

OpenFlow パイプライン処理可視化による OpenFlow コントローラ開発支援システム

嶋谷修一朗[†] 井口信和[‡]

近畿大学理工学部情報学科[†] 近畿大学情報学研究所[‡]

1. 序論

仮想化技術やクラウドサービスの普及に伴い、ネットワークの管理が複雑化している。そこでネットワークの集中制御が可能な Software-Defined Networking(以下, SDN)がデータセンターを中心に利用されている。SDN はネットワークの転送機能と制御機能を分離し、ソフトウェアで制御機能を実装するアーキテクチャであり、ネットワークの管理を簡素化することが期待されている¹⁾。SDN を実現するプロトコルに OpenFlow²⁾がある。OpenFlow はコントローラが、スイッチのフローテーブルを操作することで、パケットを制御するプロトコルである。スイッチは複数のフローテーブルでパケットを処理でき、この処理は OpenFlow パイプライン処理と呼ばれる。

OpenFlow コントローラは複雑なシステムとなるため、コントローラ開発ではネットワークのテストやデバックが重要である³⁾。OpenFlow ネットワークのデバックではコントローラのログを解析し、Mininet や ping コマンド, tcpdump のようなツールを用いて情報を収集する。コントローラ開発者はそれぞれのツールを使いこなし、得られた情報を統合する必要があるため、開発者の負担は大きい。

そこで本研究では、OpenFlow コントローラ開発時におけるデバックの負担軽減を目的に、パケットの追跡を自動化し OpenFlow パイプライン処理を可視化するコントローラ開発支援システムを開発した。これによりネットワーク全体の動作の把握が容易となり、デバック時における時間的コストや各デバックツールの学習コストの削減が期待できる。

2. 研究内容

本システムのシステム構成を図1に示す。仮想ネットワーク部は Mininet を用いて仮想ネットワ

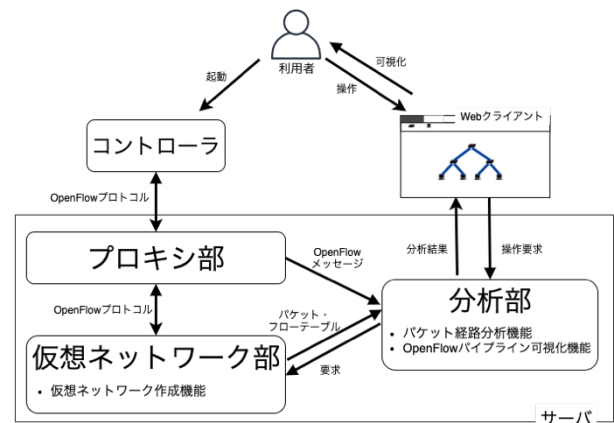


図1 システム構成

ークを作成し、パケットやフローテーブルを収集する。プロキシ部はコントローラと仮想ネットワーク部を仲介し、OpenFlow メッセージを収集する。分析部は仮想ネットワーク部とプロキシ部から得られた情報を統合し、パケットの経路を分析する。Web クライアントは、利用者がシステムを操作し、得られた情報を可視化するための GUI を提供する。また Web クライアントでは CLI が使用できる。この CLI は Mininet の CLI を用いており、Mininet に実装されているコマンドやシェルで実行可能なコマンドが使用できる。

本システムの利用方法を説明する。まず利用者は、システムと OpenFlow プロトコルが接続可能な状態でコントローラを起動する。次に仮想ネットワーク作成機能を用いて、デバックで使用する仮想ネットワークを構築する。そして Web クライアント上に用意された CLI で ping コマンドを用いてパケットを送信する。システムはパケット経路分析機能でパケットの経路と各スイッチでの処理を分析する。その後、OpenFlow パイプライン処理可視化機能により、Web クライアント上にパケットの経路や各スイッチでの処理内容が可視化され、利用者はこれらの情報を確認できる。

2.2 仮想ネットワーク作成機能

仮想ネットワーク作成機能は、利用者がコントローラをデバックするための仮想ネットワークを作成できる機能である。利用者は Web クライアント上で、仮想ネットワークを直感的な操

OpenFlow Controller Development Support System

Visualizing OpenFlow Pipeline Processing

[†]Shuichi SHIMATANI, Nobukazu IGUCHI, Department of Informatics, Faculty of Science and Engineering, Kindai University

[‡]Nobukazu IGUCHI, Cyber Informatics Research Institute, Kindai University

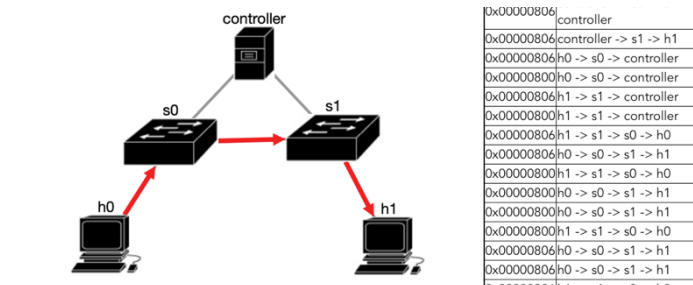


図2 パケット経路の可視化

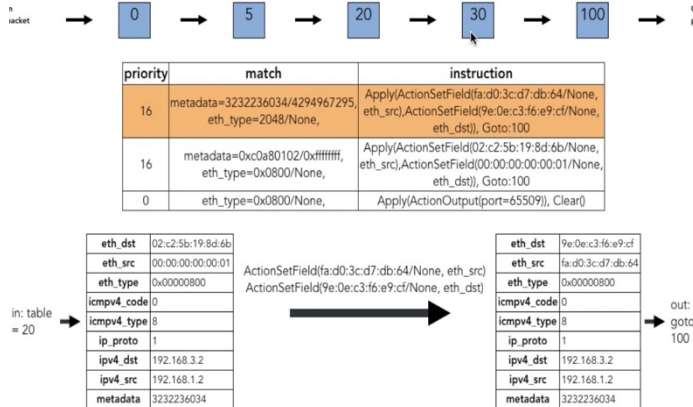


図3 パイプライン処理の可視化

作で設定できる。設定後、システムは Mininet を用いて対応する仮想ネットワークを構築する。この機能ではホストとスイッチを設定でき、スイッチには Open vSwitch を使用する。

2.3 パケット経路分析機能

パケット経路分析機能は、パケットが通った経路を分析する機能である。この分析機能では、パケットの経路を分析するために、パケットとフローテーブル、トポロジの情報を収集する。パケットは、リングごとに TShark を用いて取得する。フローテーブルは、FlowMod メッセージを解析し、定期的に Open vSwitch の dump-flows コマンドを実行することで取得する。トポロジ情報は仮想ネットワーク作成機能から取得する。

パケットの経路を特定する方法について説明する。仮想ネットワーク上で収集された各パケットに対し、パケットが到達したスイッチでどのように処理されたかをフローテーブルの情報からシステム上で再現し、次に向かうスイッチを分析する。これをホストもしくはコントローラに到達するまで繰り返すことでパケットの経路を特定する。また、PacketOut メッセージについても同様の操作でパケットの経路を特定する。

2.4 OpenFlow パイプライン処理可視化機能

OpenFlow パイプライン処理可視化機能は、パケット経路分析機能により分析したパケットの経路情報と処理内容から、各パケットがスイッチ内部でどのように処理されたかを可視化する機能である。トポロジ情報とともに、パケット

の経路とフローテーブル、フィールドの書き換えの様子を可視化する。

図2にパケット経路の可視化画面を示す。図のように、右側に流れた経路の一覧を表示する。表示された経路の一つをクリックすると、中央のトポロジにパケットの経路が矢印で表示される。また経路上のスイッチをクリックすると、パイプライン処理の可視化画面が表示される。

図3にパイプライン処理の可視化画面を示す。図のように、パケットを処理したときのフローテーブルやマッチしたフローエントリ、適用されたアクション、パケットの更新を可視化する。またこの機能では ActionSet の更新も可視化する。

3. 実験

実験は、システムが正しく動作していることを確認する目的で、動作確認実験を実施した。動作確認には、複数のフローテーブルを操作するコントローラを使用する。まずシステムを起動し、システムを用いて仮想ネットワークを構築する。次にコントローラとシステムを接続し、ping コマンドを実行する。このときにフローテーブルとパケットを収集する。収集した情報から、パケットの経路やパイプライン処理を手動で分析する。最後にシステムが可視化した内容と手動で分析した内容を比較する。システムが表示したパケット及び ActionSet の更新やパケットの経路などが手動で分析した内容と一致していたことから、システムが正しく動作していることを確認した。

4. 結論

本研究では OpenFlow コントローラ開発時におけるデバックの負担軽減を目的に、パケットの追跡を自動化し OpenFlow パイプライン処理を可視化するコントローラ開発支援システムを開発した。本システムにより、デバック時の時間的コストや学習コストの削減が期待できる。

今後は、GroupTable や MeterTable への対応や可視化可能な情報を増やすことを予定している。

参考文献

- Kim, H. and Feamster, N.: Improving network management with software defined networking, *IEEE Communications Magazine*, vol. 51, no.2, pp.114-119, (2013).
- McKeown, N., Anderson, T., Balakrishnan, H., et al.: OpenFlow: Enabling innovation in campus networks, *ACM SIGCOMM Comput. Commun. Rev.*, vol. 38, no. 2, pp. 69-74, (2008).
- Scott, C., Wundsam, A., Raghavan, B., et al.: Troubleshooting blackbox SDN control software with minimal causal sequences, *SIGCOMM Comput. Commun. Rev.*, vol. 44, no. 4, pp. 395-406, (2014).