社団法人　電子情報通信学会
THE INSTITUTE OF ELECTRONICS,
INFORMATION AND COMMUNICATION ENGINEERS

信学技報
TECHNICAL REPORT OF IEICE.

大規模ネットワーク性能評価のための フローレベルシミュレータの設計と実装

浅井　亮太† 　作元　雄輔†† 　大崎　博之†† 　今瀬　真††

† 大阪大学 基礎工学部情報科学科
〒 565-0871 大阪府吹田市山田丘 1-5
†† 大阪大学 大学院情報科学研究科
〒 565-0871 大阪府吹田市山田丘 1-5
E-mail: †r-asai@ics.es.osaka-u.ac.jp, ††{y-sakumt,oosaki,imase}@ist.osaka-u.ac.jp

あらまし　本稿では、大規模ネットワークの性能評価のための、フローレベルシミュレータ FSIM (Fluid-based SIMulator) を提案するとともに、実装した FSIM を用いてその有効性を検証する。フローレベルシミュレータ FSIM は、従来のフローレベルシミュレータよりも高精度かつ高速なシミュレーションが可能であるという特徴を持つ。シミュレーション精度の向上のため、より高精度な流体近似モデルを採用する。また、シミュレーション速度の向上のため、フローレベルシミュレーションで用いられる、微分方程式の数値計算アルゴリズムを高速化する。さらに、フローレベルシミュレータ FSIM は、既存のネットワーク性能評価ツールとの高い親和性を実現する。本稿では、実装したフローレベルシミュレータ FSIM を用いた実験を行い、FSIM のシミュレーション速度を評価する。その結果、フローレベルシミュレータ FSIM は、従来のフローレベルシミュレータよりも倍以上高速なシミュレーションが可能であることを示す。
キーワード　大規模ネットワーク、性能評価、シミュレーション、フローレベルシミュレータ、流体近似モデル

Design and Implementation of Flow-Level Simulator for Performance Evaluation of Large Scale Networks

Ryouta ASAI†, Yusuke SAKUMOTO††, Hiroyuki OHSAKI††, and Makoto IMASE††

† School of Engineering Science, Osaka University
1-5 Yamadaoka, Suita, Osaka, 565-0871 Japan
†† Graduate School of Information Science and Technology, Osaka University
1-5 Yamadaoka, Suita, Osaka, 565-0871 Japan
E-mail: †r-asai@ics.es.osaka-u.ac.jp, ††{y-sakumt,oosaki,imase}@ist.osaka-u.ac.jp

Abstract　In this paper, we propose a flow-level simulator called FSIM (Fluid-based SIMulator) for performance evaluation of large-scale networks, and verify its effectiveness using our FSIM implementation. The notable feature of our flow-level simulator FSIM is its more accurate and much faster simulation than conventional flow-level simulators. For improving simulation accuracy, our flow-level simulator FSIM utilizes accurate fluid-flow models. Moreover, for accelerating simulation speed, our flow-level simulator FSIM adopts an adaptive numerical computation algorithm of ordinary differential equations. Another notable feature of our flow-level simulator FSIM is its compatibility with other network performance analysis tools. In this paper, we evaluate the simulation speed of our flow-level simulator FSIM using our FSIM implementation. Consequently, we show that the simulation speed of our flow-level simulator FSIM is as twice as that of a conventional flow-level simulator.
Key words　Large Scale Network, Performance Evaluation, Simulation, Flow-Level Simulator, Fluid-Flow Model

1　はじめに

近年、ネットワークの大規模化が急速に進んでいる。インターネットの爆発的な普及により、ネットワークに接続されるノード数が指数関数的に増加している [1]。このようなネットワークの大規模化に伴い、ネットワーク全体の挙動を理解す

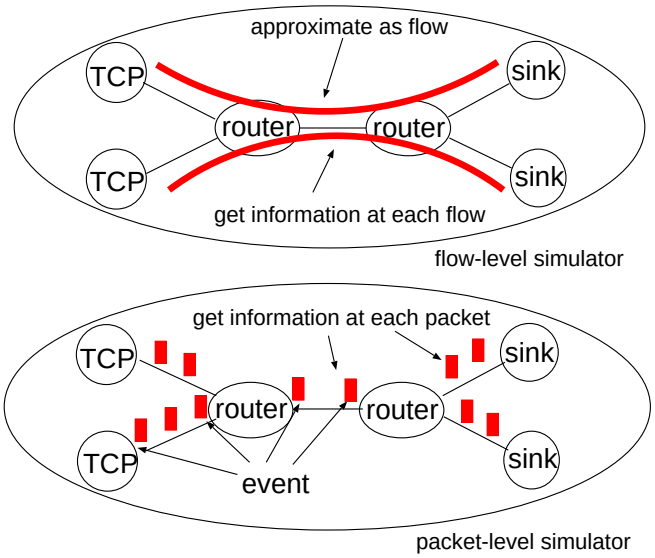

図 1: パケットレベルシミュレータとフローレベルシミュレータ
Fig. 1　Packet-level simulator and flow-level simulator

ることが困難になってきている。そのため、特に大規模ネットワークの性能評価手法に対する要求が高まっている [2]。

ネットワークの性能評価手法として、数学的解析・シミュレーション・実験の大きく 3 種類の手法が存在する [3]。

数学的解析は、対象とするネットワークを数学的にモデル化することによってその性能を解析する手法であり、比較的小規模なネットワークの性能評価に適している。数学的解析による大規模ネットワークの性能評価手法の研究も近年活発に行われている [4, 5] が、大規模ネットワークの解析には多くの近似が必要であるため、性能評価結果の精度といった点で問題がある。

シミュレーションは、対象とするネットワークの抽象的なモデルを計算機上に構築し、ネットワークの挙動を模擬することによってその性能を評価する手法である [3]。シミュレーションは、数学的解析よりも、より複雑なネットワークの性能評価が可能である。

実験は実機を用いて実際にネットワークを構築することにより性能評価を行なう手法である [3]。詳細な性能評価が可能であるが、実験には多大なコストを要するという問題がある。特に、大規模ネットワークの実験を行うためには、多数の計算機とネットワーク機器を用いて実験環境を構築する必要があり、膨大なコストが必要である。このため、大規模ネットワークの実験は現実的ではない。

得られる精度と性能評価に必要なコストの観点から、大規模ネットワークの性能評価手法として、特にシミュレーション手法への期待が高まっている。

ネットワークシミュレータには、パケットレベルシミュレータおよびフローレベルシミュレータという、利用するモデルの抽象度が異なる二種類が存在する [6]。

パケットレベルシミュレータとは、ネットワークの挙動をパケット単位でシミュレートするシミュレータである [7]。例えば、パケットレベルシミュレータでは、ルータへのパケット到着や、ルータからのパケット離脱を、パケット単位でシミュレートする。パケットレベルシミュレータは、現在主流となっているシミュレータである。

パケットレベルシミュレータの利点として、フローレベルシミュレータよりも、精度の高いシミュレーションが可能であるという点が挙げられる [8]。パケットレベルシミュレータでは、個々のパケットの挙動をシミュレートするため、パケットごとの性能指標の計測も可能である。

ただし、パケットレベルシミュレータの欠点として、ネットワークの規模が大きくなると、シミュレーションに必要な計算量およびメモリ量が共に増大し、大規模なネットワークのシミュレーションが困難になるという点が挙げられる [8]。大規模ネットワークのパケットレベルシミュレーションを可能にするための手法も提案されている [9] が、まだまだ多くの問題を解決しなければならない。

一方、フローレベルシミュレータとは、ネットワークの挙動をパケットの流れ (フロー) 単位でシミュレートするシミュレータである [6]。例えば、フローレベルシミュレータでは、ルータへの単位時間当りのパケット到着レートや、ルータからの単位時間当りのパケット離脱レートを、フロー単位でシミュレートする。

フローレベルシミュレータの利点として、パケットレベルシミュレータよりも、きわめて高速なシミュレーションが可能であることが挙げられる。複数のパケットをフローとして近似しているため、転送されるパケット数が膨大になるような、大規模ネットワークや超高速ネットワークのシミュレーションも実用的な速度で実行することが可能である [10]。

フローレベルシミュレータの欠点として、パケットレベルシミュレータよりも、シミュレーション結果の精度が低いという点が挙げられる。また、個々のパケットの挙動をシミュレートしないために、当然のことながらパケットごとの性能指標は計測できない。ただし、大規模ネットワークのシミュレーションでは、一般にフローレベルやアプリケーションレベルの性能指標が重要になるため、このことは実用上大きな問題にはならないと考えられる。

本稿では、大規模ネットワークの性能評価のための、フローレベルシミュレータ FSIM (Fluid-based SIMulator) を提案するとともに、実装した FSIM を用いてその有効性を検証する。

フローレベルシミュレータ FSIM は、従来のフローレベルシミュレータ [11, 12] よりも高精度かつ高速なシミュレーションが可能であるという特徴を持つ。シミュレーション精度の向上のため、より高精度な流体近似モデル [13] を採用する。また、シミュレーション速度の向上のため、フローレベルシミュレーションで用いられる、微分方程式の数値計算アルゴリズムを高速化する。さらに、フローレベルシミュレータ FSIM は、既存のネットワーク性能評価ツールとの高い親和性を実現する。具体的には、代表的なパケットレベルシミュレータである ns-2 [7] 形式のファイルの入出力が可能である。

本稿では、実装したフローレベルシミュレータ FSIM を用い

た実験を行い、FSIM のシミュレーション速度を評価する。その結果、フローレベルシミュレータ FSIM は、従来のフローレベルシミュレータと比較して、シミュレーション実行時間を約 65% に短縮できることを示す。

本稿の構成は以下の通りである。まず、2 章において関連研究を紹介する。3 章において、提案するフローレベルシミュレータ FSIM を説明する。特に、フローレベルシミュレータ FSIM が使用する流体近似モデルおよび微分方程式の数値計算アルゴリズムを詳細に説明する。4 章において、実装したフローレベルシミュレータ FSIM を用いた実験により、その有効性を検証する。最後に、5 章において、本稿のまとめと今後の課題について述べる。

2 関 連 研 究

文献 [11] では、TCP/RED の流体近似モデルを用いた、大規模ネットワークのシミュレーション手法が提案されている。流体近似モデルから得られる微分方程式を数値的に解くことにより、フローレベルシミュレーションを実現している。しかし、固定のステップ時間ごとに状態を更新するという単純な微分方程式の数値計算アルゴリズムを採用している。このため、ネットワークの状態がほとんど変化しない場合でも、ネットワークの状態を更新しており、シミュレーション速度低下の原因となっていると考えられる。さらに、提案されている TCP の流体近似モデルでは、TCP のタイムアウト機構がモデル化されていないため、流体近似モデルの精度が十分ではない。

文献 [13] では TCP のタイムアウト機構をもモデル化した、TCP/RED の流体近似モデルが構築されている。シミュレーション結果と解析結果を比較することにより、文献 [13] の流体近似モデルは、文献 [11] の流体近似モデルよりも高精度であることが示されている。ここで提案されているモデル化手法では、モデルの入出力が全てパケット転送レートで統一されている。そのため、それぞれの流体近似モデルを簡単に接続することが可能であり、大規模ネットワークのモデル化が容易に行える。本稿では、文献 [13] の流体近似モデルおよびモデル化手法を用いることにより、高精度なフローレベルシミュレータを実現する。

文献 [14] では、複数の流体近似モデルを一定間隔ごとに切り替えるというハイブリッドモデルが提案されている。TCP の各動作フェーズ (スロースタートフェーズ、輻輳回避フェーズ) に対応するモデルを、ネットワークの状態にあわせて切り替えることにより、モデルの精度向上を図っている。また、流体近似モデルから得られる微分方程式を、数値的に解くことによりフローレベルシミュレーションを実現している。しかし、文献 [11] と同様に、固定のステップ時間ごとに状態を更新するという、単純な微分方程式の数値計算アルゴリズムを採用している。このため、ネットワークの状態がほとんど変化しない場合でも、ネットワークの状態を更新しており、シミュレーション速度低下の原因となっていると考えられる。

3 フローレベルシミュレータ FSIM

フローレベルシミュレータ FSIM は、従来のフローレベルシミュレータよりも高精度かつ高速なシミュレーションが可能であるという特徴を持つ。シミュレーション精度の向上のため、より高精度な流体近似モデル [13] を採用する。また、シミュレーション速度の向上のため、フローレベルシミュレーションで用いられる、微分方程式の数値計算アルゴリズムを高速化する。さらに、フローレベルシミュレータ FSIM は、既存のネットワーク性能評価ツールとの高い親和性を実現する。具体的には、代表的なパケットレベルシミュレータである ns-2 [7] 形式のファイルの入出力が可能である。

以下では、フローレベルシミュレータ FSIM が利用する流体近似モデル、微分方程式の数値計算アルゴリズム、既存のネットワーク性能評価ツールとの互換性について順番に説明する。

3.1 流体近似モデル

トランスポート層プロトコルの流体近似モデルには、文献 [13] で提案されている TCP のモデルを用いる (図 2)。流体近似モデルで用いられる記号の定義を表 1 に示す。

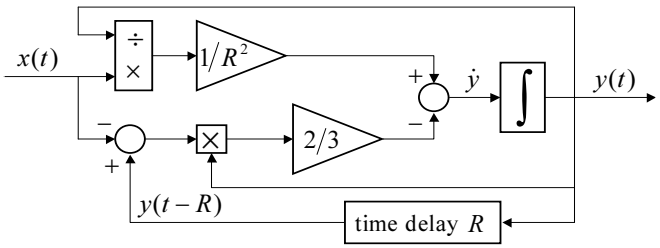

図 2: TCP の輻輳制御機構のモデルのブロック図
Fig. 2 Block diagram of congestion control mechanism of TCP

表 1 記号の定義

$x(t)$	モデルへの入力 (パケット転送レート)
$y(t)$	モデルからの出力 (パケット転送レート)
R	TCP のラウンドトリップ時間
$w(t)$	TCP のウィンドウサイズ
$p_{TO}(t)$	TCP のタイムアウト発生率
min_{th}	RED ルータの最小しきい値
max_{th}	RED ルータの最大しきい値
max_p	RED ルータの最大パケット棄却率
α	RED ルータの指数平均の重み
$c(t)$	RED ルータの処理速度
$b(t)$	RED ルータの現在キュー長
$r(t)$	RED ルータの平均キュー長
$p_b(t)$	RED ルータのパケットマーキング確率
$p(t)$	RED ルータのパケット棄却率

送信側ホストへの入力は ACK パケットの到着レート $x(t)$ であり、出力はデータパケットの転送レート $y(t)$ である。

$$\begin{aligned}\dot{y} &= \frac{x(t)}{y(t)R^2} - \frac{2}{3}y(t)z(t)\{1-p_{TO}(t)\} \\ &\quad - \left\{\frac{4}{3}y(t) - \frac{1}{R}\right\} z(t)p_{TO}(t)\end{aligned} \tag{1}$$

ここで、$p_{TO}(t)$ は TCP がタイムアウトによってパケット棄却を検出する確率であり、$p_{TO}(t) \simeq min(1, 3/w(t))$ である。また、$z(t) = y(t-R) - x(t)$ である

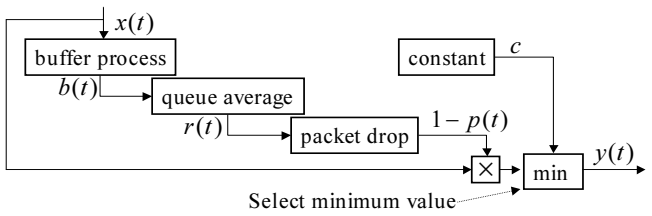

図 3: RED ルータのモデルのブロック図
Fig. 3 Block diagram of RED router

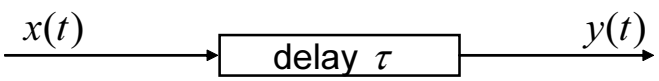

図 4: リンクの伝搬遅延のモデルのブロック図
Fig. 4 Block diagram of link propagation delay

ルータの流体近似モデルには、文献 [13] で提案されている RED ルータのモデルを利用する (図 3)。ルータへの入力は各リンクからのパケット到着レートの和 $x(t)$、出力はパケット転送レート $y(t)$ である。

$$\dot{b} = \begin{cases} x(t) - c(t) & \text{if } b(t) > 0 \\ (x(t) - c(t))^+ & \text{if } b(t) = 0 \end{cases} \tag{2}$$

$$\dot{r} = -\alpha\, c(t)(r(t) - b(t)) \tag{3}$$

$$p(t) = \frac{2p_b(t)}{1 + p_b(t)} \tag{4}$$

$$p_b(t) = \begin{cases} 0 & \text{if } r(t) < min_{th} \\ \dfrac{max_p}{max_{th} - min_{th}}(r(t) - min_{th}) & \text{if } min_{th} \leqq r(t) < max_{th} \\ \dfrac{1 - max_p}{max_{th}} r(t) - (1 - 2max_p) & \text{if } max_{th} \leqq r(t) < 2max_{th} \\ 1 & \text{if } r(t) \geqq 2max_{th} \end{cases} \tag{5}$$

ここで、$(x)^+ \equiv \max(x, 0)$ である。

リンクの流体近似モデルには、文献 [13] で提案されているリンクの伝播遅延のモデルを利用する (図 4)。リンクへの入力はパケット到着レート $x(t)$、出力はパケット転送レート $y(t)$ である。

$$y(t) = x(t - \tau) \tag{6}$$

ここで τ はリンクの伝搬遅延である。

複数のノードやリンクから構成されるネットワーク全体の流体近似モデルを得るためには、トランスポート層プロトコル、ルータ、リンクの流体近似モデルを相互に接続する必要がある。

ルータが複数の入力リンクを持つ場合は、ルータへの入力を各入力リンクからのフローの合流としてモデル化する。フローの合流は各リンクからの出力レートの和として与えられる。それぞれのフローの転送レートを $x_i(t)(1 \leqq i \leqq N)$、転送レートの和を $y(t)$ とすると、次式が成立する。

$$y(t) = \sum_{i=1}^{N} x_i(t) \tag{7}$$

ルータが複数の出力リンクを持つ場合、ルータからの出力レートを各フローに分配することによりモデル化する。フローの分配は、1 つのパケット転送レートを N 個に分配することにより記述できる。ここで、N は出力リンクの本数である。分配前のフローの転送レートを $x(t)$、分配後のフローの転送レートを $y_i(t)(1 \leqq i \leqq N)$、それぞれのフローの分配比率を $f_i(t)$ とすると、次式が成立する。

$$y_i(t) = f_i(t)x(t) \tag{8}$$

3.2 微分方程式の数値計算アルゴリズム

3. 1 章で説明した流体近似モデルでは、TCP のパケット転送レート $y(t)$、RED ルータの現在キュー長 $b(t)$ および平均キュー長 $r(t)$ がネットワークの状態変数となる。そのため、TCP のパケット転送レート $y(t)$、RED ルータの現在キュー長 $b(t)$ および平均キュー長 $r(t)$ をネットワークの状態ベクトル $\mathbf{z}(t)$ として、以下のように表記する。

$$\mathbf{z}(t) = \begin{pmatrix} y_1(t) \\ \vdots \\ y_M(t) \\ b_1(t) \\ \vdots \\ b_N(t) \\ r_1(t) \\ \vdots \\ r_N(t) \end{pmatrix} \tag{9}$$

ここで、M はネットワーク中の TCP フロー数、N は RED ルータの数である。

上記の状態ベクトル $\mathbf{z}(t)$ を用いれば、ステップ時間 Δ 後の状態は近似的に次式で与えられる。

$$\mathbf{z}(t + \Delta) \simeq \mathbf{z}(t) + \dot{\mathbf{f}}(t, \mathbf{z}(t))\Delta \tag{10}$$

ここで、$\dot{\mathbf{f}}$ は、流体近似モデル (式 (1)–(3)) から求められる。

微分方程式の数値計算アルゴリズムを用いれば、ある初期条件から開始した時の、ネットワーク状態の変化を求めることができる。フローレベルシミュレータ FSIM では、式 (10) の数値解法として、常微分方程式の解法として広く用いられている Runge-Kutta 法を用いる [15] 。また、シミュレーション高速化のために、ネットワークの状態変化の度合によってステップ時間を可変にするという、Runge-Kutta 法に対する適応刻み幅制御 [15] を用いる。

これにより、ネットワーク状態の変化量が大きい場合には、計算結果に含まれる誤差が大きくなるため、ステップ時間を小さくする。一方、ネットワーク状態の変化量が小さい場合には、

ステップ時間を大きくすることにより、計算結果に含まれる精度を一定に保ちながら計算量を削減する。これにより、一定以上の計算精度を保ちながら、シミュレーションに必要な計算量の大幅な削減が可能になる。

ただし、式 (1) より、TCP のパケット転送レートを計算するためには、ラウンドトリップ時間だけ前のネットワーク状態が必要となることがわかる。そのため、フローレベルシミュレータ FSIM では、過去のネットワーク状態をメモリに格納しておくことにより、過去の状態に依存する流体近似モデルの計算を可能にする。ただし、フローレベルシミュレータ FSIM では、シミュレーション速度向上のために、微分方程式の数値計算アルゴリズムにおけるステップ時間を可変としている。そのため、必ずしも必要とする時刻のネットワーク状態がすでに計算されているとは限らない。そこで、必要とする時刻のネットワーク状態を、近傍のネットワーク状態から内挿することによって近似する。

3.3 既存のネットワーク性能評価ツールとの互換性

フローレベルシミュレータ FSIM は、既存のネットワーク性能評価ツールとの高い親和性を実現する。具体的には、現在、ネットワーク研究者らの間で広く利用されているパケットレベルシミュレータである ns2 [7] 形式のファイルの入力が可能である。フローレベルシミュレータ FSIM では、ns2 形式のシミュレーションシナリオを読み込むことが可能である。これにより、ns2 を利用したことのあるユーザであれば、簡単にシミュレーションシナリオの作成が可能である。

4 実　　験

本章では、実装したフローレベルシミュレータ FSIM を用いた実験を行い、FSIM のシミュレーション速度を評価する。実装したフローレベルシミュレータ FSIM、従来のフローレベルシミュレータ FFM [16]、パケットレベルシミュレータ ns-2 [7] のシミュレーション速度を比較する。それぞれのシミュレータ上では、同じトポロジおよびパラメータ設定のシミュレーションを実行した。実験に使用したシミュレーションのネットワークトポロジを図 5 に示す。

パケットレベルシミュレータもしくはフローレベルシミュレータを実行する計算機として、Debian GNU/Linux (sarge) (Linux カーネル 2.4.32) が稼働する Xeon 3.06 [GHz] プロセッサ、3 [Gbyte] メモリを搭載する計算機を用いた。

まず、フローレベルシミュレータが、パケットレベルシミュレータよりもどの程度高速かを調査する。そのため、二種類のフローレベルシミュレータ (FSIM および FFM) とパケットレベルシミュレータ ns-2 をそれぞれ用いて、TCP フロー数 100 ~ 1000 と変化させながら 50 [s] 間のシミュレーションを実行し、シミュレーション実行時間を計測した (図 6)。この結果より、フローレベルシミュレータは、パケットレベルシミュレータよりも圧倒的に高速であることが分かる。また、パケットレベルシミュレータは TCP フロー数が増加するにつれ、シミュレーション実行時間がほぼ線形に増加している。その一方、フローレベルシミュレータは、TCP フロー数が増加してもシミュレーション実行時間がほとんど増加していない。このことからも、フローレベルシミュレータが大規模ネットワークのシミュレーションに有効であることが分かる。

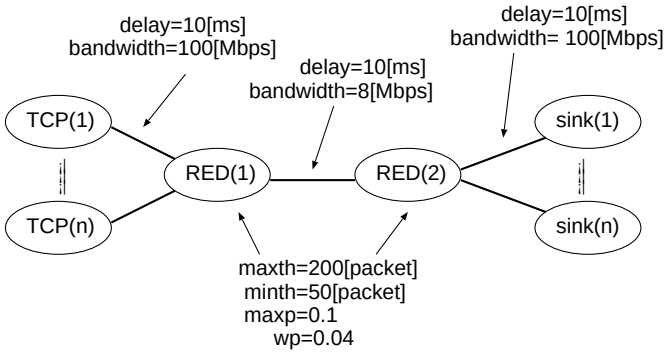

図 5: シミュレーションのネットワークトポロジ
Fig. 5　Network topology used in experiments

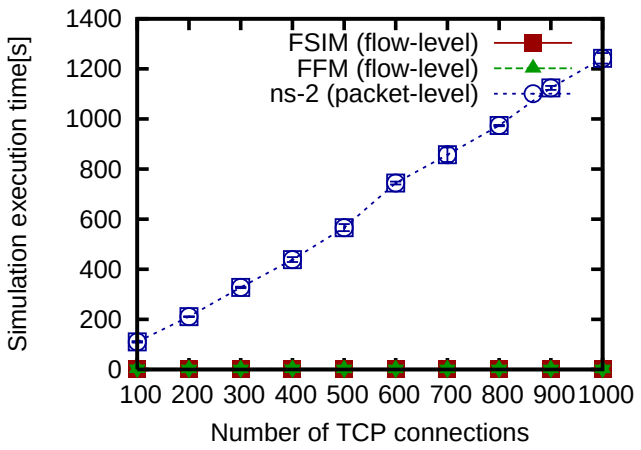

図 6: フローレベルシミュレータおよびパケットレベルシミュレータのシミュレーション実行時間の比較
Fig. 6　Comparison of simulation execution time with flow-level and packet-level simulators

次に、我々の実装したフローレベルシミュレータ FSIM が、従来のフローレベルシミュレータ FFM と比較してどの程度高速にシミュレーションが可能かを評価する。TCP フロー数を 1,000 と固定し、シミュレーション時間を 100 ~ 1000 [s] と変化させながらシミュレーションを実行し、シミュレーション実行時間を計測した (図 7)。この結果より、フローレベルシミュレータ FSIM は、従来のフローレベルシミュレータ FFM と比較して、シミュレーション実行時間が約 65% に短縮されていることが分かる。

5 まとめと今後の課題

本稿では、大規模ネットワークの性能評価のためのフローレベルシミュレータ FSIM (Fluid-based SIMulator) を提案するとともに、実装した FSIM を用いてその有効性を検証した。その結果、フローレベルシミュレータ FSIM は、従来のフローレベル

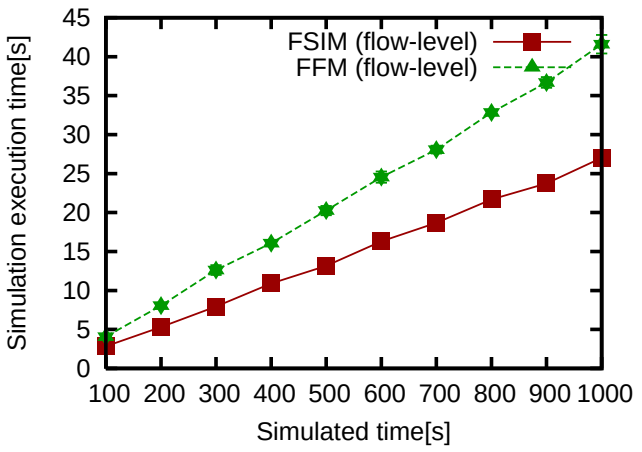

図 7: フローレベルシミュレータ FSIM および FFM のシミュレーション実行時間の比較
Fig. 7 Comparison of simulation execution time with FSIM and FFM

シミュレータと比較して、シミュレーション実行時間を約 65% に短縮できることを示した。

今後の課題として、フローレベルシミュレータ FSIM のシミュレーション精度および使用メモリ量の評価、微分方程式の数値計算アルゴリズムのさらなる改良などが挙げられる。

謝　　辞

本研究を実施するにあたり、有意義な議論をしていただいた、大阪大学大学院情報科学研究科の村田正幸氏に感謝いたします。

文　　献

[1] "Hobbes' Internet timeline v6.1." available at `http://www.zakon.org/robert/internet/timeline/`.

[2] S. Floyd and V. Paxson, "Why we don't know how to simulate the Internet," Oct. 1999. available at `http://www.aciri.org/floyd/papers/wsc.ps`.

[3] R. Jain, *The Art of computer systems performance analysis*. Wiley - Interscience, Apr. 1991.

[4] R. Pan and B. Prabhakar, "SHRiNK: A method for enabling scaleable performance prediction adn efficient network simulation," in *Proceedings of IEEE GLOBECOM 2005*, pp. 1108–1113, June 2005.

[5] V. Misra, W.-B. Gong, and D. Towsley, "Fluid-based analysis of a network of AQM routers supporting TCP flows with application to RED," in *Proceedings of ACM SIMCOMM 2000*, vol. 30, pp. 151–160, Aug. 2000.

[6] B. Liu, D. R. Figueired, Y. Guo, J. Kurose, and D. Towsley, "A study of networks simulation efficiency: Fluid simulation vs. packet-level simulation," in *Proceedings of IEEE INFOCOM 2001*, vol. 3, pp. 22–26, Jan. 2001.

[7] "The network simulator – ns-2." available at `http://www.isi.edu/nsnam/ns/`.

[8] Y. GU, Y. Lie, and D. Towsley, "On integragin fluid model with packet simulation," in *Proceedings of IEEE INFOCOM 2004*, vol. 4, pp. 2856–2866, Mar. 2004.

[9] C. Kiddle, R. Simmonds, and B. Unger, "Improving scalability of network emulation through parallelism and abstraction," in *Proceedings of IEEE ANSS 2005*, pp. 119–129, Apr. 2005.

[10] J. Zhou, Z. Ji, M. Takai, and R. Bagrodia, "Maya: a multi-paradigm network modeling framework for emulating disributed applications," in *Proceedings of IEEE PADS 2003*, pp. 162–170, June 2003.

[11] Y. Liu, F. L. Presti, V. Misra, D. Towsley, and Y. Gu, "Fluid models and solutions for large-scale IP networks," in *Proceedings of ACM/SIGMETRICS 2003*, pp. 91–101, June 2003.

[12] B. Melamed, S. Pan, and Y. Wardi, "Hybrid discrete-continuous fluid-flow simulation," in *Proceedings of IEEE SPIE 2001*, vol. 4526, pp. 263–270, July 2001.

[13] H. Ohsaki, J. Ujiie, and M. Imase, "On scalable modeling of TCP congestion control mechanism for large-scale IP networks," in *Proceedings of IEEE SAINT 2005*, pp. 361–369, Feb. 2005.

[14] A. Kavimandan, W. Lee, M. T. A. Gokhale, and R. Viswanathan, "Netowrk simulation via hybrid system modeling: A time-stepped approach," in *Proceedings of ICCCN 2005*, pp. 531–536, Oct. 2005.

[15] W. H.Press, W. H.Vetterling, S. A.Teukolsky, and B. P.Flannery, *Numerical Recipes in C*. Cambridge University Press Cambridge, May 1993.

[16] "Simulating large networks using fluid flow models (FFM)." available at `http://www-net.cs.umass.edu/fluid/fluid.html`.