RISE4.0 の性能改善手法についての考察

原 拓也†，河合 栄治†，石井 秀治†，寺田 直美†，金海 好彦‡，斉藤 修一‡

†情報通信研究機構 〒105-0123 東京都小金井市貫井北町 4-2-1

‡日本電気株式会社 〒108-8001 東京都港区芝 5-7-1

E-mail: †t-hara@nict.go.jp

あらまし NICT では，２００９年より SDN/OpenFlow テストベッド RISE を構築, 運用している. RISE3.0 では，マルチテナント化とトポロジの仮想化を実現した．しかし，RISE3.0 では物理スイッチに実装されている機能に依存する制限（ユーザによる VLAN や OpenFlow Ver1.3 が使用不可）がある．そこで，RISE4.0 では仮想スイッチを利用した RISE アーキテクチャを提案し実装した．しかし，RISE4.0 ではネットワークスループットが非常に低くなる問題が見つかった．本論文では，RISE4.0 の性能改善のための方法を報告する．

キーワード SDN, OpenFlow, 仮想スイッチ，テストベッド，Lagopus

1. まえがき

外部プログラムによりネットワーク機器を制御可能にする SDN（Software-Defined Networking）のコンセプトが浸透し，OpenFlow [1]等の SDN 実装技術が研究開発で広く利用されている．ネットワーク技術の研究開発では，大規模な環境で成果の有効性を検証することが重要であるが，個々の研究開発プロジェクトがそうした環境を専用で用意することは難しい．そこで NICT では，研究者，学生，エンジニア，オペレータなどが利用可能な大規模 SDN/OpenFlow テストベッド RISE（Research Infrastructure for large-Scale network Experiments）[2]を構築し, 2009 年より運用している．RISE は多くのプロジェクトで利用され, 研究開発のインフラとして貢献した．また，RISE の運用を通じて得られた知見は R&E ネットワークのコミュニテイにおいて共有を図った．

NICT は, 研究開発を通じて RISE の機能拡張を図ってきた．RISE3.0 [3, 4]と呼ばれる現行の RISE は，3 つの特徴を有している．広域性，マルチユーザ，トポロジの仮想化である．

1 つ目の RISE の広域性とは，いわゆる広域ネットワーク上で SDN 機能が利用可能であることを指す. そのために，NICT が運用している広域ネットワークである JGN [5]上で RISE をオーバレイとして構築するための要件を検討した．具体的には，広域ネットワーク（アンダーレイ）側での仮想回線構成や，MAC アドレス学習回避，ループトラフィックに代表される障害発生時の対応方法等が挙げられる．

2 つ目の RISE のマルチユーザとは, 複数のユーザが SDN テストベッドを共有し，同時並行的に試験可能であることを指す．必要となるテストベッドリソースの量や期間はユーザ毎に異なることから，マルチユーザにより多重効果を得ることは重要である．このマルチユーザを実現するためには, 各ユーザの SDN 環境を仮想的に独立させる必要があり，そのための既存の仕組みには FlowVisor [6]や OpenVirteX [7]がある. しかし，これらの方式はコントローラとネットワーク機器の間に設置されるプロキシを用いており，それが単一障害点となる問題がある．そこで RISE では，ネットワーク機器を複数の仮想インスタンスとして動作させる仮想化機能を用いる方針とした．各ユーザのコントローラはネットワーク機器の仮想インスタンスに直接接続できるため，プロキシを用いることなくマルチユーザを実現している．

3 つ目の RISE のトポロジ仮想化とは, ユーザが望む SDN 環境のトポロジを自由に提供可能にすることである. 先に述べた通り，RISE は広域ネットワーク JGN の上に構築されており，JGN が提供する仮想回線を組み合わせて SDN 環境のトポロジを自由に構築する仮想化機能を開発した．

NICT は，これらの広域性，マルチユーザ，トポロジの仮想化を実現した SDN テストベッド環境を RISE3.0 として 2014 年より運用している. 一方で, この RISE3.0 が抱える問題も明らかになってきている．例えば，ネットワーク機器で設定可能な仮想インスタンスの数が限られており，それを超えるユーザを収容できないことが挙げられる．また，接続拠点の数だけ物理ポートが必要であることから，拠点数を増やすには，高速かつ搭載ポート数の多い高価な SDN 対応ネットワーク機器が必要となること，大規模な（接続拠点の多い）トポロジを構成できないといった問題もある．

これらの問題を検討した結果，ネットワーク機器として汎用サーバ上で動作するソフトウェアスイッチ（仮想スイッチ）を導入することで，スケールアウト的に解決が可能であることが判明した．そこで，仮想スイッチをベースにした SDN テストベッドを RISE4.0

[8]として提案し，その実現に向けた研究開発を行っている．本論文では，その中で取り組んだ仮想スイッチの性能問題について述べる．

RISE4.0 では，RISE3.0 で使用していたハードウェア・ネットワーク機器を仮想スイッチに置き換えている．そのため，ネットワークスループットが RISE3.0 と比較して低くなることは想定していた．ところが，実際に性能の評価を行ったところ，想定よりも著しく低い結果となった．原因を調べたところ，仮想スイッチと VM 間の接続に問題があることを突き止めた．RISE4.0 の構成では，仮想スイッチと VM 間でパケットを交換する際に，カーネルとユーザスペース間でコンテキストスイッチが頻発することによる性能低下が起きていた．そこで，VM と仮想スイッチ間の接続方法について調査を行い，接続方法を変更して再度性能測定を実施した．その結果，スループットが改善されたことが確認できた．

以下，本論文の構成を示す．2 節と 3 節において RISE3.0 および RISE4.0 について説明する．続いて，4 節において RISE4.0 の性能問題について整理し，5 節でその改善方法，6 節で性能評価結果を示す．7 節で提案手法の課題等を考察し，8 節でまとめを述べる．

2. RISE3.0 について

2.1. RISE の特徴

本節では，RISE3.0 の特徴であるマルチユーザとトポロジの仮想化機能について述べる．

マルチユーザ

前出のように，ユーザにより必要となるテストベッドリソースの量や期間が異なることから，マルチユーザにより多重効果を得ることは重要である．マルチユーザの実現には，各ユーザのスライスは他のユーザのスライスから独立している必要がある．

RISE3.0 では，単一の物理 OpenFlow スイッチ（OFS）を複数の仮想インスタンスとして動作させる Virtual Switch Instance (VSI 機能)を利用してマルチユーザ機能を実現している．

この VSI 機能により，1 台の OFS が複数の仮想 OFS として動作する．ただし，利用機器の仕様からくる制限より，同時に利用できる仮想 OFS は最大 16 台である．

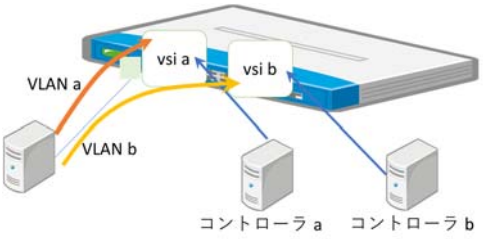

図 1 VSI のイメージ

図 1 に示すように，各仮想 OFS には，異なる VLAN 値が割り当てられ，その値により他の仮想 OFS から分離されている．具体的には VLAN 値 a を持つパケットは，a に対応する仮想 OFS a で処理され，値 b を持つパケットは対応する別の仮想 OFS b で処理される．

トポロジの仮想化

ユーザが要求するネットワークトポロジはそれぞれ異なる（ユーザ A はループトポロジ，ユーザ B はツリートポロジなど）．一方で，JGN は固定されたトポロジを持つ広域ネットワークである．そのため，JGN のトポロジに制限されず，ユーザ毎にトポロジを構成できる必要がある．

そこで，RISE3.0 では，トポロジ仮想化機能を実装した．具体的には， ユーザ用仮想 OFS 間の隣接をオーバレイとして構成される論理リンクによって実現した．RISE3.0 では，この論理リンクにより様々なトポロジを作成できる．

2.2. RISE3.0 のアーキテクチャ

RISE3.0 はユーザに提供する OpenFlow ネットワークと広域でのパケット伝送を行う JGN との間にユーザ環境管理用の OpenFlow ネットワークを配備した階層構造をとる．RISE3.0 のアーキテクチャを図 2 に示す．

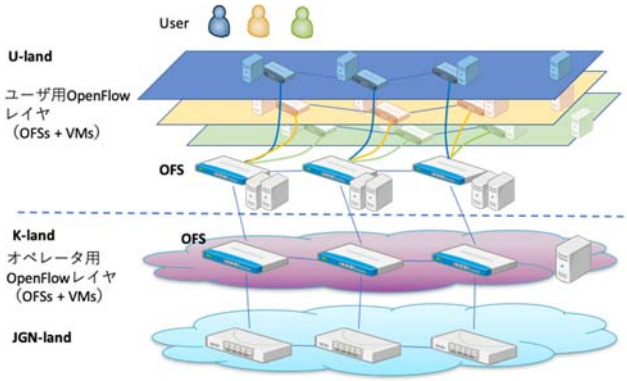

図 2 RISE3.0 アーキテクチャ

RISE3.0 の上位層と中間層は OFS により構成される．上位層はユーザのための複数のスライスが存在し，ユーザコントローラが自身に割り当てられたスライスを自由に制御できる．本論文では，この層を U-land と呼び，U-land の OFS を U-land-OFS と呼ぶ．U-land-OFS は VSI モードで動作する．中間層は RISE オペレータのための OpenFlow ネットワークであり，オペレータのコントローラが制御する．本論文では，この層を K-land と呼び，K-land の OFS を K-land-OFS と呼ぶ．K-land-OFS は U-land-OFS とは異なり，通常の（非 VSI モードの）OpenFlow スイッチとして動作している．

2.3. RISE3.0 の具体的構成

RISE3.0 の各層のスイッチ間の接続関係を明確にするために，RISE3.0 の具体的構成を図 3 に示す．

RISE3.0 では，ユーザ毎の論理リンクによる接続対向拠点を U-land-OFS と K-land-OFS 間の物理接続リンクにより決定している．ユーザが最大で RISE が有するすべての拠点への接続を求める可能性を想定しているため，このリンクの必要数は最大で対向拠点数（全拠点数-1）である．

この物理接続リンクは，ユーザ間で共有されている．例えば，図 3 の左のリンクの U-land-OFS のポート P は，ユーザ A には拠点 X の U-land−OFS と仮想的につながっているポート P_X，ユーザ B には拠点 Y の U-land-OFS と仮想的につながるポート P_Y として割り当て可能である．

次に，K-land-OFS に必要なポート数について述べる．K-land-OFS の必要ポート数は，U-land-OFS とのリンクを収容するためのポート数と，JGN を構成するスイッチ（以降，JGN-Switch）と接続するためのポート数の合計値である．

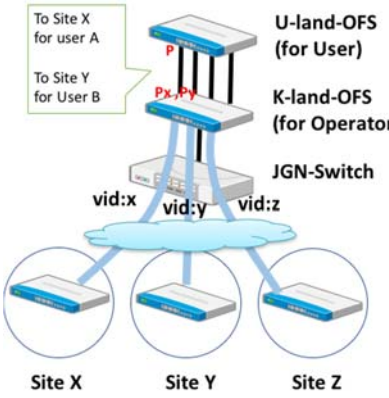

図 3 RISE3.0 の基本構成

2.4. RISE3.0 の動作

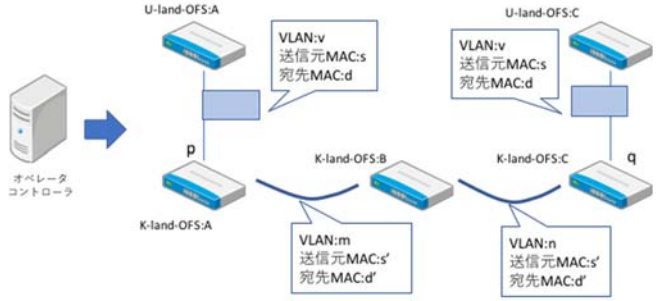

図 4 RISE3.0 の動作例

RISE3.0 の動作例を図 4 に示す．K-land-OFS を制御するオペレータコントローラは， K-land-OFS 間の論理パスを設定により把握している（後述するように，実際のフロー設定は当該フローのパケット発生時に行う）．この論理パスはユーザ識別 VLAN 及び，ユーザ毎に設定されたトポロジで定義される．U-land-OFS:A から U-land-OFS:C へパケットを送信すると，K-land-OFS:A が受信し，K-land-OFS:A は packet in メッセージをオペレータコントローラに送信する．オペレータコントローラは，受信パケットの VLAN 値 v（ユーザ識別 VLAN）および受信ポート p を特定する．v と p から対向の K-land-OFS:C の物理ポート q を決定できる（ユーザ毎の論理パスがあらかじめ設定されているため）．オペレータコントローラは，K-land−OFS:A のポート p から K-land-OFS:C のポート q までの経路を計算し，該当パケットを転送するフローを途中経路にある K-land-OFS に設定する．このとき，一意な仮想リンク ID を生成し，送信元/宛先 MAC アドレスフィールドを書き換えるアクションを K-land-OFS:A に設定する．途中にある K-land-OFS:B は，仮想リンク ID をマッチ条件として，VLAN 値を n に書き換えて送信する．各 K-land-OFS 毎に VLAN 値が割り当てられており，この VLAN 値により隣接する K-land-OFS にパケットが送信されることになる．最後に，オペレータコントローラは，書き換えた送信元/宛先 MAC アドレスの情報から元のユーザ識別 VLAN 値および MAC アドレスに戻すアクションを，K-land-OFS:C に設定する．

2.5. RISE3.0 の課題

RISE3.0 は広域ネットワーク上で OpenFlow を用いたネットワーク制御の実証ができる数少ない環境として広く利用された．一方で，その運用を通じて以下の 4 件の課題があることが判明した．

- ユーザは VLAN フィールドを使用した実験を行うことができない．これは，ユーザの識別に VLAN を使用しているためである．
- RISE3.0 の拠点数が物理 OFS のポート数に制約される．これは，ユーザ所望のトポロジに対して，U-land-OFS と K-land-OFS 間の物理ポートは対向拠点数分必要となるためである．
- ユーザは，OpenFlow のバージョンが 1.0 しか使えない．これは，マルチユーザのために使用している OFS の VSI 機能の制限によるものである．
- 同時に利用できるユーザ数に制限がある．これも，VSI 機能による制約によるものである．

これらの課題について，次節で説明する RISE4.0 で解決することを目指している．

3. RISE4.0 について

3.1. RISE4.0 の特徴

RISE4.0 は RISE3.0 の特徴でもあるマルチユーザとトポロジ仮想化機能を維持しつつ，汎用サーバ上でソフトウェアスイッチ（仮想スイッチ）を利用することによって RISE3.0 での問題を解決する．したがって，機能的な変更はほとんどない．仮想スイッチにより，論理ポートを自由に設定することができるようになり，ハードウェアスイッチに起因する制約から解放される．

3.2. RISE4.0 の具体的構成

RISE4.0 のアーキテクチャを図 5 に示す．汎用サーバ上で動作する U-land-OFS と K-land-OFS，ならびに物理 OpenFlow スイッチにより構成される．U-land-OFS はユーザに提供する OpenFlow スイッチ層であり，図の A,...,N が該当する．K-land-OFS はオペレータのための OpenFlow スイッチである．これらの K-land-OFS および U-land-OFS は lagopus [9] がサーバ上で動作する．

U-land-OFS と K-land-OFS 間は vlink による仮想リンクで接続し，この仮想リンクをユーザ毎かつ宛先毎に作成する．そのため，オペレータコントローラはポート情報よりユーザと宛先拠点を一意に特定できる．

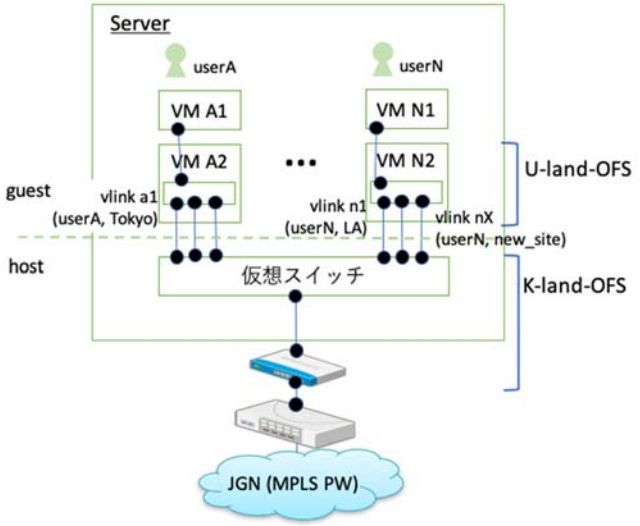

図 5 RISE4.0 アーキテクチャ

各拠点では，この汎用サーバを複数台稼働させることができ，汎用サーバは物理 OpenFlow スイッチにより広域ネットワークを提供する JGN と接続される．この物理 OpenFlow スイッチは，RISE3.0 の K-land-OFS の機能のうち，仮想リンクに関する VLAN 値の管理及び書き換えのみを担っている．

次に，RISE3.0 の課題が解決されていることについて述べる．まず，RISE4.0 では，VLAN フィールドをユーザに開放可能である．RISE3.0 では，ユーザごとに VLAN 値を割り当て，VLAN 値を用いてユーザを識別していた．RISE4.0 では，ソフトウェアスイッチ上でユーザと宛先に応じて論理ポートを多数（実用上十分な数）作成できるため，VLAN によりユーザを識別する必要がない．また，この論理ポートを多数作成できることにより，拠点数の制約も解決される．また，lagopus は OpenFlow バージョン 1.3 をサポートしており，利用可能である．最後に，ユーザ数の制約については，サーバの性能及びリソース量に応じて収容できる VM の数が決まることから，RISE3.0 のようなネットワーク機器の仕様で定められている固定的な上限はなくなっている．

4. RISE4.0 性能問題

我々は，開発した RISE4.0 をサービス開始するために，事前検証を行ってきた．その中でネットワークスループットの測定を行ったところ，性能が著しく低かった．

本節では，性能劣化箇所の特定及び性能劣化の原因を明らかにする．

4.1. 性能劣化箇所特定の調査

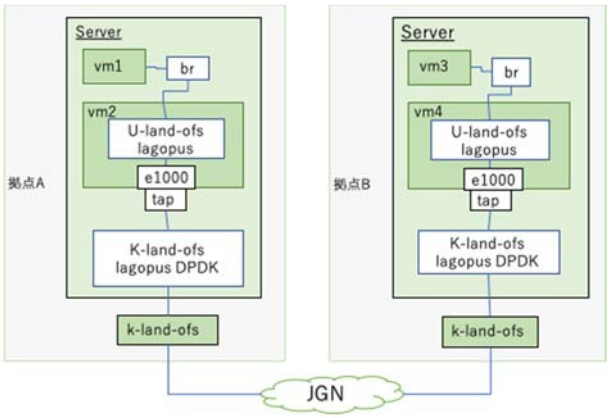

図 6 RISE4.0 構成図

図 6 は，性能測定時の構成図である．ネットワークスループットの測定は，vm1 - vm 3 間を iperf3 コマンドを使用して実施した．その結果 TCP で約 1 2 Mbps と非常に遅いことが判明した．NIC エミュレーションである e1000 を使用した場合，パケットの送受信処理で性能が出ないことは既知である [10]. そのため, vm2 の U-land-OFS とサーバ 1 のホスト OS 上の K-land-OFS の間がボトルネックとなっていると見当をつけて，該当箇所のスループットを測定した．測定時の構成を図 7 に示す．本構成は vm2 と K-land-OFS 間のスループットを測定することを主とするために, 他の VM や U-land-OFS の lagopus を削除している．なお，サーバ 1 – サーバ 2 間の NIC は 1 0 G インタフェースであり，この区間の性能に問題がないことは確認済である．

本構成でスループットを測定したところ，TCP で約 1 4 Mbps であった. この結果から, ボトルネック箇所は lagopus – VM 間であると特定した．

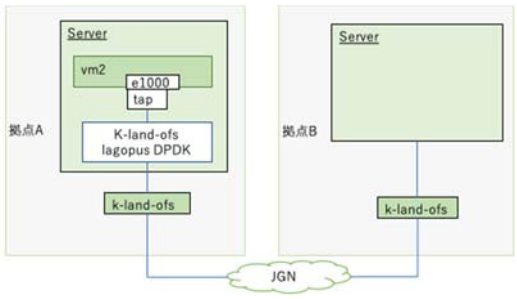

図 7 性能測定用構成図

4.2. 性能劣化の理由

前節で示した通り，仮想 NIC の e1000 がネットワークのボトルネックとなることは既知である.図 8 は仮想 NIC e1000 を利用したときのパケット送信処理を示している．この図の点線を矢印が越えるとコンテキストスイッチが発生する．この図から多くのオーバヘッドがあることが分かる.

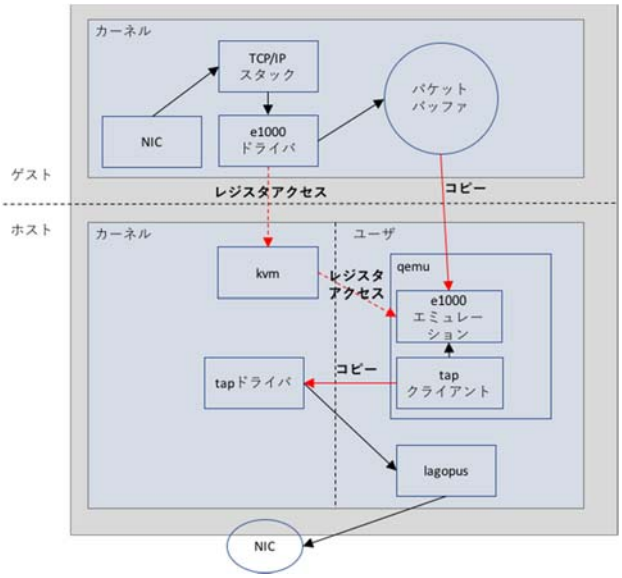

図 8 e1000 送信処理

実際にホスト OS でコンテキストスイッチが頻発しているかを vmstat コマンドにより確認した．結果を以下の表に示す.

表 1 コンテキストスイッチ数

コンテキストスイッチ数（1 秒あたり）	iperf 実行前	約 600
	iperf 実行中	約 10000

表 1 の通り，iperf3 コマンド実行中にコンテキストスイッチ数が約 14 倍増加した.

5. 改善方法

KVM の仮想 NIC は以下のものがある.

- NIC エミュレーション（e1000,...）
  qemu [11]を使って実在の NIC をエミュレートする．仮想レジスタへのアクセスのたびに VMX non-root モードと VMX root モードの切替が発生する.
- virtio-net（準仮想化 IO）
  IO 仮想化フレームワーク「virtio」を用いてパケットの入出力を行う．qemu とゲスト OS 間で共有されたキューを通じてデータの入出力を行う．キューに対してデータの入出力を行うことで，レジスタアクセスが不要となり，モードの遷移回数が削減される.
- vhost-net
  ゲスト OS とホストのカーネル間で共有メモリを用いることで, qemu への切替を行うのに伴うオーバヘッドを削減
- vhost-user
  ゲスト OS とホストのユーザスペース間で共有メモリを用いることで，ユーザスペースとカーネル切替が削減される.

RISE4.0 では，VM と lagopus がユーザスペース内に存在するため，可能な限りコンテキストスイッチを削減するために vhost-user [12, 13]を使用する設計に変更することにした.

5.1. Vhost-user

Vhost-user は，vhost client と vhost backend で構成され，ユーザスペース内に存在する 2 つのアプリケーション間でやり取りされるデータは共有メモリ上で共有される.

RISE4.0 では，lagopus が vhost backend，VM が vhost client として動作させることにより，lagopus – VM 間のパケット送受信によるコンテキストスイッチ数を削減させる.

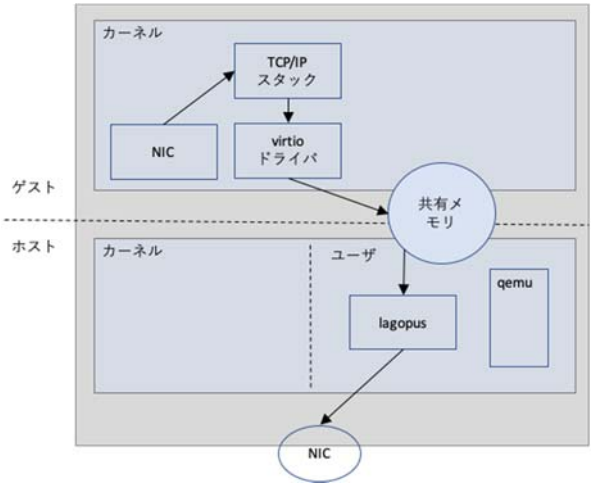

図 9 vhost-user 使用時の送信処理

実際にコンテキストスイッチ数の削減を確認するために，先ほどと同様の手順で測定を行った結果を以下の表に示す.

表 2 vhost-user 使用時のコンテキストスイッチ数

コンテキストスイッチ数（1 秒あたり）	iperf 実行前	約 500
	iperf 実行中	約 3000
スループット	約 6 Gbit/sec	

上の表の通り，vhost-user 変更前と比較してコンテキストスイッチ数が約 1 / 3 になり，ネットワークスループットも改善されたことが確認できた.

6. 性能評価

図 6 に示した RISE4.0 の構成の VM-lagopus 間を vhost-user に変更し，vm1−vm3 間のネットワーク性能測定を実施した．使用した計算機の仕様は，CPU が Xeon E5 2630（8 コア），メモリ 32GB，10Gbit イーサネットであり，Ubuntu 18.04，Lagopus 0.2.11，Libvirtd 4.0.0 を

動作させた.

図 10 に RISE4.0 変更前と変更後の性能評価結果を示す. ネットワークスループットが改善されていることを確認できた.

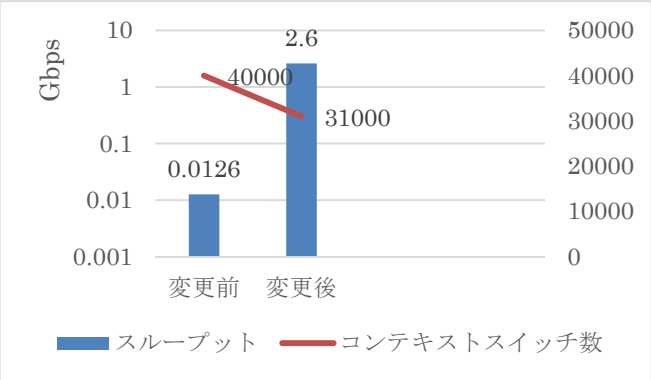

図 10 性能測定結果

7. 考察

RISE4.0 が課題としていた性能問題は，同一サーバ内の VM と lagopus 間でのパケット送受信におけるコンテキストスイッチ数を低減することで，ネットワークトラフィックの性能が改善することが分かった. コンテキストスイッチ数の削減には，VM-lagopus 間の接続に vhost-user を用いることで実現した.

一方で，RISE4.0 の高速化のために使用している技術（dpdk や vhost-user の使用）は，RISE3.0 の運用で培った監視技術が適用できなくなる. 従来運用者が各接続点のパケットを保存しておくことで，障害解析を行ってきた. dpdk や vhost-user を使用したインタフェースでは従来のパケットキャプチャが不可能なため，解析が困難になる課題がある.

8. まとめ

本稿では，RISE4.0 のネットワークトラフィック性能問題改善の検証を行った. 検証の結果, VM と lagopus 間を Vhost-user に変更することで，大幅に性能が改善されたことを確認した. 今後は，全国展開済のサーバに本変更内容を適用し，RISE4.0 のサービスとしてユーザへ提供できるように進めていく予定である.

参考文献

[1] N. McKeown, T. Anderson, H. Balakrishnan, G. Parulkar, L. Peterson, J. Rexford, S. Shenker and J. Turner, "OpenFlow: enabling innovation in campus networks," *ACM SIGCOMM Computer Communication Review,* vol. 38, no. 2, pp. 69-74, 4 2008.

[2] Y. KANAUMI, S.-i. Saito, E. KAWAI, S. ISHII, K. KOBAYASHI and S. SHIMOJO, "Rise: A wide-area hybrid openflow network testbed," *IEICE TRANSACTIONS on Communications,* vol. 96, no. 1, pp. 108-118, 2013.

[3] 石井秀治, 河合栄治, 金海好彦, 齋藤修一, 高田智明, 小林和真 , 下條真司, “RISE 3.0 用コントローラに関する一検討（ネットワークシステム）,” *信学技報*, 第巻 113, 第 89, pp. 7-12, 2013.

[4] S. Ishii, E. Kawai, Y. Kanaumi, T. Takata, K. Kobayashi and S. Shimojo, "A study on designing OpenFlow controller RISE 3.0," in *19th IEEE International Conference on Networks (ICON)*, Singapore, 2013.

[5] “JGN ウェブサイト,” [オンライン]. Available: https://www.jgn.nict.go.jp. [アクセス日: 18 9 2018].

[6] R. Sherwood, G. Gibb, K.-K. Yap, G. Appenzeller, M. Casado, M. Nick and P. Guru, "FlowVisor: A Network Virtualization Layer," OpenFlowSwitch.org, 2009.

[7] A. Al-Shabibi, M. D. Leenheer, M. Gerola, A. Koshibe, G. Parulkar, E. Salvadori , B. Snow, “OpenVirteX: make your virtual SDNs programmable,” 著: *HotSDN '14*, Chicago, Illinois, USA, 2014.

[8] 伊藤暢彦, 金海好彦, 齋藤修一, 鈴木一哉, 河合栄治 , 下條真司, “広域展開に向けた RISE アーキテクチャの検討,” *信学技報, NS2015-108,* 第 巻 115, 第 251, pp. 109-114, 10 2015.

[9] Y. Nakajima, T. Hibi, H. Takahashi, H. Masutani, K. Shimano , M. Fukui, “ Scalable, High-performance, Elastic Software OpenFlow Switch in Userspace for Wide-area Network,” 著: *Open Networking Summit 2014*, 2014.

[10] “仮想化環境におけるパケットフォワーディング,” [オンライン]. Available: https://www.nic.ad.jp/ja/materials/iw/2011/proceedings/s09/s09-02.pdf.

[11] “ QEMU, ” [オンライン]. Available: https://www.qemu.org.

[12] “Vhost-User Feature for QEMU,” [オンライン]. Available: http://www.virtualopensystems.com/en/solutions/guides/snabbswitch-qemu/ . [アクセス日: 18 9 2018].

[13] “Documentation/vhost-user-ovs-dpdf,” [オンライン]. Available: https://wiki.qemu.org/Features/vhost-user-ovs-dpdk. [アクセス日: 18 9 2018].