

ネットワーク障害におけるパスの統合・分割による障害箇所の推定

堤 陽祐[†] 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: [†]tytsutsumi@kashiwa.nagaokaut.ac.jp, ^{††}{k_watabe,nakagawa}@nagaokaut.ac.jp

あらまし 近年、ネットワークの大規模化・複雑化が進み、サイレント故障などの自律的に検知が困難な障害の検知及び障害箇所の推定が大きな課題となっている。本稿では、ネットワークに流入する全パケットをフィルタにより分類し、廃棄率を計測することにより、障害箇所を推定する手法を提案する。提案手法では、パケットが通過するパスを統合・分割することによって Measurement Unit (MU) を構成し、少ないフィルタで障害箇所推定を実現する。フィルタの設置による廃棄率計測のほか、Management Information Base (MIB) を利用することで、より多くの障害を検知可能にする。提案法の推定精度を評価するため、3つの実ネットワークの構成データを利用して、様々な障害パターンについて障害箇所を推定する際の推定精度を検証し、大部分の障害パターンをポート単位まで絞り込み可能なことを示した。

キーワード ネットワーク障害、パッシブ計測、障害箇所推定、ネットワークモニタリング、ネットワークトモグラフィ

Estimation of Failure Locations by Path Integration/partition in Network Failure

Yosuke TSUTSUMI[†], Kohei WATABE^{††}, and Kenji NAKAGAWA^{††}

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

E-mail: [†]tytsutsumi@kashiwa.nagaokaut.ac.jp, ^{††}{k_watabe,nakagawa}@nagaokaut.ac.jp

Abstract As networks become larger and more complex in several decades, fault localization for silent failures has been difficult. In this paper, we propose a method to estimate a failure location by classifying incoming packets with packet filters. In the proposed method, Measurement Units (MU) are constructed by integrating/dividing paths through which packets pass. It achieves failure localization with minimal number of filters. In addition to measurement of packet loss rate by installing filters, various failures are enabled to be detected by using Management Information Base (MIB). By using the configuration data of the three real networks, we evaluate the localization accuracy of the proposed method. Localization accuracy is verified for various failure patterns, and we can confirm that it is possible to narrow down the large part of the failure patterns to a port.

Key words Network failure, Passive measurement, Failure localization, Network monitoring, Network tomography

1. はじめに

近年、Internet of Things (IoT) やクラウドサービスなど、ネットワークを利用した様々なサービスが急速に発展しており、ネットワークの大規模・複雑化が進んでいる。複雑化したネットワークを管理する技術として、ネットワークの構成をソフトウェアによって定義するネットワーク仮想化の普及が進んでいる。ネットワーク仮想化によりネットワーク構成を柔軟に変更できるようになった反面、オペレータの設定ミスに起因す

るサイレント障害がネットワーク障害の主要な原因となっている [1]。

サイレント障害はネットワークを構成する機器の設定ミスなどにより、機器が正常に動作しているにも関わらずパケットがネットワーク内を正しく転送されない障害である。サイレント障害により、通過するべきパケットが意図せず廃棄されたり、逆に廃棄されるべきパケットが転送されてしまうなどといった問題が発生する。サイレント障害ではシステムがアラートを発しないため、障害の発生を自動で検知することが困難である。

オペレーターが個別に機器を監視することでサイレント障害の検知が可能だが、ネットワークの大規模化に伴う膨大な数の機器を監視することはオペレーターの大きな負担となり、現実的ではない。また、トラフィック監視などにより QoS の低下から障害の発生を検知できた場合でもその障害がネットワーク内のどの箇所で行っているのか推定することは難しく、サイレント障害の自動検知及び障害箇所の推定が大きな課題となっている [2]。

ネットワークの遅延やパケット廃棄などの状態を監視する手法は、アクティブ計測とパッシブ計測の 2 つに分けられる。アクティブ計測は計測対象のネットワークにプローブパケットを送出し、そのパケットの状態からネットワークの品質を推定するものである。パッシブ計測は実際に計測対象のネットワークを流れるユーザーのトラフィックを監視することでネットワークの状態を推定する。

パッシブ計測はアクティブ計測と比べて多くのメリットがある。ネットワークが混雑している時、アクティブ計測のプローブパケットは計測対象のネットワークを更に混雑させてしまうが、パッシブ計測ではプローブパケットを新たにネットワークにパケットを流す必要が無いため、ネットワークに負荷をかけることがない。また、実際にネットワーク内を流れるトラフィックを監視するため、ユーザーのトラフィックが経験するネットワークの状態を正確に推定することができる。一方でパッシブ計測を行うにはネットワーク上にパケットの計測装置を設置をしなければならないためアクティブ計測に比べて計測に必要なコストがかかるデメリットがある。

本研究では、ネットワーク障害の検知及び障害箇所の推定を、最小限のパッシブ計測のコストで実現する方法を提案する。本手法では、ネットワークに流入するパケットのパスの統合と分割により Measurement Unit (MU) を構成し、パッシブ計測を用いて MU 毎にパケット廃棄率を計測する。パケット廃棄率を計測するためには、ネットワーク内の各ポートでパケットヘッダを読み取り、通過パケット数をカウントするパケットフィルタを設置する必要があるが、本手法では、このフィルタ数をコストと定義して最適化を行う。パスの統合により計測に必要なコストを抑えつつ、サイレント障害などによる Quality of Service (QoS) の低下を検知し、ネットワークの障害箇所を精度良く推定する。ネットワークを通過するトラフィックのみならず、内部で廃棄されるトラフィックについても障害検知及び障害箇所の推定に利用する。

2. ネットワークにおける障害と原因

ネットワークでは様々な障害によって転送途中のパケットが破棄されてしまい、受信先まで転送されないネットワーク障害が生じる。パケットが破棄される原因はネットワーク機器の物理的な障害によるものとオペレーターの設定ミスによる障害に大別される。

ネットワーク機器が部品の寿命を迎えたり、トラフィックの処理に対してメモリなどのリソース不足を起こした場合に異常停止や再起動を繰り返す事がある。このようなネットワーク機

器の物理的な障害が発生した場合、障害した機器やポートを通過する全てのパケットが影響を受ける障害が発生する。

一方、オペレーターの設定ミスによる障害では、帯域設定の値や QoS 制御の適用するヘッダ範囲のミスなど制御ルールの設定ミスにより、その制御ルールの適用を受ける一部のパケットのみが影響を受けて破棄される。設定ミスによる障害の場合、ネットワーク機器自体は設定に従い正常に動作を続けるため、QoS が低下しているにも関わらずアラートを発しない。また、物理的な障害の場合のように障害箇所を通過する全てのパケットが影響を受けるのではなく、一部のパケットのみが影響を受けるため、障害発生を検知及び原因の推定が難しい。このような自動検知の困難な障害をサイレント障害と呼ぶ。

SDN の普及に従いネットワークの設定は動的に変更されるようになり、オペレーターの設定ミスによる障害がネットワーク障害の主要な原因の一つとなっている。2017 年には米 Google がネットワークの経路情報の設定ミスにより大規模なネットワーク障害を引き起こし、日本国内のインターネット接続や電子決済などの様々なサービスが一時的に利用できなくなった。このような設定ミスに起因するネットワーク障害の検知、及び障害箇所の推定が大きな課題となっている。

3. 提案手法

3.1 システム構成

本研究では、パッシブ計測による計測コストを抑えたネットワーク障害の検知及び障害箇所の推定手法を提案する。本稿では、フォワーディングやアクセス制御などのネットワークの機能毎に障害が発生すると想定し、Forwarding Information Base (FIB) ならば転送先のポート毎、Access Control List (ACL) ならば適用されるポート毎に実行される処理をアクションと定義する。アクションは、障害発生の最小単位であり、本研究では、ポート毎やスイッチ毎の障害は、ポートまたはスイッチに含まれるアクションがすべて障害状態に陥ると捉える。つまり、障害の発生パターンは、障害を起こすアクションの組み合わせで表現される。

本稿で提案する障害箇所推定技術では、APV [3] のコンセプトに基づき、パスの一部を統合・分割することで計測パスとなる MU を構成し、通過するパケットの廃棄率を計測する。提案法では図 1 に示すようなシステムの構成を用い、MU の端に位置する各ポートで MDD [4] を利用したフィルタを設置してパケット数をカウントする。フィルタが記録したパケット数を定期的に集約し、集約した情報を基に得られる MU 毎の廃棄率より障害原因となっているアクションを推定する。

計測に用いるフィルタ数が増えるほど、設置されたスイッチにおいてメモリなどのリソースが消費される。提案法では、フィルタ数を計測に必要なコストとし、フィルタ数を最小化する。

3.2 障害箇所の絞り込み

障害箇所の絞り込みを行うために、提案法では、複数のパスを統合・分割した MU を構成し、廃棄率を計測することで、品質劣化した MU を特定する。ネットワーク内で適用されるア

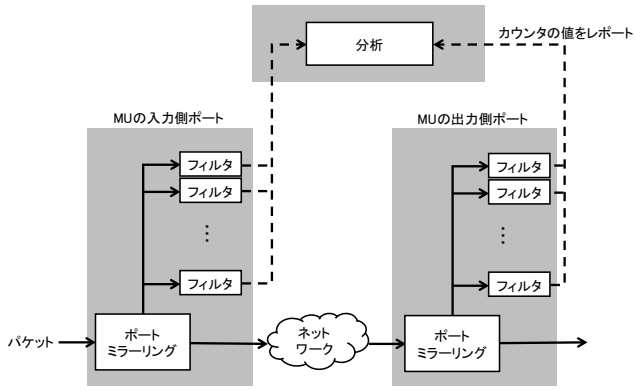


図1 提案する障害箇所推定法のシステム構成

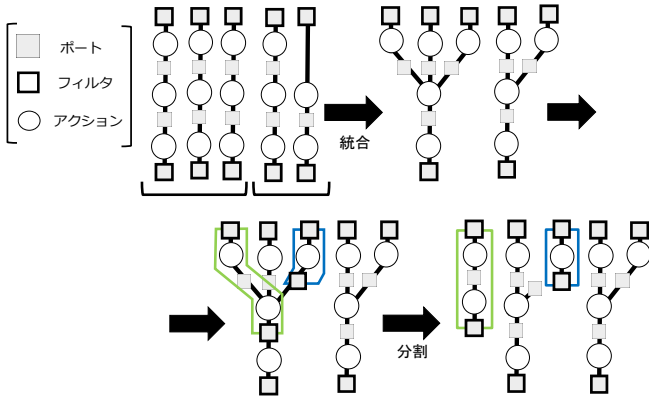


図2 パスの構成

クシヨンの全集合を $R_{\Omega} = \{a_j | j = 0, 1, \dots, n\}$ として、パケットが通過するアクションとポートの列をパスと定義する。ネットワークを通過するトラフィックであれば、パスの両端には、必ずポートが存在し、上流側のポートを入力ポート、下流側のポートを出力ポートと呼ぶ。また、図2(上)のように出力ポートが共通のパスの一部を結合したものを MU $\{t_i\}$ と定義する。MU は、図2(下)のように、任意の点で2つの MU に分割することも可能である。MU の端に位置するポートのフィルタで単位時間あたりに通過したパケットの総数をカウントし、比較することで、MU 毎の廃棄率 $\{l_{t_i}\}$ を計測することができ、品質劣化が発生している MU を検知することができる。計測したパケット廃棄率に対してしきい値 δ を設けて、 $l_{t_i} > \delta$ となる MU を品質劣化 MU と定義する。

上記のように特定した品質劣化 MU のパターンから、品質劣化の原因となっている障害箇所のアクションを絞り込むことができる。アクション a_j ($j = 1, 2, \dots$) を含む MU の集合を T_{a_j} とし、アクション a_j に関する障害により T_{a_j} のすべての MU のパケット廃棄率が δ 以上となると仮定する。品質劣化 MU の集合を $T_{\text{loss}} = \{t_i | l_{t_i} > \delta\}$ 、MU t_i に含まれるアクションの集合を R_{t_i} とすると、障害箇所と推定されるアクションは $\bigcap \{R_{t_i} | t_i \in T_{\text{loss}}\} - \bigcup \{R_{t_i} | t_i \in \bar{T}_{\text{loss}}\}$ に絞られる。この絞り込みのアプローチは、Automatic Test Packet Generation (ATPG) [2] における障害箇所の絞り込みと同様のアプローチである。

絞り込みの精度とコストの間には、トレードオフの関係があ

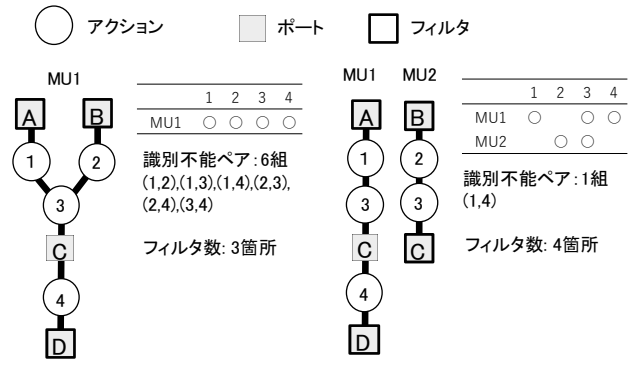


図3 パスの分割による障害アクションの識別

る。MU の数が多いほど、カウントを行う地点が増えるため、設置するフィルタ数が増加し、コストが増大する。しかし、MU を細かく分割するほど、障害箇所となっているアクションを絞り込むための廃棄率の情報が増えるため、絞り込みの精度が向上する。

3.3 MU の構成アルゴリズム

提案法では、MU を構成を最適化し、最低限のフィルタ数で、障害箇所のアクションの推定を行う。ここでは MU の構成アルゴリズムについて説明する。

まず、提案法では、図2(上)のように、出力側のアクションが等しいパス同士をすべて統合する。パスの統合を行うことで出口側でのパケットの分類が減り、計測に必要とするフィルタ数が削減される。

次に、図2(下)のように、MU の任意のポートにフィルタを追加して通過するパケットを分類することで、フィルタを追加したポートから任意の入力ポートまでの部分パスを MU から分割する。分割後のそれぞれの MU でパケット廃棄を計測することで、障害の原因がどちらの MU に含まれるアクションにあるか識別することができる。例えば、図3の例では、ポート C にフィルタを追加し、ポート B から C までの部分パスを分割することで、5 対のアクションペアの障害を識別できるようになっている。

必要最低限の数だけフィルタを追加する問題を考えると、集合被覆問題 (Minimum set covering problem) [5] として定式化することができる。ここで、ポート s_i にフィルタを追加して MU t_i を分割した場合に、障害原因が互いに識別可能となるアクションペアの集合を P_{s_i} と定義する。さらに、 $\{P_{s_i}\}$ の和集合を $P_{\Omega} = \bigcup_{i=1}^n P_{s_i}$ とすると、識別可能な障害原因となるアクションペア数を最大化しつつ、フィルタ数を最小化する問題は、 $\{P_{s_i}\}$ のうち P_{Ω} をすべてカバーする最小の部分集合を求める集合被覆問題であることがわかる。集合被覆問題は一般には NP 困難であるが、貪欲法により良い解が得られることが知られているため、提案法では貪欲法 [5] を利用したアルゴリズムを用いて分割パスの選択を行う。

4. MIB を利用した MU の追加

MU の構成では、MU の出力ポートにフィルタを設置してい

表 1 検証に利用したネットワークの統計情報

	Internet2	Stanford	Purdue
スイッチ数	9	16	1,646
エッジスイッチ数	9	12	92
利用ポート数	157	114	3,541
ネットワークエッジポート数	116	39	482

たため、ネットワークを通過するパケットに関連するアクションしか推定できなかったが、廃棄されるパケット数を記録する Management Information Base (MIB) を利用することで、内部で廃棄されるパケットに関連するアクションの障害も推定可能になる。ネットワーク内部で破棄されるトラフィックでは、パケットはフィルタを通過することなく破棄されるため、パケットカウントを行うことができないが、スイッチの管理情報を格納している MIB を利用することで、各ポートやスイッチで破棄されたパケット数を得ることができる。MIB を利用することでネットワーク内部で破棄されるトラフィックについても MU の構成が可能になる。ネットワーク内部で破棄されるトラフィックの場合、MU の終端部分にフィルタがないため、部分パスの分割はできない。

5. 提案手法の評価

提案法による障害箇所の推定精度を調査するために実ネットワークの構成データを用いて検証を行った。構成データには Stanford [6], Internet2 [7], Purdue [8] を用い、それぞれ FIB のみを対象にした場合と FIB と ACL の両方を対象にした場合の条件でそれぞれ推定精度の評価を行った。検証に使用したネットワークの規模は表 1 に示す通りである。

5.1 アクション毎の障害箇所推定

まず、アクション毎に障害が発生すると仮定した場合の障害箇所推定精度を検証した。最初の検証では、MIB を利用した MU の構成は行わず、ネットワークの外部から外部へ通過するトランジットトラフィックが適用を受けるアクションを推定対象とし、対象のアクションのうち一つが障害状態に陥るとして障害パターンを設定した。障害パターンが一律の確率で発生する場合に、提案法により、スイッチ、ポート、アクションのそれぞれの単位で障害箇所を一意に絞り込みできる確率と推定に必要なフィルタ数の評価を行った。また、[9] を利用して、パケットをパス毎に分類した場合の障害箇所推定精度についても同様の条件で検証を行い、これを従来法として提案法との推定精度とフィルタ数の比較を行った。

検証の結果を表 2 に示す。検証を行った結果、FIB のみを対象にした障害箇所の推定では、3 つ全てのネットワークにおいて識別することのできないアクションペアが存在せず、ネットワーク内のどのアクションが障害原因となった場合でも必ず障害箇所をアクション単位で一意に推定できることを確認した。一方、FIB と ACL の両方を対象にした場合、識別不能なアクションペアが存在し、アクション単位で障害箇所を一意に推定できる確率は 8~40% であり、一意特定は困難であるとの結果が得られた。しかし、この場合でも障害箇所をポート単位では

90%以上の確率で一意に推定することを確認した。

障害箇所推定に必要とするフィルタ数を従来法のフィルタ数と比較した結果、MU の構成において多くのパスが統合され、従来法の 51 %~58 % のフィルタ数で同等の障害箇所推定が可能であることがわかった。

ここまでの検証では、障害が発生するアクションは高々 1 つであると仮定して、検証を行ったが、複数のアクションで同時に障害が発生すると仮定した場合についても検証を行った。単体のアクションの障害に加え、任意の 2 つ、または 3 つのアクションが同時に障害状態に陥る可能性があり、それらの障害パターンが一律の確率で発生すると仮定した場合の Stanford(FIB) の検証結果を表 3 に示す。結果から、障害パターンは飛躍的に増加しているに関わらず、フィルタ数はそれほど増加していないことが確認できる。同時に発生する障害が離れた箇所であれば、ほとんどの場合、それらを識別するために必要な MU は異なるため、単体のアクションの障害を識別する MU 構成と同じ構成で個別に識別可能である。

加えて、MIB を利用した MU を加えた場合に、対象となるアクション数がどの程度増加するかも検証した。各構成データ毎の結果を表 4 に示す。結果から、対象となるアクション数が増加し、より多くの障害を推定可能になっていることが確認できる。

5.2 アクション、ポート、及びスイッチ毎の障害の障害箇所推定

アクション毎の障害に加えて、ポート及びスイッチ毎の障害を想定して、障害箇所の推定精度を評価した。すべてのアクション、ポート、スイッチで障害が発生する可能性があり、同時に障害が発生する点は高々 1 つであると仮定して、障害パターンを設定した。障害パターンが一律の確率で発生する場合に、スイッチ、ポート、アクションのそれぞれの単位で障害箇所を一意に推定できる確率と推定に必要なフィルタ数の評価を行った。また、前節と同様に、[9] の従来法も提案法と比較した。

比較した結果を表 5 に示す。検証の結果、アクション毎の障害のみを想定した障害箇所推定と同様に障害箇所をポート単位では 90%以上の確率で一意に推定し、スイッチ単位では必ず一意に推定できることを確認した。また、計測に必要なフィルタ数は Purdue(FIB) を除き、ポート毎、スイッチ毎の障害を考慮した場合でも増加することはなかった。これは、アクション毎の障害を推定する時点で多くのフィルタが追加されていることと、Purdue は Internet2, Stanford に比べネットワークの規模が大きく、ポート毎やスイッチ毎の障害を考慮することで障害パターンが Internet2, Stanford に比べ多く増加したためだと考えられる。

6. 関連研究

ネットワークを運用していく上で、ネットワークの障害箇所を推定する技術は重要であり、現在までに様々な技術が提案されている。ネットワークトモグラフィは、エンドツーエンドのフローの品質とトポロジーの情報を利用して、ネットワーク内部の状態を推定する障害箇所推定技術である。アクティブ計測

表 2 アクション毎に障害が発生する場合の障害箇所推定精度

データセット	Stanford				Internet2				Purdue			
	FIB		FIB/ACL		FIB		FIB/ACL		FIB		FIB/ACL	
障害箇所推定技術	従来法	提案法	従来法	提案法	従来法	提案法	従来法	提案法	従来法	提案法	従来法	提案法
障害パターン数	58	58	427	427	140	140	304	304	526	526	1174	1174
MU 数	319	50	379	67	11159	118	11183	142	3015	378	4716	529
フィルタ数	638	368	758	437	22318	11277	22366	11301	6030	3393	9432	5226
識別不能なアクションペア数	0	0	49	49	0	0	140	140	0	0	409	409
アクションまで絞込できる確率	100%	100%	26%	26%	100%	100%	8%	8%	100%	100%	40%	40%
ポートまで絞込できる確率	100%	100%	94%	94%	100%	100%	100%	100%	100%	100%	95%	95%
スイッチまで絞込できる確率	100%	100%	100%	100%	100%	100%	100%	100%	100%	100%	100%	100%

表 3 Stanford (FIB) 複数同時障害を考慮した障害箇所推定精度

ネットワークデータ	Stanford (FIB)	
最大同時障害数	2	3
障害パターン数	1711	30914
MU 数	79	80
フィルタ数	399	403
識別不能なアクションペア数	0	0
障害アクションを一意に推定できる確率	100%	100%
障害ポートを一意に推定できる確率	100%	100%
障害スイッチを一意に推定できる確率	100%	100%

を使ったアプローチが主に用いられてきたが[10]~[14], TCP SYN/ACK パケットのパッシブ計測を利用したアプローチも提案されている[15]~[17]. また, ユーザートラフィックを対象にしたパッシブ計測では[9]があり, ネットワークのエッジポートと一部の内部ポートでフロー毎にパケットを計測することでカウントを行うポート数を最小化しつつ, ネットワークのリンク障害の検知及び障害箇所の推定を行っている. その他, QoS 制御などの高度なネットワーク機能に関する障害箇所推定技術として, ATPG[2]なども提案されている.

本研究で利用するパケットフィルタを実現する技術として, Atomic Predicate Verifier(APV)[3], Binary Decision Diagrams(BDD)[18], AP Classifier[19], Multi-valued Decision Diagram(MDD)[4]などがある. パケットのフィルタリングには, パスに対応するパケットヘッダを特定し, ヘッダに基づきパケットをラインレートで高速に分類する必要がある. APVでは, ネットワーク内での経路などパケットの振る舞い毎にパケットを分類することを可能にする技術であり, BDD, AP Classifier, MDD は, 高速な分類を可能にする技術である.

7. おわりに

本稿では, ネットワークを流入するパケットのパスを統合・分割し, パケット廃棄率を計測することで, 最小の計測コストでネットワークの障害箇所を検知し, 推定する技術を提案した. 提案法では, 廃棄率計測のために設置するパケットフィルタの

数をコストと捉え, フィルタ数を最小化する. 実ネットワークの構成データを利用して, 障害箇所の推定精度を検証することで, FIB に関する障害箇所をアクション単位で一意に特定できることを示した. また, FIB と ACL に関する障害箇所を特定する場合には, 大部分の障害をポート単位まで一意に推定できることを示した. パス毎にパケットを分類する障害箇所推定手法に比べて, パスの統合・分割により同等の精度を実現しながら, 計測に必要なフィルタ数をほぼ半数に抑えることを確認した.

提案法では, パスの統合により出口側のパケット分類を統一し, フィルタ数を削減している. これに加えて, 入口側でもパスの統合を行うことで, よりフィルタ数を削減できる可能性がある.

謝 辞

本研究の一部は, 電気通信普及財団による助成を受けたものである.

文 献

- [1] C. Guo, L. Yuan, D. Xiang, Y. Dang, R. Huang, D. Maltz, Z. Liu, V. Wang, B. Pang, H. Chen, Z.-W. Lin, and V. Kurien, "Pingmesh: A Large-Scale System for Data Center Network Latency Measurement and Analysis," ACM SIGCOMM Computer Communication Review, vol.45, no.4, pp.139–152, Oct. 2015.
- [2] H. Zeng, P. Kazemian, G. Varghese, and N. McKeown, "Automatic Test Packet Generation," IEEE/ACM Transactions on Networking, vol.22, no.2, pp.554–566, April 2014.
- [3] H. Yang and S.S. Lam, "Real-time Verification of Network Properties Using Atomic Predicates," 2013 21st IEEE International Conference on Network Protocols (ICNP 2013), pp.1–11, Goettingen, Germany, Oct. 2013.
- [4] T. Inoue, T. Mano, K. Mizutani, S.-i. Minato, and O. Akashi, "Rethinking Packet Classification for Global Network View of Software-Defined Networking," Proceedings of 2014 IEEE 22nd International Conference on Network Protocols (ICNP 2014), pp.296–307, Raleigh, NC, USA, Oct. 2014.
- [5] V.V. Vazirani, Approximation Algorithms, Springer, 2013.
- [6] P. Kazemian, G. Varghese, and N. McKeown, "Header Space Analysis: Static Checking for Networks," 9th Symposium on Networked Systems Design and Implementation (NSDI 2012), San Jose, CA, USA, April 2012.

表 4 MIB を利用した MU での障害箇所推定精度

データセット	Stanford				Internet2				Purdue	
	FIB		FIB/ACL		FIB		FIB/ACL		FIB	
障害箇所推定技術	従来法	提案法	従来法	提案法	従来法	提案法	従来法	提案法	従来法	提案法
推定対象アクション数	58	76	132	188	140	149	304	313	58	896
故障パターン数	58	76	132	188	140	149	304	313	526	896
MU 数	319	67	379	104	11159	126	11183	150	3015	724
フィルタ数	638	430	758	642	22318	11616	22366	11640	6030	4225
識別不能なアクションペア数	0	0	49	53	0	0	140	140	0	0
障害アクションを一意に推定できる確率	100%	100%	26%	44%	100%	100%	8%	11%	100%	100%
障害ポートを一意に推定できる確率	100%	100%	94%	96%	100%	100%	100%	100%	100%	100%
障害スイッチを一意に推定できる確率	100%	100%	100%	100%	100%	100%	100%	100%	100%	100%

表 5 アクション、ポート、及びスイッチ毎に障害が発生する場合の障害箇所推定精度

データセット	Stanford				Internet2				Purdue	
	FIB		FIB/ACL		FIB		FIB/ACL		FIB	
障害箇所推定技術	従来法	提案法	従来法	提案法	従来法	提案法	従来法	提案法	従来法	提案法
障害パターン数	73	73	203	203	149	149	453	453	641	641
MU 数	319	50	379	68	11159	118	11183	142	3031	394
フィルタ数	638	368	758	437	22318	11277	22366	11301	6062	3426
識別不能なアクションペア数	0	0	135	135	0	0	372	372	0	0
アクションまで絞込できる確率	100%	100%	28%	28%	100%	100%	13%	13%	100%	100%
ポートまで絞込できる確率	100%	100%	95%	95%	100%	100%	100%	100%	100%	100%
スイッチまで絞込できる確率	100%	100%	100%	100%	100%	100%	100%	100%	100%	100%

- [7] “Internet2 - Forwarding Information Base”.
<http://vn.grnoc.iu.edu/Internet2/fib/>
- [8] Y.-W.E. Sung, X. Sun, S.G. Rao, G.G. Xie, and D.A. Maltz, “Towards Systematic Design of Enterprise Networks,” IEEE/ACM Transactions on Networking, vol.19, no.3, pp.695–708, 2011.
- [9] S. Agrawal, K.V.M. Naidu, and R. Rastogi, “Diagnosing Link-level Anomalies Using Passive Probes,” Proceedings of the 26th IEEE International Conference on Computer Communication (INFOCOM 2007), pp.1757–1765, Anchorage, AK, USA, March 2007.
- [10] M. Coates and R. Nowak, “Network Loss Inference Using Unicast End-to-End Measurement,” Proceedings of the 13th ITC Specialist Seminar on IP Traffic, Measurement, and Modeling, Monterey, CA, USA, Sept. 2000.
- [11] N. Duffield, “Network Tomography of Binary Network Performance Characteristics,” IEEE Transactions on Information Theory, vol.52, no.12, pp.5373–5388, Dec. 2006.
- [12] H.X. Nguyen and P. Thiran, “Network Loss Inference with Second Order Statistics of End-to-end Flows,” Proceedings of the 7th ACM SIGCOMM Conference on Internet Measurement (IMC 2007), p.227, San Diego, CA, USA, Oct. 2007.
- [13] M.H. Firooz and S. Roy, “Network Tomography via Compressed Sensing,” Proceedings of 2010 IEEE Global Telecommunications Conference (GLOBECOM 2010), Miami, FL, USA, Dec. 2010.
- [14] T. Matsuda, M. Nagahara, and K. Hayashi, “Link Quality Classifier with Compressed Sensing Based on ℓ_1 - ℓ_2 Optimization,” IEEE Communications Letters, vol.15, no.10, pp.1117–1119, Oct. 2011.
- [15] V.N. Padmanabhan, L. Qiu, and H.J. Wang, “Server-based Inference of Internet Link Lossiness,” Proceedings of the 22nd IEEE International Conference on Computer Communication (INFOCOM 2003), pp.145–155, San Francisco, CA, USA, March 2003.
- [16] Y. Tsang, M. Yildiz, P. Barford, and R. Nowak, “Network Radar: Tomography from Round Trip Time Measurements,” Proceedings of the 11th ACM SIGCOMM Conference on Internet Measurement (IMC 2004), pp.175–180, Sicily, Italy, Oct. 2004.
- [17] Y. Tsang, M. Yildiz, P. Barford, and R. Nowak, “On the Performance of Round Trip Time Network Tomography,” Proceedings of 2006 IEEE International Conference on Communications (ICC 2006), pp.483–488, Istanbul, Turkey, June 2006.
- [18] R.E. Bryant, “Graph based algorithms for Boolean function manipulation,” IEEE Transactions on Computers, vol.358, no.8, pp.677–691, Aug. 1986.
- [19] H. Wang, C. Qian, and S.S. Lam, “Practical Network-wide Packet Behavior Identification by AP Classifier,” Proceedings of the 11th ACM SIGCOMM Conference on Emerging Networking Experiments and Technologies (CoNext 2015), Heidelberg, Germany, Dec. 2015.