

コントローラの設定情報と論理トポロジの視覚的な関連付けを可能とする OpenFlow ネットワーク運用管理支援システム

藤田 紘生[†] 井口 信和[†]

近畿大学理工学部情報学科[†]

1. 序論

サーバ仮想化やクラウドサービスの普及に伴い、ネットワークの要件が変化している。従来のネットワークでは、ネットワーク機器ごとに設定を施すため、ネットワークの要件が変化する度に各機器を再設定する必要がある。そのため、従来のネットワークでは、ネットワークの要件の変化にすばやく対応することが困難な場合がある。この問題を解決できる技術の一つに、OpenFlow¹⁻⁴⁾がある。

OpenFlow ネットワークはコントローラにより制御され、その論理トポロジは動的に変化する。コントローラを設定するとき、人為的なミスを含め、誤った設定を施す場合があるため、設定内容が正しいことを確認する必要がある。このとき、コントローラの設定情報と論理トポロジの関係を把握できれば、設定ミスや、その原因をすばやく特定できる。そのため、コントローラの設定情報と論理トポロジの関係を容易に把握できる仕組みが有用となる。

そこで本研究では、コントローラの設定情報と論理トポロジの視覚的な関連付けを可能とする OpenFlow ネットワーク運用管理支援システム（以下、本システム）を開発する。本システムを用いることで、タッチディスプレイ上でコントローラの設定情報と論理トポロジの関係を視覚的に把握できる。さらに、タッチ操作によりコントローラを直感的に設定でき、設定変更に伴う論理トポロジの変化をタッチディスプレイ上で容易に把握できる。

2. 研究内容

本システムの構成を図 1 に示す。本システムは、クライアントアプリケーションとコントローラから構成され、REST API を用いて通信する。コントローラは、管理対象とする OpenFlow ネットワークと接続し、OpenFlow プロトコルを用いて制御する。以下、本システムの構成要素であるクライアントアプリケーションとコントローラの詳細を述べる。

2.1 クライアントアプリケーション

タッチディスプレイで動作するクライアントアプリケーションは、コントローラへ一定時間ごとに OpenFlow ネットワークの情報を要求する HTTP リクエストを自動的に送信する。その後、コントローラ

A System for Supporting Management of OpenFlow Network Operation by Visualization of Relations between Controller Setting Information and Logical Topology

[†]Hiroki FUJITA, Nobukazu IGUCHI, School of Science and Engineering, Kindai University

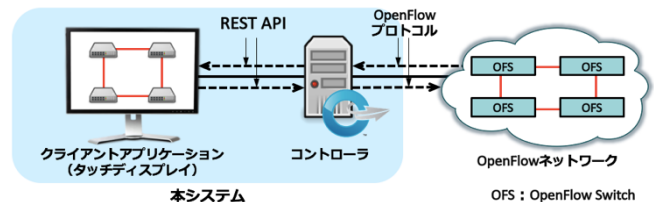


図 1 : システム構成

ラより受信する HTTP レスポンスから、OpenFlow ネットワークの情報を取得する。この情報を基に、クライアントアプリケーションは論理トポロジをタッチディスプレイに表示する。これにより、利用者はタッチディスプレイ上で論理トポロジを視覚的に把握できる。

また、タッチディスプレイに表示される論理トポロジをタッチ操作することで、クライアントアプリケーションからコントローラを設定できる。設定可能な項目として、論理的な結線の追加や削除、VLAN 設定、プロトコル指定による優先制御、および帯域幅制限の 4 種類がある。クライアントアプリケーションは、利用者がこれらの設定を行う操作をしたとき、コントローラへ設定変更を要求する HTTP リクエストを送信する。その後、コントローラより受信する HTTP レスポンスから、設定変更後の OpenFlow ネットワークの情報を取得する。この情報を基に、クライアントアプリケーションは、タッチディスプレイ上の論理トポロジを更新する。

本システムを用いることで、タッチディスプレイ上に可視化された論理トポロジや、VLAN 情報などの設定情報をタッチにより操作でき、直感的にコントローラを設定できる。さらに、設定変更に伴う論理トポロジの変化をタッチディスプレイ上で容易に把握できる。

2.2 コントローラ

コントローラは、クライアントアプリケーションから受信する HTTP リクエストの処理と、OpenFlow ネットワークの制御を行う。コントローラが受信する HTTP リクエストは、OpenFlow ネットワークの情報を要求するメッセージと OpenFlow ネットワークの設定変更を要求するメッセージの 2 種類がある。

コントローラは、クライアントアプリケーションから OpenFlow ネットワークの情報を要求する HTTP リクエストを受信したとき、自身が収集・保持している OpenFlow ネットワークの情報を HTTP レスポンスとしてクライアントアプリケーションへ送信す

る。また、コントローラの設定変更を要求する HTTP リクエストを受信したとき、その要求を基に、OpenFlow スイッチに対してフローエントリの追加、削除または書き換えの操作を行う。本システムでは、設定の種類ごとにフローエントリを作成しており、それらの各フローエントリはフローテーブルごとに管理している。そのため、コントローラの設定変更は、設定の種類に対応したフローテーブルにのみ影響を及ぼす。OpenFlow スイッチは、受信したパケットに対してフローテーブルの昇順に処理を行うことで、2.1 節で述べた 4 種類すべての設定に基づく経路制御を実現している。設定変更が完了すると、コントローラは設定変更後の OpenFlow ネットワークの情報を HTTP レスポンスとしてクライアントアプリケーションへ送信する。

3. 実験

本システムを評価するために、動作検証および性能評価実験を行った。

3.1 動作検証

動作検証では、物理トポロジの種類によらず本システムの全ての機能が正常に動作することを確認した。検証には、Linux Network Namespace を利用して構築した仮想 OpenFlow ネットワークを用いた。検証は、1つの PC 上で行い、ホスト OS にてクライアントアプリケーションを、ゲスト OS にて仮想 OpenFlow ネットワークおよびコントローラを動作させた。図 2 に示す 6 つのトポロジを Linux 上に仮想 OpenFlow ネットワークとして構築し、それぞれで同様の検証を行った。まず、構築した OpenFlow ネットワークと本システムのコントローラを接続する。次に、接続した OpenFlow ネットワークの論理トポロジがタッチディスプレイ上に表示されることを確認した。そして、クライアントアプリケーションを用いてコントローラを設定し、コントローラにより設定が OpenFlow ネットワークに正常に反映されることを確認した。設定した項目およびその確認方法を表 1 に示す。動作検証の結果、本システムの全ての機能が正常に動作することを確認した。

3.2 性能評価実験

性能評価実験では、コントローラが制御する OpenFlow スイッチの台数の増加がシステムの応答時間に及ぼす影響を検証した。応答時間は、クライ

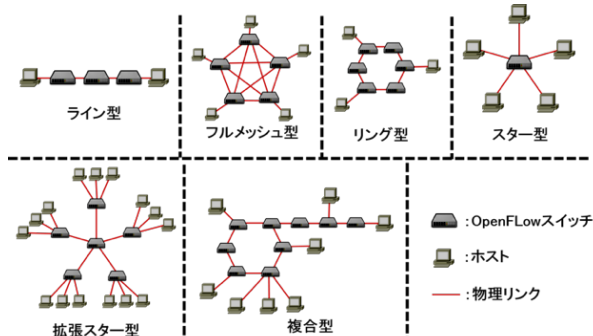


図 2：実験に用いたトポロジ

表 1：設定する項目およびその確認方法

| 設定 | 操作 | 確認事項 |
|---------------------|---|--|
| 論理情報 (ルート) の追加および削除 | 3個のホストを用意し、各ホスト間で通信が可能もしくは不可能になるように、ルートの追加や削除を複数回行う | 設定の度にそれぞれのホスト間のうち、論理情報 (ルート) が存在するホスト間で ping が到達すること 設定の度にそれぞれのホスト間のうち、論理情報 (ルート) が存在しないホスト間で ping が到達不可であること |
| プロトコル指定による優先制御 | 2個のプロトコルに優先度の差をつけ、両方のプロトコルのパケットを同時に流す | 優先度が高いプロトコルのパケットが優先して処理・転送されること |
| 帯域幅制限 | 経路に帯域幅の制限値を設定する | 設定を施した経路上に制限値を超えるパケットを流した際、パケット流量が制限値を超えないこと |
| VLAN | 互いに経路情報 (ルート) が存在する3個のホストのうち、1個のホストのみが異なる VLAN に属し、残りの2個のホストは同一 VLAN に属するように VLAN 設定を施す | 同一 VLAN 上のホスト同士で ping が到達すること 異なる VLAN 上のホスト同士で ping が到達不可であること |

アントアプリケーションが OpenFlow ネットワークの情報を要求する HTTP リクエストを送信してから、タッチディスプレイに論理トポロジが表示されるまでの時間と、OpenFlow ネットワークの設定変更を要求する HTTP リクエストを送信してから、タッチディスプレイに表示された論理トポロジが更新されるまでの時間とする。この 2 種類の応答時間について、OpenFlow スイッチの台数を増加させながら繰り返し計測した。その結果、それぞれの応答時間について、OpenFlow スイッチの台数が 1 台のときと 50 台のときの差はいずれも 0.05 秒以下となり、OpenFlow スイッチの台数を 50 台程度増加させてもほとんど遅延が発生しないという結果であった。このことから、OpenFlow スイッチの台数の増加が本システムの応答時間に及ぼす影響はわずかであることがわかった。

4. 結論

本研究では、コントローラの設定情報と論理トポロジの視覚的な関連付けを可能とする OpenFlow ネットワーク運用管理支援システムを開発した。本システムを用いることで、タッチディスプレイ上でコントローラの設定情報と論理トポロジの関係を視覚的に把握できる。さらに、タッチ操作によりコントローラを直感的に設定でき、設定変更に伴う論理トポロジの変化をタッチディスプレイ上で容易に把握できる。

参考文献

- 1) N. McKeown, T. Anderson, H. Balakrishnan, et al.: OpenFlow: Enabling Innovation in Campus Networks, ACM SIGCOMM Computer Communications Review, Vol.38, Issue2, pp.69-74, 2008.
- 2) B. A. A. Nunes, M. Mendonca, X. Nguyen, K. et al.: A Survey of Software-Defined Networking: Past, Present, and Future of Programmable Networks, IEEE Communications Surveys & Tutorials, vol.16, no.3, pp.1617-1634, 2014.
- 3) D. Kreutz, F. M. V. Ramos, P. E. Verissimo, et al.: Software-Defined Networking: A Comprehensive Survey, Proceedings of the IEEE, vol.103, no.1, pp.14-76, 2015.
- 4) A. Blenk, A. Basta, M. Reisslein, and W. Kellerer: Survey on Network Virtualization Hypervisors for Software Defined Networking, IEEE Communications Surveys & Tutorials, vol.18, no.1, pp.655-685, 2016.