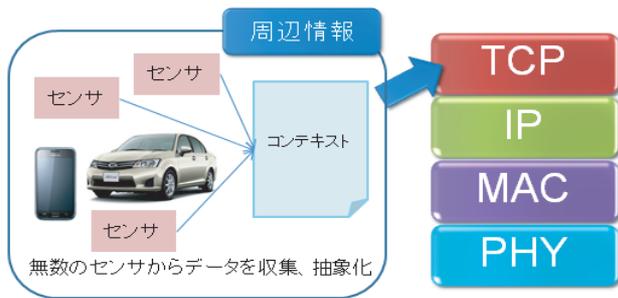


図1 センサから取得したコンテキストを通信の上位層へ利用

を使用して通信を行う端末同士で均等に分けて通信を行う状態を目指す。これにより、効率よく通信可能帯域を使い切ることができ、端末間の公平性も保つことができる。そこで、本研究では、このようにして算出した1台あたりの通信可能帯域を、これよりも下げる必要のない値、つまり、輻輳制御における cwnd の下限値とみなす。

$$\begin{cases} \text{帯域幅遅延積 [bit]} = \text{帯域幅 [bit/s]} \times \text{RTT[s]} \\ \text{cwnd 下限値} = \text{帯域幅遅延積} \div \text{端末数} \end{cases} \quad (1)$$

## 4. 基礎実験 I

### 4.1 概要

AP から 100m の位置にある 30 台の端末が、AP を経由し、サーバにパケットを送信する(図 2)。ここで、端末は走行中の自動車を想定し、使用したネットワークシミュレータ Scenargie [13] の Fading 設定により、全ての端末が時速 30km で移動しているものとした。

このモデルにおいて、提案するコンテキストを利用する制御手法(以下、TCP ocha)、および、比較対象として既存の TCP である TCP NewReno, TCP CUBIC, TCP Vegas の 4 種類の TCP を用い、それぞれの性能を評価するための基礎実験を行った。シミュレーションパラメータを表 1 に示す。

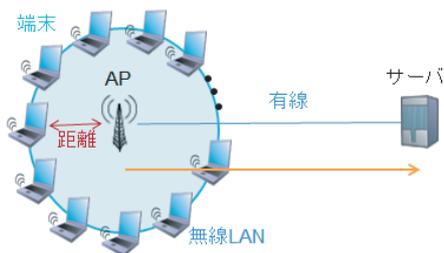


図2 ネットワークトポロジー

### 4.2 シミュレーション結果

まず、各 TCP のトータルスループットを図 3 に示す。

次に、端末間の公平性を測る指標として、Jain の Fairness Index を用いる [14]。FI の値 ( $f_i$ ) は、式 (2) のように定義され、0.0 から 1.0 で表し、1.0 に近い程、公平な値であることを示す。

ネットワークシミュレータ	Scenargie ver1.6
無線規格	IEEE802.11g
電波伝搬モデル	TwoRayGround
フェージングモデル	RAYLEIGH
ドップラ速度	30km/h
シミュレーション時間	60s
伝送レート	6Mbps
RTT	200msec
端末と AP 間の距離	200m
端末数	30 台

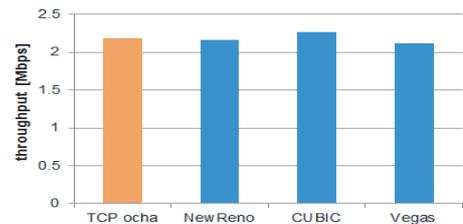


図3 トータルスループット

$$f_i = \frac{(\sum_{i=1}^k x_i)^2}{k \sum_{i=1}^k x_i^2} \quad (1 \leq i \leq k) \quad (2)$$

本実験結果の Fairness Index を図 4 に示す。

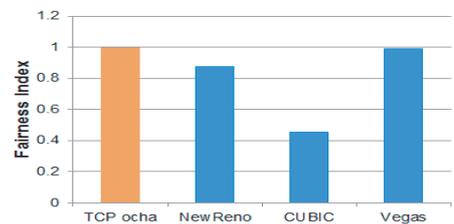


図4 Fairness Index

図 3 から、どの TCP も同程度のスループットが得られることがわかった。一方で、図 4 から、端末間の公平性には TCP による差が見られ、TCP ocha では 1.0 に非常に近い値となり、公平性では最も優れているという結果が得られた。

## 5. 基礎実験 II

### 5.1 概要

本節では、数種類の TCP が混在する環境での各 TCP の性能評価を行う。基礎実験 I と同様のトポロジを使用し、AP から 200m の位置にある 40 台の端末が、AP を経由し、サーバにパケットを送信する(図 2)。端末数以外のシミュレーションパラメータは表 1 と同じである。端末 40 台のうち、提案したコンテキストを利用し通信制御を行う TCP ocha、および、既存の TCP である TCP NewReno, TCP CUBIC, TCP Vegas をそれぞれ 10 台ずつとし、4 種類の TCP が混在する環境で実験を行った。ここで、その他のシミュレーションパラメータは表 1 と同様であり、時速 30km で移動しているものとする。

## 5.2 シミュレーション結果と考察

図5に、各TCPのスループットおよび全体のスループットを示す。また、図6に各TCPのFairness Indexを示す。

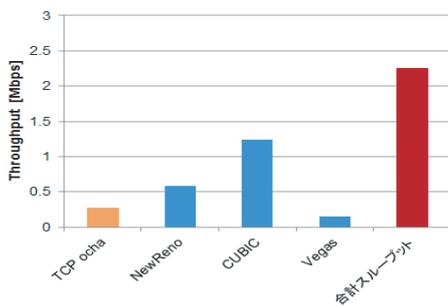


図5 各TCPのスループット

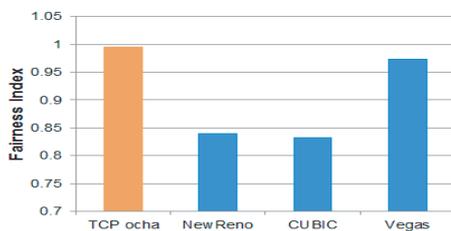


図6 各TCPのFairness Index

一般的に、RTTの実測値と理論値を比較して、回線の混雑状況を判断してcwnd制御をする遅延ベース方式を用いたTCP Vegasでは、ロスベース方式の輻輳制御を行う他のTCPとの競合で、スループットが低下することが知られている。ロスベース方式のTCPと遅延ベース方式のTCPの環境下では、ロスベース方式のTCPがパケットロスを検知するまでcwndを増加させてしまうことから、ルータなどボトルネックとなる箇所のバッファ溢れが発生し、これにより、接続のRTTが増加するため、遅延ベース方式であるTCP Vegasはcwndを減少させてしまい、スループットが低下する。

TCP ochaにおいても、TCP Vegasと同様、端末間の公平性は高く保たれている一方で、スループットが低下している。これは、通信可能帯域を均等に使い切るというcwnd制御を行ったため、理想的な通信環境である場合には良い性能を出すのが、本実験のように、振舞いの異なる数種類のTCPと混合した場合には、TCP CUBICのような他のアグレッシブなTCPに、帯域を奪われてしまい、TCP ochaが適切な制御をできなかったものと推測される。すなわち、cwndを固定することで、通信量を自ら制御してしまう結果となったため、期待通りの結果が得られなかったと考えられる。しかし、同じTCPを使用する端末間の公平性においては、非常に高い効果があることがわかった。

## 6. 既存のTCPへのコンテキストを利用

ロスベース方式のTCPであるTCP NewRenoおよびTCP CUBICにコンテキストを利用することで、性能の改良手法を提案する。すなわち、第4章では、cwndを下限値とみなした値に固定して振舞いを評価したが、本章では、実際に、これらのTCP

にコンテキストと周辺端末数を利用することで、cwndに初期値と下限値を設置する。これにより、それぞれの端末が、自分の使用可能帯域を最低限使い切り、尚且つ、帯域が余っている場合には通信可能な端末が使い切りトータルスループットを向上、また、通信開始からcwndをほとんど増加できていない端末をなくすことで公平性の向上をそれぞれ見込むことができる。

## 7. 検証実験

### 7.1 概要

実験Iと同様のトポロジを使用し、APから200mの位置にある複数の端末が、APを経由し、サーバにパケットを送信する(図2)。端末数を10台から50台へと増加させるが、それ以外のパラメータは表1と同じである。本実験では、ロスベース方式のTCPとしてTCP NewRenoとTCP CUBICを用い、通常のTCPと改良したTCP、および、それらを半数ずつ混合した場合の合計3パターンの評価を行う。

### 7.2 結果

図7、図8にTCP NewRenoのトータルスループットとFairness Indexを示す。また、図9、図10に通常TCP NewRenoと改良TCP NewRenoを半数ずつ混合した場合のスループットの内訳とFairness Indexを示す。Fairness Indexは、TCP別の結果をそれぞれ表したものと、2つのTCPを合わせた結果を表示している。

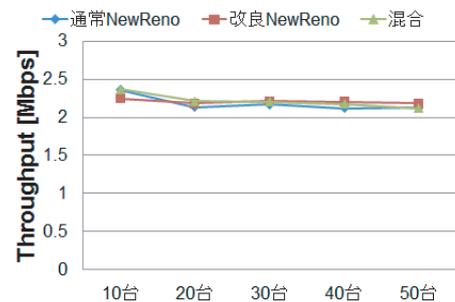


図7 NewRenoのトータルスループット

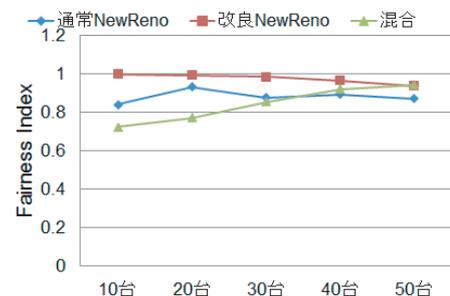


図8 NewRenoのFairness Index

図7、図8から、通常TCP NewRenoと改良TCP NewRenoを比較すると、トータルスループットはほとんど変わらないが、Fairness Indexは改良したTCP NewRenoがより1に近づき、公平性が改善された。また、混合時における端末数が小さい場合のFairness Indexは、TCPの改良有無による性能差が表れてし

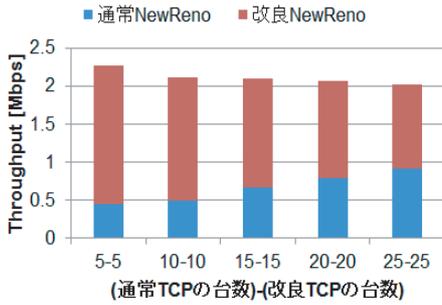


図9 NewReno 混合時のスループット内訳

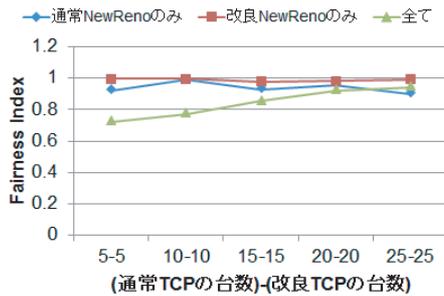


図10 NewReno 混合時の Fairness Index

まい、通常 NewReno のみよりも低くなった。しかし、図 10 から通常 TCP NewReno と改良 TCP NewReno の混合時の実験結果を見ると、Fairness Index は改良 TCP NewReno が、より 1 に近いが、通常 TCP と改良 TCP のどちらも 1 に近いので、同じ TCP を用いる端末間では、端末間の公平性が保たれていることがわかった。そして、図 9 では、端末数の増加につれ、通常 TCP と改良 TCP の差が減少していることがわかる。これらの結果から、端末数が増加すると端末数が大きい場合には、スループットがほぼフェアシェアであり、同じ TCP を使用している端末間の公平性も高いため、極めて理想的な状態で通信を行っているといえる。

続いて、図 11、図 12 に TCP CUBIC のトータルスループットと Fairness Index を示す。また、図 13、図 14 に通常 TCP CUBIC と改良 TCP CUBIC を半数ずつ混合した場合のスループットの内訳と Fairness Index を示す。図 11、図 12 を見ると、通常の

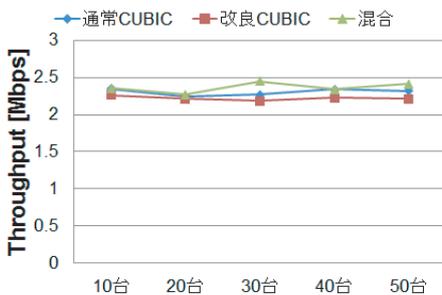


図11 CUBIC のトータルスループット

TCP CUBIC は、トータルスループットが高いが、Fairness Index が大きく低下しているため、帯域は効率よく埋まっているが公平な通信を行っているとは言えない。一方で、改良した TCP

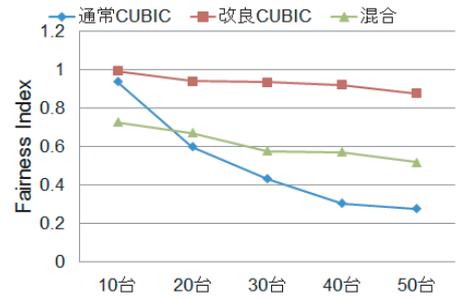


図12 CUBIC の Fairness Index

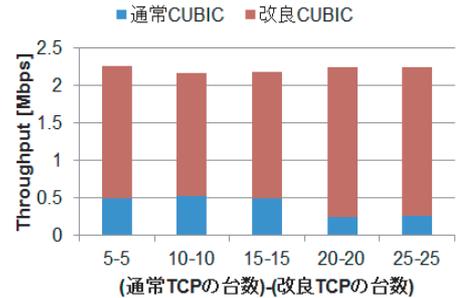


図13 CUBIC 混合時のスループット内訳

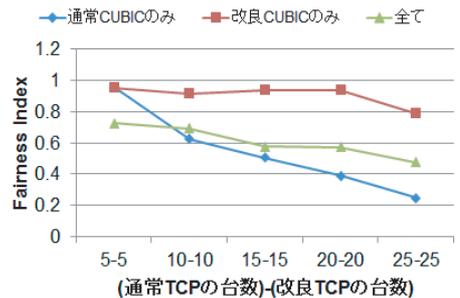


図14 CUBIC 混合時の Fairness Index

CUBIC は通常 TCP CUBIC と比較してトータルスループットは同程度出ており、Fairness Index は大きく改善した。Fairness Index は、50 台の時には、約 3.2 倍に改善されていることがわかった。また、図 13 から、改良 TCP CUBIC は通常 TCP CUBIC よりもアグレッシブなため、実環境では、他の TCP と混在した環境で勝ちすぎる可能性もあると推測されるが、図 14 から、通常と改良を混在させた場合においても、改良 TCP CUBIC では、通常 TCP CUBIC よりも端末数に関わらず同じ改良 TCP を使用する端末間の公平性が保たれているため、コンテキストを利用した TCP の改良が、効果的であると言える。

## 8. まとめ

WLAN 通信環境において、コンテキストを利用して周辺の端末間で帯域を均等に分けるような cwnd を下限値とみなし、各端末がその cwnd に固定して通信を行う TCP の制御手法を提案し、その基礎的な性能を評価した。また、cwnd に下限値を設定することで既存のロスベース方式の TCP を改良し、その性能を評価した。TCP NewReno および TCP CUBIC では、輻輳制御に初期値および下限値を設定することで、同じ TCP を使

用する端末間の公平性が大きく改善されたため、輻輳制御へ周辺情報を利用することの有用性が示された。

## 9. 今後の課題

コンテキストを利用した TCP を使用することで、既存の TCP とのフレンドリネスを保ちつつ、端末間の公平性の高い通信ができる可能性が大いにあると考えられるため、今後は、数種類の TCP が混在する環境における改良 TCP の振る舞い評価を行いたい。

## 謝 辞

本研究を進めるにあたり、株式会社トヨタ IT 開発センターの Onur Altintas 氏、松本真紀子氏に大変有用なアドバイスをいただきました。深く感謝いたします。

## 文 献

- [1] 西川 佳史, TCP 詳説 (1999)  
<https://www.nic.ad.jp/ja/materials/iw/1999/notes/C3.PDF>
- [2] V. Jacobson, "Congestion avoidance and control," Proceedings of SIGCOMM '88 Symposium proceedings on Communications architectures and protocols, Pages 314-329
- [3] Stevens, W.R.: "TCP Slow Start, Congestion Avoidance, Fast Retransmit, and Fast Recovery Algorithms," The Internet Society (RFC2001), 1997.
- [4] Kevin Fall, Sally Floyd, "Simulation-based comparisons of Tahoe, Reno and SACK TCP," Newsletter ACM SIGCOMM Computer Communication Review Homepage archive Volume 26 Issue 3, Pages 5-21, July 1996
- [5] Sangtae Ha, Injong Rhee and Lisong Xu, "CUBIC: A New TCP-Friendly High-Speed TCP Variant," Proceedings of SIGOPS Operating Systems Review, 42(5):64-74, 2008.
- [6] L. Xu, K. Harfoush, and I. Rhee, "Binary Increase Congestion Control for Fast, Long Distance Networks," Proceedings of Tech. Report, Computer Science Department, NC State University, 2003.
- [7] Brakmo, Lawrence S., Sean W. O'malley, and Larry L. Peterson. "TCP Vegas: New techniques for congestion detection and avoidance," Vol. 24, no. 4. ACM, 1994.
- [8] La, Richard J., Jean Walrand, and Venkatachalam Anantharam, "Issues in TCP vegas," Electronics Research Laboratory, College of Engineering, University of California, 1999.
- [9] 倉田 謙二, 長谷川 剛, 村田 正幸, "TCP Reno と TCPVegas の混在環境における公平性の評価", 電子情報通信学会技術報告 SSE99-98, pp.67-72, November 1999
- [10] Day, Anind K., "Understanding and Using Context"(2001). Human-Computer Interaction institute.Paper 34.  
<http://repository.cmu.edu/hcii/34>
- [11] Daniel Salder, Anind K.Dey and Gregory D. Abowd, "The Context Toolkit: Aiding the Development of Context-Enabled Applications", In Proceedings of CHI'99, Pittsburgh, PA, May, 1999, ACM Press.
- [12] W.Richard Stevens 著, 橋 康雄, 井上 尚司訳 (2000)『詳解 TCP/IP Vol.1 プロトコル』ピアソンエデュケーション 327pp.
- [13] Scenargie Visual Lab :  
<http://www.spacetime-eng.com/>
- [14] D.-M. Chiu and R. Jain, "Analysis of the increase and decrease algorithms for congestion avoidance in computer networks," Computer Networks and ISDN Systems, vol. 17, pp. 1-14, 1989.
- [15] 松本真紀子, 小口正人, "無線端末の周辺情報を利用した MAC 層におけるマルチプルアクセス手法の一検討", 電子情報通信学会ネットワークシステム研究会, NS2011-106, pp.13-18, 2011 年 11 月