

# 未知のトラヒック特性を持つネットワークの 高精度なパケット廃棄率推定手法

寺内 将大 渡部 康平 中川 健治

長岡技術科学大学 大学院工学研究科 〒940-2188 新潟県長岡市上富岡町 1603-1

E-mail: m\_terauchi@kashiwa.nagaokaut.ac.jp, {k\_watabe,nakagawa}@vos.nagaokaut.ac.jp

あらまし ネットワークの Quality of Service (QoS) を評価するためにシミュレーションは重要である。シミュレーションにより正確に評価するにはトラヒック特性の高精度なモデル化が重要だが、一般には困難である。仮に精度良くモデル化が達成できたとしても、稀にしか発生しない事象 (稀事象) を評価する場合、評価が困難である。評価したいネットワークモデルが既知の場合、Importance Sampling (IS) 法があるが、簡単なモデルでしか適用できない。モデルを利用しない IS として Trace driven IS があるが、単一フローやフロー間に相関がある場合には適用できない。本稿では、評価したいネットワークのトラヒックのモデル化を行うことなく、シミュレーションによりパケット廃棄率を高精度に推定する方法を提案する。そして、提案法の有効性を検証するため、簡単な待ち行列モデルや実トラヒックデータを用いた待ち行列の入力トラヒックを未知として扱い、検証を行う。

キーワード ネットワーク、シミュレーション、インポートランスサンプリング、パケット廃棄率、QoS 推定

## On Accurate Packet Loss Estimation Method for Networks without Traffic Models

Masahiro TERAUCHI, Kohei WATABE, and Kenji NAKAGAWA

Graduate School of Engineering, Nagaoka University of Technology, Kamitomiokamachi 1603-1, Nagaoka,  
Niigata, 940-2188 Japan

E-mail: m\_terauchi@kashiwa.nagaokaut.ac.jp, {k\_watabe,nakagawa}@vos.nagaokaut.ac.jp

**Abstract** It is important to accurately model network traffic when we evaluate Quality of Service (QoS) of networks through simulations. However, for traffic in real networks, it is a tough task to select an appropriate traffic model and tune its parameters. Even if the accurate traffic modeling is achieved, it is also difficult to accurately estimate low packet loss rate in the modern Internet. In this paper, we propose a method to accurately estimate a packet loss rate through a simulation with unknown traffic. We also show the effectiveness of the approach in a simple queueing system as a first step in our development.

**Key words** Network, Simulation, Importance sampling, Packet loss rate, QoS estimation

### 1 はじめに

ネットワークの Quality of Service (QoS) を評価するためにシミュレーションは重要である。ネットワークサービスプロバイダとユーザの間で決められる Service Level Agreements を保証するため、QoS の正確な評価は必要である。QoS は実ネットワークの観測により評価できるが、大規模なネットワークでは構築コストが高いため、ネットワークの挙動を模擬したシミュレーションで QoS を評価する場合がある。

シミュレーションで QoS を正確に評価するためには、トラヒック特性の高精度なモデル化が重要であるが、数あるトラ

ヒックモデルの中から目的に適切なものを選定し、モデルのパラメータを調整する作業は、困難な作業である。最も基礎的なトラヒックのモデルとして、一般にはポアソントラヒックモデルなどが挙げられるが、実際のネットワークのトラヒックには、自己相似性 [1] や長期依存性 [2] などポアソントラヒックモデルにはない特性を持つことが知られている。多数提案されているトラヒックモデルの特性を理解して、目的に沿ったモデルの選定やチューニングは、専門家であっても難しい。

仮に正確なモデル化が達成できたとしても、稀にしか発生しない事象 (稀事象) の高精度な評価は難しい課題である。ネットワークの QoS 評価に関しては、ITU-T の勧告 [3] により、パ

ケット廃棄率が  $10^{-5}$  以下に抑えることが求められており、パケット廃棄事象をシミュレーションにより高精度に評価することは難しい。

評価したいトラヒックモデルが既知の場合、稀事象の高精度シミュレーションを実現する Model-based Importance Sampling (IS) 法と呼ばれる方法が提案されているが、極めて原始的なトラヒックモデルにしか適用できない。Model-based IS 法では推定したい事象が発生しやすい条件に変更してシミュレーションを実行し、結果を統計的確率変換によって本来の条件での発生確率に補正する [4]。統計的確率変換は、解析的に導出されることから、ポアソントラヒックモデルなど簡単なトラヒックモデルを除いて、適用は困難である。

一方で、トラヒックモデルを利用しない IS として Trace driven IS [5] [6] が提案されているが、対象とするトラヒックが単一フローの場合や、フロー間に相関がある場合は適用できない。Trace driven IS では、計測したネットワークの複数トラヒックの位相を調節し、稀事象が発生しやすい条件のシミュレーションを実施する。そして、シミュレーションの結果を統計的確率変換により本来の条件での発生確率に補正する。Trace driven IS はその構造上、フロー別に計測したトラヒックデータが必要である他、それらが時間的に無相関でなければ、統計的確率変換の解析的な導出ができない。

本稿では、トラヒックのモデル化を行うことなく、シミュレーションによりパケット廃棄率を高精度に推定する Model-less Approach を提案する。本稿で提案する手法では、図 1 の太い実線矢印で示すように、実ネットワークから計測したトラヒックデータをモデル化を介することなく、統計的確率変換を実現する。実際に行うシミュレーションでは、トラヒックデータとは全く異なるトラヒックを使ってシミュレーションを行っているにも関わらず、適切な推定を実現できる。これにより、図 1 内の二重線矢印で示すモデル選択の困難性や、点線矢印で示す統計的確率変換導出の困難性を回避して、任意のトラヒックの高精度な廃棄率シミュレーションが実現できる。単一フローはもちろん、相関のある複数フローに関しても拡張の可能性がある。Trace driven IS が内包する問題も回避することができる。本稿の貢献は、主に以下の 3 つである。

- (1) ネットワークのトラヒックデータを基に、モデル化を介することなく統計的確率変換を導出するための理論的枠組を構築した。
- (2) 提案法の有効性を検証するために、簡単な待ち行列モデルによるトラヒックを未知として扱い、高い精度でパケット廃棄率の推定を実現できることを確認した。
- (3) 実ネットワークから収集されたトラヒックデータを利用して、提案手法の有効性を検証した。

本稿の構成は以下の通りである。まず、第 2 章で、提案法の基礎となる IS 法を概説した後、第 3 章で、本研究で扱う問題を定式化する。次に、第 4 章において、提案法のアルゴリズムを説明する。第 5 章では、提案法の有効性を待ち行列システムにおいて、ポアソントラヒックや実トラヒックデータを用いて

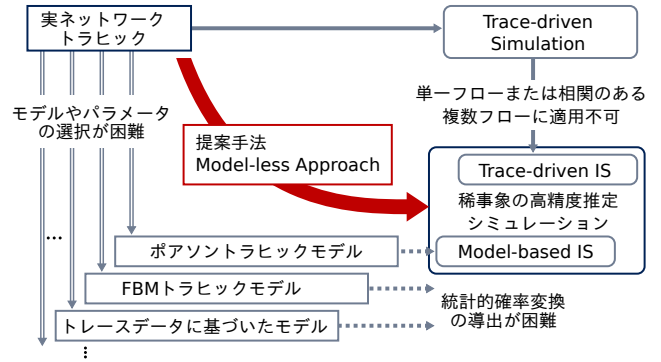


図 1 高精度シミュレーションのための困難性と Model-less approach

検証する。最後に、第 6 章で、本稿のまとめと今後の課題を述べる。

## 2 Model-based IS

Model-based IS 法とは推定したい事象が発生しやすい条件に変更してシミュレーションを実行し、結果を統計的確率変換によって本来の条件での発生確率に補正する手法である [4]。Model-based IS 法をパケット廃棄率推定に適用する場合、ネットワークを待ち行列モデルに置き換え、本来の条件より高い利用率のシミュレーションを行う。図 2 に、Model-based IS 法によるパケット廃棄率推定の概要を示す。シミュレーションネットワークモデルでは推定対象ネットワークモデルより到着率を増やしたり、サービス率を減らしたりして、パケット廃棄が発生しやすい条件に変更する。シミュレーション後、キューの遷移確率を用いて統計的確率変換し、推定対象ネットワークの廃棄率に補正する。

Model-based IS 法により、ネットワークの廃棄率を推定する場合、廃棄率の推定式は、

$$\hat{I}_{IS} = \frac{1}{\tilde{c}} \sum_{j=1}^{\tilde{c}} \left\{ \mathbf{1}_{\{j \in \phi\}} \frac{p(\tilde{\omega}_j)}{\tilde{p}(\tilde{\omega}_j)} \right\}$$

となる。ここで、 $\tilde{c}$  はシミュレーション内の送信パケット数、 $\phi$  は廃棄パケット番号の集合、 $\tilde{\omega}_j$  はシミュレーションネットワークの  $j$  番目のパケット到着までのキュー長過程、 $p(\tilde{\omega}_j)$  は推定対象ネットワーク上で  $\tilde{\omega}_j$  が発生する確率密度、 $\tilde{p}(\tilde{\omega}_j)$  はシミュレーションネットワーク上で  $\tilde{\omega}_j$  が発生する確率密度を示す。 $p(\tilde{\omega}_j)/\tilde{p}(\tilde{\omega}_j)$  が統計的確率変換を実現する部分であり、推定対象のネットワークとシミュレーションネットワークでのキュー長過程  $\tilde{\omega}_j$  が発生する確率密度の比により、結果を変換している。統計的確率変換部分がなければ、推定対象から条件を変更せずにシミュレーションを実施する通常のモンテカルロシミュレーションの推定式と一致する。

Model-based IS では、統計的確率変換にキュー長遷移確率  $p(\tilde{\omega}_j)$  及び  $\tilde{p}(\tilde{\omega}_j)$  を用いるため、キュー長遷移が発生する確率をトラヒックモデルを基に解析的に導出する必要がある。そのため、Model-based IS 法は推定対象のシステムのモデルが既知であり、解析的に取り扱い可能である必要がある。しかし、前述の通り、実ネットワークトラヒックデータからモデルの特

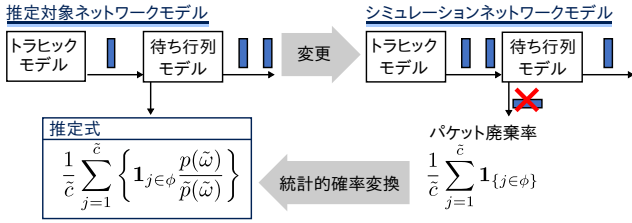


図2 Model-based IS を適用したパケット廃棄率推定

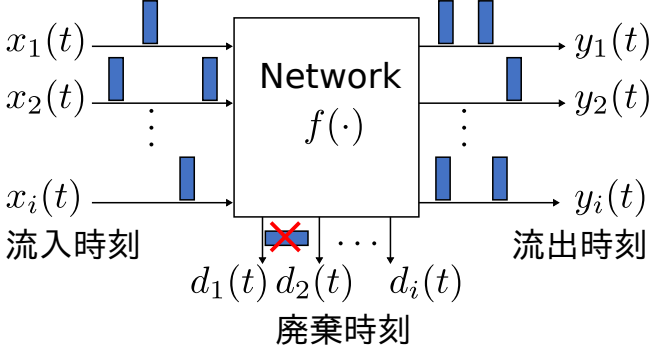


図3 ネットワークの定式化

定することは困難である上に、解析的な取り扱いが可能なシンプルなトラヒックモデルでは、実ネットワークのトラヒックが持つ特性を十分に表現できるとは限らない。

### 3 定式化

本稿では、ネットワークを入出力トラヒックの確率関数と捉え、問題を定式化する。入出力トラヒックにほとんど仮定を設けておらず、本モデルは極めて高い一般性を持つ。本章では、本稿におけるシステムモデル(図3参照)と問題の定式化について述べる。

本稿では、フロー  $i$  がネットワークに流入・流出するトラヒックを、パケット通過時刻に関する点過程  $x_i(t)$ ,  $y_i(t)$  として、それぞれ表現する。ここで、 $i \in \mathbb{N}$  とする。流入量に関する点過程は  $x_i(t) = \sum_j \delta(t - T_{\text{In}\{i,j\}})$  と表すことができる。ここで、 $\delta(t)$  はディラックのデルタ関数、 $T_{\text{In}\{i,j\}}$  はフロー  $i$  の  $j$  番目のパケット流入時刻を示す。一方、パケット流出時刻に関する点過程は、 $y_i(t)$  は  $y_i(t) = \sum_j \delta(t - T_{\text{Out}\{i,j\}})$  と表す。ここで、 $T_{\text{Out}\{i,j\}}$  はフロー  $i$  の  $j$  番目のパケット流出時刻を示す。加えて、パケット流入・流出時刻の点過程の時刻  $t$  までの実現値として、 $x_i(t)$ ,  $y_i(t)$  それぞれの時刻  $t$  までのパスを  $y_{i,t}$ ,  $y_{i,t}$  と定義する。

入出力トラヒックと同様に、パケット廃棄についても、廃棄時刻に関する点過程  $d_i(t)$  により表現する。廃棄の点過程  $d_i(t)$  は、 $\sum_j \delta(t - T_{\text{Drop}\{i,j\}})$  により定義する。ここで、 $T_{\text{Drop}\{i,j\}}$  はフロー  $i$  の  $j$  番目の廃棄パケットの廃棄時刻である。

パケット廃棄は、ネットワークに流入するトラヒックに依存して確率的に発生するとし、ネットワークのパケット廃棄率  $l = (l_1, l_2, \dots)$  は、ネットワークの挙動を表す確率関数  $f(\cdot)$  により  $l = f(x_1(t), x_2(t), \dots, y_1(t), y_2(t), \dots)$  と表現される。

$x_i(t)$  は比較的容易に計測可能である一方、パケット廃棄が

稀現象であるために、 $d_i(t)$  の計測により  $l$  を正確に求めることは困難であるとする。ITU-T の勧告 [3] に示されるパケット廃棄率の基準が  $10^{-5}$  であることから分かる通り、現代のインターネットを始めとするネットワークにおけるパケット廃棄率は非常に小さく、限られた期間の計測から得られる  $d_i(t)$  から確率関数  $f(\cdot)$  の特性を把握することは難しい。一方、トラヒック量の計測は、ネットワーク管理の観点から日常的に実施されており、これらのデータを活用することができる。

加えて、ネットワークの挙動  $f(\cdot)$  はシミュレーションにより再現可能であるとし、本稿では、計測により得られた  $x_i(t)$  の情報と  $f(\cdot)$  のシミュレーションによりパケット廃棄率の平均  $E[l]$  の高精度計測を実現する問題を取り扱う。ns-3 [7] など、パケットレベルの挙動を精密に再現するネットワークシミュレーションツールが開発されており、 $f(\cdot)$  の挙動を再現することは難しい。一方で、ネットワークに流入するトラヒックの挙動をモデル化して再現することは難しいことから、実トラヒックデータの情報を利用する。実トラヒックのデータをそのまま入力して、シミュレーションを行う Trace driven シミュレーションも可能であるが、本稿では、トラヒックデータの情報を利用しつつ、全く異なるトラヒックをシミュレートし、より高精度な推定が可能な手法を目指す。

### 4 提案法

我々の目標は、実ネットワークのトラヒックを模擬したトラヒックモデルを仮定することなく、実ネットワークにおける有限時間内の流入するトラヒック  $x_i(t)$  のトラヒックデータを元に、適切な統計的確率変換を行い、廃棄率のシミュレーションを高精度化することである。本章では、提案手法による推定式と実際に行うシミュレーションにおけるトラヒックの生成法について説明する。

#### 4.1 提案法の推定式

我々は提案手法を、従来の Model-based IS の拡張として機能しつつ、対象とするトラヒックの種類、ネットワークポロジなどの制約を排除し、高い汎用性を持つよう設計する。3章の表記法で、Model-based IS の推定式を表現すると、

$$\hat{l}_{\text{IS}} = \frac{1}{\tilde{c}} \int_0^\infty \tilde{d}_1(t) \frac{p(\tilde{x}_{1,t} \wedge \tilde{y}_{1,t})}{\tilde{p}(\tilde{x}_{1,t} \wedge \tilde{y}_{1,t})} dt$$

と書き直すことが可能である。ここで、 $\tilde{x}_{i,t}$  及び  $\tilde{y}_{i,t}$  はそれぞれ、時刻  $t$  までにシミュレーション上でフロー  $i$  パケットの流入・流出時刻を表す点過程のうち、時刻  $t$  までのパスである。また、 $\wedge$  は論理積を表し、 $p(\tilde{x}_{i,t} \wedge \tilde{y}_{i,t})$  及び、 $\tilde{p}(\tilde{x}_{i,t} \wedge \tilde{y}_{i,t})$  はそれぞれ、推定対象とシミュレーション上のネットワークにおいて、 $\tilde{x}_{i,t}$  かつ  $\tilde{y}_{i,t}$  となる事象が発生する確率である。 $\tilde{x}_{1,t}$  と  $\tilde{y}_{1,t}$  は  $\mathbb{R}^c$  空間上に定義されるため、統計的確率変換を厳密に特定するには、Model-based IS のようにモデルを仮定して解析的に算出する以外に方法はない。

提案法では、トラヒック  $x_i(t)$  のトラヒックデータに基づいた統計的確率変換を可能にするため、パラメータ  $\Delta$  を用いて流入トラヒック  $x_i(t)$  と流出トラヒック  $y_i(t)$  を離散化し、 $X_i(n)$

と  $Y_i(n)$  ( $n \in \mathbb{N}$ ) に置き換える (図 4 参照).

$$X_i(n) = \int_{(n-1)\Delta}^{n\Delta} x_i(t)dt, \quad Y_i(n) = \int_{(n-1)\Delta}^{n\Delta} y_i(t)dt.$$

そして,  $\mathbf{X}_{i,N} = \{X_i(n)\}_{0 < n \leq N}$ ,  $\mathbf{Y}_{i,N} = \{Y_i(n)\}_{0 < n \leq N}$  とし, シミュレーション上のトラヒックも同様に離散化を行うことで,  $p(\tilde{x}_{i,t} \wedge \tilde{y}_{i,t})$  と  $\tilde{p}(\tilde{x}_{i,t} \wedge \tilde{y}_{i,t})$  はそれぞれ,  $P(\tilde{\mathbf{X}}_{i,N} \wedge \tilde{\mathbf{Y}}_{i,N})$ ,  $\tilde{P}(\tilde{\mathbf{X}}_{i,N} \wedge \tilde{\mathbf{Y}}_{i,N})$  と置き換えられる. ここで,  $P(A)$ ,  $\tilde{P}(A)$  は, それぞれ推定対象のネットワークとシミュレーションネットワークで事象  $A$  が発生する確率である.  $n = t/\Delta$  として,  $\Delta \rightarrow 0$  とすると,  $P(\tilde{\mathbf{X}}_{i,N} \wedge \tilde{\mathbf{Y}}_{i,N}) \rightarrow p(\tilde{x}_{i,t} \wedge \tilde{y}_{i,t})$ ,  $\tilde{P}(\tilde{\mathbf{X}}_{i,N} \wedge \tilde{\mathbf{Y}}_{i,N}) \rightarrow \tilde{p}(\tilde{x}_{i,t} \wedge \tilde{y}_{i,t})$  に収束する. 加えて,  $\tilde{d}_i(t)$  も同様に離散化すると,  $\tilde{D}_i(n) = \int_{(n-1)\Delta}^{n\Delta} \tilde{d}_i(t)dt$  となる. 結果として, シミュレーションによる廃棄率の推定値は,

$$\frac{1}{\tilde{c}_1} \sum_{n=1}^{\infty} \tilde{D}_1(n) \frac{P(\tilde{\mathbf{X}}_{1,n} \wedge \tilde{\mathbf{Y}}_{1,n})}{\tilde{P}(\tilde{\mathbf{X}}_{1,n} \wedge \tilde{\mathbf{Y}}_{1,n})} \quad (1)$$

となる. (1) の  $P(\tilde{\mathbf{X}}_{1,n} \wedge \tilde{\mathbf{Y}}_{1,n})$  は  $\mathbb{N}^{2N}$  上で定義される確率であるため, トラヒックデータが十分に存在すれば理論上は推定することが可能である. また,  $\tilde{P}(\tilde{\mathbf{X}}_{1,n} \wedge \tilde{\mathbf{Y}}_{1,n})$  は, シミュレーションネットワークのトラヒックモデルが既知で単純なものであれば, 解析的に導出可能である.

(1) の推定値は,  $\mathbb{N}^{2N}$  の空間が広大であることから, 現実的にはほとんど機能しないため, 本研究では次元を  $\mathbb{N}^{2N}$  から  $\mathbb{N}^{2k}$  ( $k < N$ ) に削減する. 廃棄パケット数  $\tilde{D}_i(n)$  は  $(n-k)\Delta$  以前の流入トラヒックの影響をほぼ受けないと仮定し,  $P(\tilde{\mathbf{X}}_{i,n} \wedge \tilde{\mathbf{Y}}_{i,n})/\tilde{P}(\tilde{\mathbf{X}}_{i,n} \wedge \tilde{\mathbf{Y}}_{i,n}) \simeq P(\tilde{\mathbf{X}}_{i,n,k} \wedge \tilde{\mathbf{Y}}_{i,n,k})/\tilde{P}(\tilde{\mathbf{X}}_{i,n,k} \wedge \tilde{\mathbf{Y}}_{i,n,k})$  となる. ここで,  $\tilde{\mathbf{X}}_{i,n,k} = \{X_i(n)\}_{N-k < n \leq N}$ ,  $\tilde{\mathbf{Y}}_{i,n,k} = \{Y_i(n)\}_{N-k < n \leq N}$  とする.

Model-based IS の議論は 1 フローが中心であるが, 我々はこれを複数フローに拡張することで, フロー  $i$  の廃棄率の推定値  $\hat{l}_i$  は

$$\frac{1}{\tilde{c}_i} \sum_{n=1}^{\infty} \tilde{D}_i(n) \frac{P(\tilde{\mathbf{X}}_{1,n,k} \wedge \tilde{\mathbf{X}}_{2,n,k} \wedge \dots \wedge \tilde{\mathbf{Y}}_{1,n,k} \wedge \tilde{\mathbf{Y}}_{2,n,k} \wedge \dots)}{\tilde{P}(\tilde{\mathbf{X}}_{1,n,k} \wedge \tilde{\mathbf{X}}_{2,n,k} \wedge \dots \wedge \tilde{\mathbf{Y}}_{1,n,k} \wedge \tilde{\mathbf{Y}}_{2,n,k} \wedge \dots)}$$

となる. 統計的確率変換の部分は, ネットワークから計測したトラヒックデータを基に, 各パターンがどの程度の頻度で出現するかをカウントすることで, 算出することができる.

## 4.2 シミュレーションにおけるトラヒック生成

提案法では, 対象となるネットワークのトラヒックデータで観測されたトラヒックパターンを抽出し, それらのトラヒックパターンの発生回数が等しくなるように, シミュレーションで流入するトラヒック  $\tilde{x}_i(t)$  を生成する. ここで言うトラヒックパターンとは, 離散化したトラヒックデータ  $X_i(n)$  の中の連続する  $k$  要素からなる  $\mathbf{X}_{i,n,k}$  ( $n \in \mathbb{N}$ ) である.  $X_i(n)$  内に出現するすべてのトラヒックパターンを重複なしに集めた集合を  $\Omega$  と定義する.

生成のアルゴリズムは, Algorithm 1 に示すとおりである. アルゴリズムは, 実トラヒックデータ  $X_i(n)$  と提案法のパラ

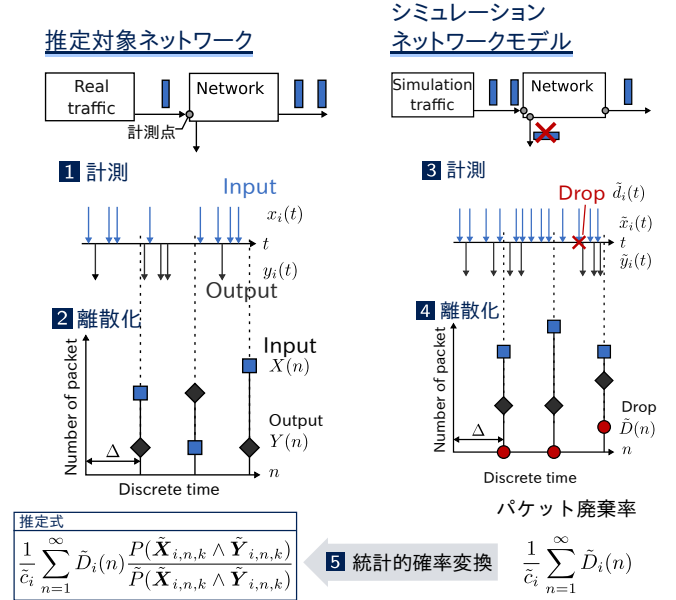


図 4 提案法の概要

メータ  $k$ ,  $\Delta$ , トラヒックを生成する時間  $T_{\text{sim}}$  から, シミュレーションに inputs する離散化されたトラヒック  $\tilde{X}_i(n)$  を生成する. まず, 各変数の初期化 (Line 1–3) を行った後に, シミュレーション時間  $T_{\text{sim}}$  から,  $\Delta$  で離散化されたトラヒックデータのスロット数  $N_{\text{sim}}$  を算出する (Line 4). 次に, Line 5–10 のループで,  $\tilde{X}_i(n)$  を生成していく. Line 6 では,  $\Omega$  に含まれるトラヒックパターンのうち, シミュレーショントラヒック  $\tilde{X}_i(n)$  内で最も発生頻度の低いものを無作為に選択する. ここで,  $h(\chi)$  は  $\tilde{X}_i(n)$  内で  $\chi$  が出現する回数,  $I_\kappa(\tilde{X}_i(n), \chi)$  は  $\tilde{X}_i(n)$  の最後の  $\kappa$  要素と  $\chi$  の最初の  $\kappa$  要素が一致するとき 1, それ以外で 0 となる指示関数である. もし条件を満たす  $\chi$  が存在しない場合,  $\kappa$  を減算して,  $\tilde{X}_i(n)$  と  $\chi$  の重複条件を緩和する (Line 7). 条件を満たす  $\chi$  があれば, 選択したトラヒックパターンを  $\tilde{X}_i(n)$  に接続し, 発生回数をカウントし,  $\kappa$  を初期化する (Line 8–11). このようにして得られた離散化トラヒック  $\tilde{X}_i(n)$  を基に,  $\Delta$  内のパケット到着時刻を一様乱数で決定することで, シミュレーショントラヒック  $\tilde{x}_i(t)$  を得ることができる.

## 5 評価

本稿では, 提案法の有効性を検証するため, 待ち行列モデルの入力トラヒックを推定対象のネットワークから計測により得られた特性が未知のトラヒックデータとして扱い, パケット廃棄率の推定を行った. トラヒックデータは, ポアソントラヒック, 2 状態 MMPP トラヒック, 実トラヒックデータを使い, これらのトラヒックを G/M/i/K 待ち行列モデルに入力した場合の廃棄率を推定した. 比較対象は, トラヒックデータと同じ入力をそのままシミュレートした場合, すなわち Trace driven シミュレーションによる推定結果とした. 各待ち行列モデルのパケット廃棄率推定を提案法と Trace driven シミュレーションで行い, 推定値の平均と分散を用いて推定精度の比較を行っ

**Algorithm 1** シミュレーショントラヒックの生成**Require:**  $X_i(n), k, \Delta, T_{\text{sim}}$ **Ensure:**  $\tilde{X}_i(n)$ 

```

1:  $\tilde{X}_i \leftarrow [0], \kappa \leftarrow k - 1$ 
2: for all  $\chi' \in \Omega$  do
3:    $h(\chi') \leftarrow 0$ 
4:  $N_{\text{sim}} \leftarrow \lceil T_{\text{sim}}/\Delta \rceil$ 
5: while length of  $\tilde{X}_i < N_{\text{sim}}$  do
6:    $\chi' \leftarrow \text{Random Sampling from } \arg \min_{I_{\kappa}(\tilde{X}_i, \chi)=1} h(\chi)$ 
7:   if  $\chi' = \emptyset$  then  $\kappa \leftarrow \kappa - 1$ 
8:   else
9:      $[\chi'(\kappa + 1), \dots, \chi'(k)]$  push to  $X_i$ 
10:     $h(\chi') \leftarrow h(\chi') + 1$ 
11:     $\kappa \leftarrow \kappa - 1$ 

```

た．提案法のトラヒックを離散化する時間間隔  $\Delta$  とトラヒックパターン長さ  $k$  を変化させながら推定を行った．

**5.1** ポアソントラヒックによる検証

まずは，最も基礎的な検証として，ポアソントラヒックを G/M/1/K 待ち行列に入力した場合，すなわち M/M/1/K 待ち行列モデルの packets 廃棄率推定を行った．packets 廃棄率の理論値が  $10^{-5}$  となるよう packets の到着率を  $\lambda = 329.1$  [packet/s]，サービス率を  $\mu = 1000$  [packet/s] とし，最大系内数  $K = 10$  [packet] とした．また，推定対象とシミュレーションの待ち行列において測定時間を 2000 [s]，シミュレーションの試行回数は 30 回とした．

提案法における推定値の平均を図 5 に示す．横軸は提案法のパラメータの積  $k\Delta$ ，縦軸は提案法による推定値の平均  $E[\hat{I}]$  を示す．また， $1.0 \times 10^{-5}$  上に引かれている黒線は，今回検証している M/M/1/K 待ち行列の packets 廃棄率の理論値を表している．図より， $k\Delta \geq 5 \times 10^{-2}$  となる領域で，バイアスなく推定できていることが確認できる．

推定値の分散と Trace driven シミュレーションの結果の比較を図 6 に示す．横軸は提案法のパラメータ  $k\Delta$ ，縦軸は提案法による推定値の分散  $\text{Var}[\hat{I}]$  を示す．推定値の分散が小さいほど，推定値のばらつきが小さく，高精度に推定できることを表している．このグラフより，提案法は従来の Trace driven シミュレーションと比較して推定値の分散が小さい領域が存在することが確認でき，図 5 と合わせると，提案法は Trace driven シミュレーションと比較して高精度に推定できる領域が存在する．特に，提案法において  $k = 4, k\Delta = 5 \times 10^{-2}$  [s] のとき，従来法の推定値の分散と比較しておよそ 1/12 程度となっていることが確認できる．

**5.2** 2 状態 MMPP トラヒックによる検証

次に，2 状態 MMPP トラヒックを入力した場合の packets 廃棄率推定を行った．2 状態 MMPP/M/1/K 待ち行列モデルの packets 廃棄率が  $10^{-5}$  となるように各状態における packets の到着率  $\lambda_1 = 100.0$  [packet/s]， $\lambda_2 = 338.6$  [packet/s]，各状態における平均滞在時間  $1/\alpha_1 = 1/\alpha_2 = 1$  [s]，サービス率  $\mu = 1000$  [packet/s]，最大系内数  $K = 10$  [packet] とした．ま

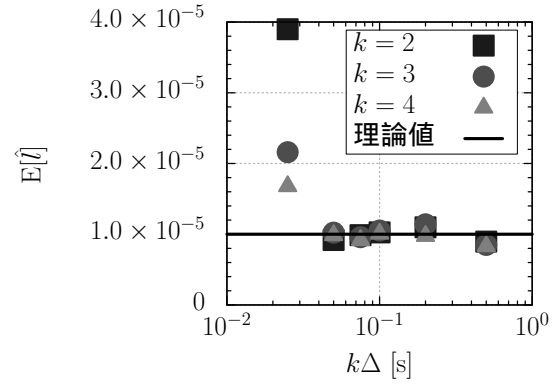


図 5 M/M/1/K 待ち行列の packets 廃棄率推定値の平均

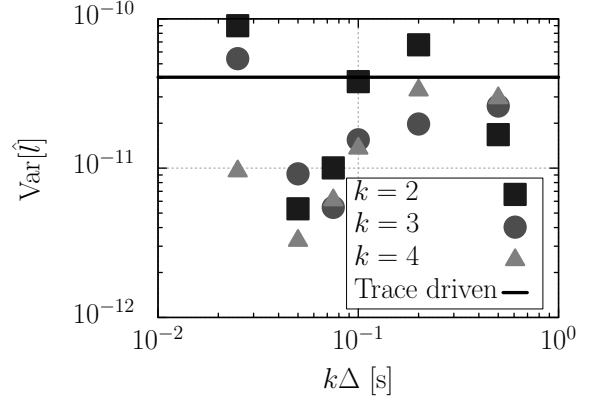


図 6 M/M/1/K 待ち行列の packets 廃棄率推定値の分散

た，推定対象とシミュレーションの待ち行列において測定時間は 5000 [s]，試行回数は 30 回とした．

提案法における推定値の平均と分散を図 7 と図 8 に示す．2 つのグラフより，提案法は従来法と比較し高精度に推定できる領域が存在する．特に，提案法において  $k = 4, k\Delta = 5 \times 10^{-2}$  [s] のとき，従来法の推定値の分散と比較しておよそ 1/12 程度となっていることが確認できる．

**5.3** 実トラヒックデータによる検証

最後に，提案手法が実データに対しても適用可能であることを示すために，Widely Integrated Distributed Environment (WIDE プロジェクト) [8] が公開している packets トレースデータの 2018 年 9 月 11 日に計測されたデータの packets 到着時刻を抽出してシミュレーションを行った．実トラヒックデータを用いているため，packets 廃棄率を解析的に求めることが困難である．そのため，packets 廃棄率がおよそ  $10^{-4}$  程度になるように，Trace driven シミュレーションを行いながら退去率を調節し， $\mu = 10^6$  [packet/s] とした．最大系内数については， $K = 10$  [packet] とした．また，推定対象とシミュレーションの待ち行列において測定時間を 100 [s]，シミュレーションの試行回数は 10 回とした．

推定値の平均と分散を図 9 と図 10 にそれぞれ示す．このグラフより， $1.5 \times 10^{-5} \leq k\Delta \leq 2.0 \times 10^{-5}$ ， $k \geq 4$  の場合，推定値の平均が Trace driven シミュレーションの推定値の平均に近く，かつ分散も小さくなっていることが確認でき，提案法が高精度に推定できる領域がある．特に， $k = 6, k\Delta = 2.0 \times 10^{-5}$  [s]



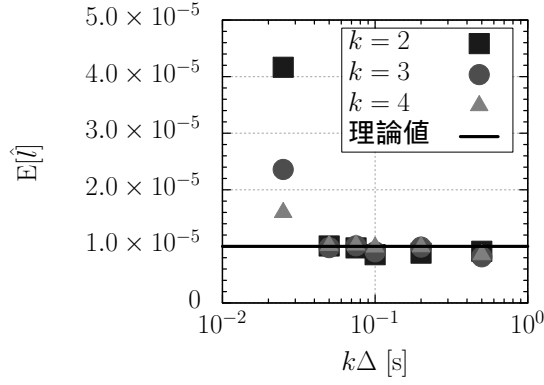


図 7 2 状態 MMPP/M/1/K 待ち行列のパケット廃棄率推定値の平均

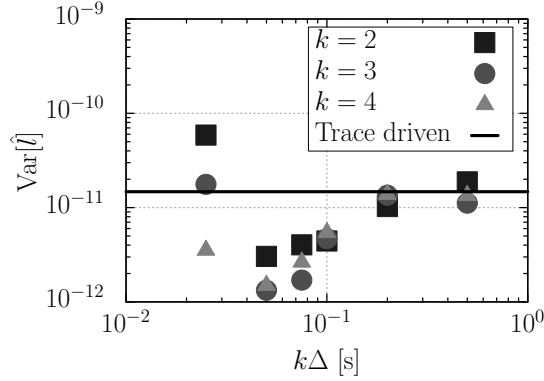


図 8 2 状態 MMPP/M/1/K 待ち行列のパケット廃棄率推定値の分散

の時, Trace driven シミュレーションと比較して推定値の分散が 1/160 程度小さくなっていることが確認できる。

## 6 おわりに

本稿では, ネットワークのパケット廃棄率推定について, 推定対象のトラヒックをモデル化することなく, 統計的確率変換を行うことでシミュレーションで発生確率を高精度に推定する方法を提案した。提案手法では, トラヒックを離散化することで, トラヒックパターンの頻度分布を算出し, 統計的確率変換を実現している。提案法の有効性を確認するため, 実トラヒックデータを含む複数のトラヒックについて, G/M/1/K 待ち行列モデルを用いて検証を行った。検証の結果, パケット廃棄率を従来法と比較して高精度に推定できる領域が存在することを確認し, 推定値の分散が従来法と比較しておよそ 1/12~1/160 程度小さくなることが確認できた。

今後の課題として, 提案法の最適なパラメータの設計手法の検討が挙げられる。今回の検証により, 提案法の推定性能はパラメータ  $k$ ,  $\Delta$  に対して大きく影響があることを確認しており, 追加の検証が必要である。また, より現実的なネットワークモデルでの有効性の検証も今後予定している。

## 謝 辞

本研究の一部は, JSPS 科研費 JP17K00008, および JP18K18035 の助成を受けたものである。

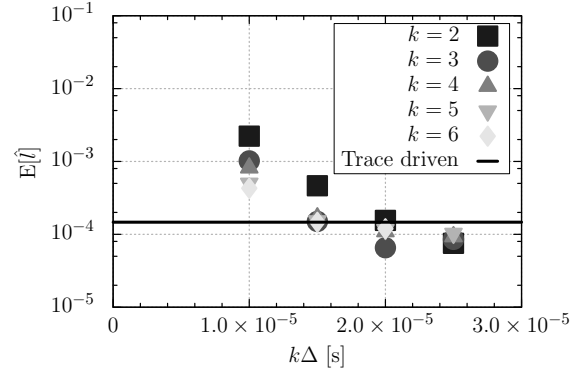


図 9 実データを用いた待ち行列のパケット廃棄率推定値の平均

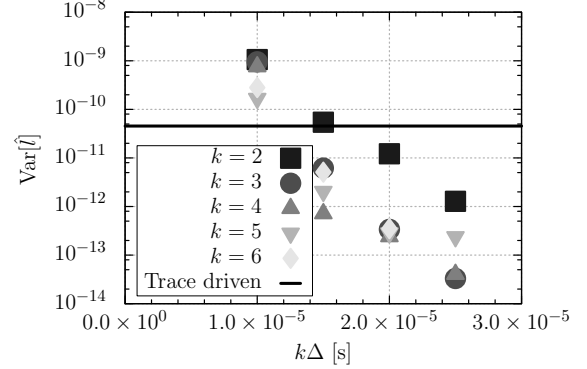


図 10 実データを用いた待ち行列のパケット廃棄率推定値の分散

## 文 献

- [1] W. E. Leland, M. S. Taqqu, W. Willinger, and D. V. Wilson. On the self-similar nature of ethernet traffic (extended version). *IEEE/ACM Transactions on Networking*, Vol. 2, No. 1, pp. 1–15, 1994.
- [2] Vern Paxson and Sally Floyd. Wide area traffic: The failure of poisson modeling. *IEEE/ACM Transactions on Networking*, Vol. 3, No. 3, pp. 226–244, 1995.
- [3] ITU-T Rec. Y.1541. Network performance objectives for ip-based services. 2011.
- [4] J. Morio, M. Balesdent, D. Jacquemart, and C. Vergè. A survey of rare event simulation methods for static input–output models. *Simulation Modelling Practice and Theory*, Vol. 49, pp. 287–304, 2014.
- [5] I. C. Paschalidis and S. Vassilaras. Importance sampling for the estimation of buffer overflow probabilities via trace-driven simulations. *IEEE/ACM Transactions on Networking*, Vol. 12, No. 5, pp. 907–919, 2004.
- [6] P.E. Heegaard, B.E. Helvik, and R.O. Andreassen. Application of rare event techniques to trace driven simulation. In *Proceedings of the 37th Winter Simulation Conference*, 2005.
- [7] T.R. Henderson, M. Lacage, G.F. Riley, C. Dowell, and J. Kopena. Network simulations with the ns-3 simulator. In *Proceedings of SIGCOMM 2008 demonstration*, p. 527, 2008.
- [8] WIDE: Widely Integrated Distributed Environment, <http://www.wide.ad.jp/>.