

SDN を用いたモビリティマネジメントに関する一考察

畑 美純^{†1} 和泉 諭^{†2} 阿部 亨^{†1,†2} 菅沼 拓夫^{†1,†2}

^{†1} 東北大学大学院情報科学研究科 ^{†2} 東北大学サイバーサイエンスセンター

1 はじめに

モバイル端末の高機能化や無線通信の高速化、クラウドサービスの普及に伴い、IP サービスの継続性を確保するモビリティマネジメントの重要性が高まっている。モビリティマネジメントの実現には、広範囲でのモビリティを確保するためにモバイル端末がドメインを移動した際の処理や、通信の遅延を軽減するために通信経路の最適化が必要である。現在、IP レベルでのモビリティマネジメントの標準として Mobile IP が普及しているが、ハンドオーバー時の経路の最適化が困難である等の課題がある。

その解決にあたって、本研究では Software Defined Network (SDN) を適用したアプローチに着目し、このアプローチに基づき、ドメインを跨ぐより広域でのハンドオーバーにおける経路最適化の実現を図る SDN 型モビリティマネジメント手法を提案する。本稿では、本手法の概念を述べ、基本動作の設計を行う。

2 関連研究と課題

モビリティマネジメントに関する技術として、Mobile IPv4[1] が標準化されている。これはモバイル端末が移動しても移動前と同じ IP アドレスの使用を可能にする技術である。IPv6 に対応した技術としては Mobile IPv6[2] がある。Mobile IPv6 には経路最適化機能が備わっているが、本機能に対応している IPv6 端末のみが使用できることから、全ての端末の通信経路を最適化することは困難である。また、Proxy Mobile IPv6[3] は全ての通信を LMA を介して行う必要があり、通信経路の最適化が困難である。

Mobile IP における経路最適化の課題を解決するため、SDN をモビリティマネジメントに適用するアプローチがある。SDN はドメイン内のフローを集中管理するため、ドメイン内ハンドオーバーの際に経路の最適化を実現できる。また、SDN はネットワーク機器が対応していれば適用でき、端末の機能には依存しないため、Mobile IPv6 より経路最適化機能の導入が容易である。

これら SDN を適用したアプローチでは、OpenFlow を用いているが、管理外のドメインとの通信制御を OpenFlow はサポートしていない。そのため、管理外のドメインとの通信の制御を必要とする。

る、ドメインを跨いだハンドオーバーの際に経路の最適化を実現することが困難である。文献 [4] はドメイン間のハンドオーバーに Mobile IP を適用する手法を提案しているが、モバイル端末に Mobile IP の経路最適化機能が備わっていない場合、経路を最適化することができない。

文献 [5] はドメイン間のハンドオーバーの際には全コントローラにモバイル端末の移動を通知している。しかしドメイン間ハンドオーバーが頻繁に起こった場合やドメインが増えた場合にはコントローラ間の通信量が大きくなる。文献 [6] ではモバイル端末に ID を付与し、ハンドオーバーの際には ID と IP アドレスの紐付け情報を更新するが、ドメイン間ハンドオーバーの際における経路最適化は実現されていない。

3 SDN を用いたモビリティマネジメント手法の提案

3.1 概要

前節で述べた課題を解決するため、本研究ではドメイン間ハンドオーバーの際に経路を最適化する、SDN を用いたモビリティマネジメント手法を提案する。本手法は IPv6 ネットワークを対象としており、モバイル端末は IPv6 対応であること以外に特別な機能は必要ない。ドメイン間の経路を最適化するためには、通信に関係するドメインが協調し、モバイル端末 (MN) や MN の通信相手 (CN) の現在位置を把握している必要がある。

そこで、本手法では MN がドメインを移動した際にその端末の情報を、1. 移動前に MN が属していたドメイン、2. MN が移動後属しているドメイン、3. CN が属するドメインの三つのドメインを中心とした最小限のドメイン間で共有し、MN-CN 間の経路を検索することにより、コントローラ間の通信量を抑えつつ経路の最適化を実現する。

3.2 設計

本手法において、MN が移動した際の基本動作の設計を行う。本手法のネットワーク構成を図 1 に示す。各ドメイン内には複数の SDN スイッチとコントローラが接続されており、コントローラがドメイン内のネットワークの状況や SDN スイッチの設定を管理する。MN が移動した際にドメイン間で端末紐付け情報を交換するが、端末紐付け情報は移動した MN の移動前後の位置を紐付ける情報で、以下の情報から構成される。

- MAC アドレス

A Study on SDN Based Mobility Management
 Misumi HATA^{†1}, Satoru IZUMI^{†2}, Toru ABE^{†1,†2}, and Takuo SUGANUMA^{†1,†2}
^{†1} Graduate School of Information Sciences, Tohoku University
^{†2} Cyberscience Center, Tohoku University

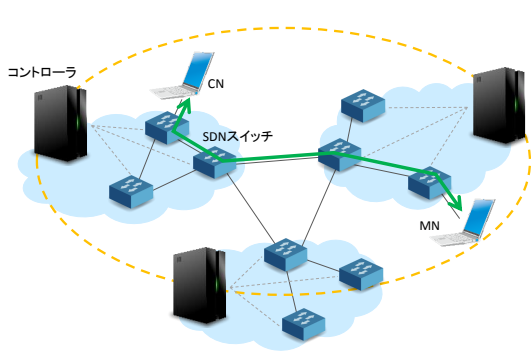


図1 ハンドオーバー前の構成

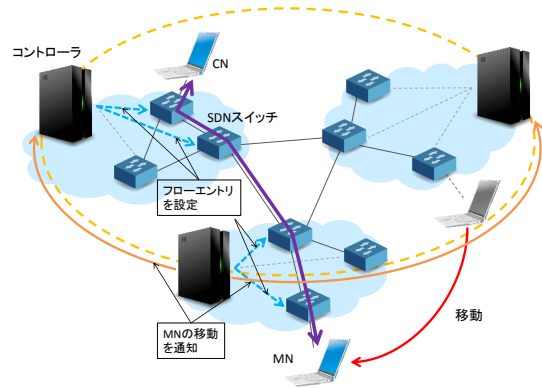


図2 ハンドオーバーの処理およびその後の構成

- ネットワークプリフィクス
- 移動前の IP アドレス
- 移動後の IP アドレス

ドメイン間ハンドオーバーの処理は以下の流れで行われる。ハンドオーバー後のネットワークの構成は図2の通りである。MN がドメインを移動した際の経路最適化手順を以下に示す。

1. 移動先ドメインのコントローラが、MN の移動を検知し、移動前に所属していたドメイン (移動元ドメイン) を検索。
2. 移動元ドメインのコントローラが応答し、移動元ドメインで MN に割り当てられていた IP アドレスを移動先ドメインのコントローラに通知。
3. 移動先ドメインのコントローラが MN の端末紐付け情報を生成。
4. 移動先ドメインのコントローラが移動前の MN と通信している CN を検索。
5. 移動先ドメインのコントローラが、移動元ドメインおよび CN が属するドメインのコントローラに MN の端末紐付け情報を通知。
6. 移動先ドメインのコントローラおよび CN が属するドメインのコントローラが MN と CN の通信の最適な経路を算出。
7. CN が属するドメインのコントローラが、端末紐付け情報に基づきドメイン内のスイッチに、MN に宛てたパケットの宛先 IP アドレスを移動後の IP アドレスに書き換え転送するフローエントリを設定。
8. 移動先ドメインのコントローラが、ドメイン内のスイッチに、MN から CN に宛てたパケットを転送するフローエントリを設定。

以上の手順により MN は移動後も最適化された経路で CN と通信を継続することができる。さらに、SDN スイッチで IP アドレスを書き換える処理を行うため、他のドメインへ移動した場合でも元の IP アドレスで通信を継続することができる。

また、コントローラ間の相互通信を可能とするためにはオーバーレイ P2P ネットワークを構築し、その上にコントローラを配置することや移動後の端末の同定には MAC アドレスを用いることなどを検討している。

4 おわりに

本稿では SDN を用いたモビリティマネジメント手法の提案および基本動作の設計を行った。今後は、設計を基に実装を行い、仮想ネットワーク上のシミュレーションで提案手法の評価を進める。さらに、ドメインの数が増加した際に重要となる MN の移動先探索アルゴリズム、ドメイン内におけるフローエントリのダウンロード戦略や MN と CN 両方がドメインを移動した場合の処理について検討していく。

参考文献

- [1] Perkins, C.: IP Mobility Support for IPv4 (2010). <https://tools.ietf.org/html/rfc5944>.
- [2] Johnson, D. et al.: Mobility Support in IPv6 (2004). <https://tools.ietf.org/html/rfc3775>.
- [3] Gundavelli, S. et al.: Proxy Mobile IPv6 (2008). <https://tools.ietf.org/html/rfc5213>.
- [4] Wang, Y. et al.: A Solution for IP Mobility Support in Software Defined Networks, *23rd International Conference on Computer Communication and Networks (ICCCN)*, pp. 1 – 8 (2014).
- [5] Wang, Y. et al.: Design and Implementation of a Software-Defined Mobility Architecture for IP Networks, *Mobile Networks and Applications*, Vol. 20, No. 1, pp. 40–52 (2015).
- [6] Li, Y. et al.: Software Defined Networking for Distributed Mobility Management, *IEEE Globecom Workshops*, pp. 885–889 (2013).