

OpenFlow を用いたデータ収集トラフィックの動的制御に関する研究 A Study on Dynamic Control of Data Acquisition Traffic by Using OpenFlow

西村 俊彦[†] 長坂 康史[†] 山木戸 啓亮[†] 堀 裕貴[†]
Toshihiko Nishimura Yasushi Nagasaka Keisike Yamakido Hiroki Hori

1. はじめに

情報通信技術の発展に伴い、ネットワーク上には多種多様な通信トラフィックが発生し、近年、その量は急激に増加している。また、通信トラフィックの量は時間と共に変動しており、トラフィック量の変動を考慮した通信経路の制御が重要視されている。

一方で、大規模高エネルギー物理学実験におけるデータ収集 (DAQ) システムでは、複数の測定器から発生するデータを、千台規模のコンピュータで取得し、それらのネットワークスイッチを用いて一台の解析用コンピュータに集約する^[1]。このシステムでは、膨大なデータの効率的な収集のために並列処理を行う。そのため、並行する複数の通信処理を通信経路内で競合させることなく処理する必要があり、通信経路の選択がシステム全体の効率に影響する^[2]。このため、高効率なデータ収集システムには、通信経路上のトラフィック変動に応じて動的に通信制御を行う機構が要求される。

そこで本研究では、DAQ システムの通信効率の向上のため、ネットワークの動的集中制御を可能とする OpenFlow 技術^[3]を用いてトラフィック変動を考慮した高効率データ収集システムの通信経路の動的制御手法を提案する。

2. OpenFlow

OpenFlow はソフトウェアによるネットワークの動的制御をコンセプトとする SDN (Software Defined Networking) を実現する要素技術の 1 つである。

OpenFlow を用いたネットワークは、一般的に経路制御を行う単一の OpenFlow Controller (OFC) と、パケット転送を行う複数の OpenFlow Switch (OFS) によって構成される。OFC はネットワーク内の OFS を集中管理する制御端末であり、OFS のパケット処理条件 (フローエントリ) を OFS の持つ経路表 (フローテーブル) に対して定義する。OFS は、OFC によって定義されたフローエントリに基づいてパケットの転送や破棄、宛先の書き換えなどを実行する。また、OFS はフロー毎、ポート毎の送受信パケット数やパケットサイズ等を含む統計情報を保持しており、OFC はこれらの情報を任意のタイミングで取得することができる。

3. 提案手法

従来の DAQ システムでは、複数の通信トラフィックが特定の通信経路に偏ることによって、大幅な通信遅延が生じる問題があった。そこで本研究では、トラフィック状況に応じた動的な経路制御を行うため、OFC が通信経路内の OFS に対して、定期的にトラフィックの偏りを最

小化した経路を導出・適用する手法を提案する。提案手法の概要を図 1 に示す。本提案手法は、トラフィックの偏りを検知するトラフィック管理機能と、通信経路の導出を行う経路計算機能を OFC に実装することで実現する。

トラフィック管理機能では、ネットワーク上の各 OFS に対して定周期で統計情報の要求を行い、OFS の受信パケットサイズを取得する (図 1 ①, ②, ③)。そして、取得した受信パケットサイズを閾値と比較する (図 1 ④)。このとき、取得した受信パケットサイズが閾値以上であれば、該当する OFS の経路コストを加算する (図 1 ⑤)。この経路コストは、通信経路の混雑状況を定量化しており、OFC は各 OFS の情報と対応する経路コストの組を管理する。

一方、経路計算機能ではトラフィック管理機能によって加算される各 OFS の経路コストと対応する OFS の情報を一定の周期で取得する (図 1 ⑥, ⑦)。その後、それらの情報を基に、トラフィック量が最小である経路をダイクストラ法 (Dijkstra's algorithm)^[4]によって導出する (図 1 ⑧)。そして、導出した通信経路に該当するフローエントリを OFS に対して定義する (図 1 ⑨)。

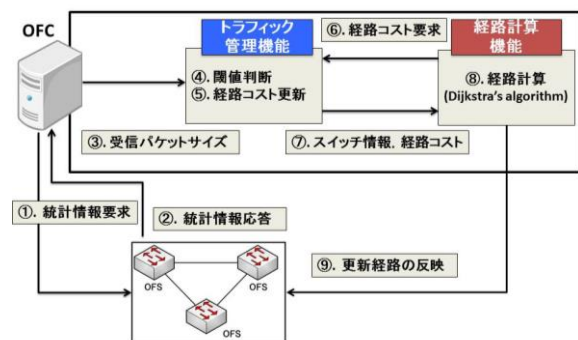


図 1 提案手法概要

4. 性能測定

4.1 測定環境

提案手法の有用性を検証するため、提案手法を用いた動的経路制御を行う場合と、行わない場合での通信速度を測定した。測定環境を図 2 に示す。

この図に示すとおり、スイッチは 1 台の実機 OFS P-3290 と、4 台の仮想スイッチを OpenvSwitch^[5]を用いて構成した仮想環境上に用意し、それぞれを接続している。実機 OFS と仮想環境は RJ-45 ケーブルによって接続され、通信帯域は OFS3 と hostD 間が 10 Gbps で、その他は 1 Gbps である。

測定で使用した PC の仕様を表 1 に示す。表 1 の環境における OpenFlow のバージョンは 1.3 で、OFC のプログラミングフレームワークとして Trema-edge^[6]を使用する。

[†] 広島工業大学 Hiroshima Institute of Technology

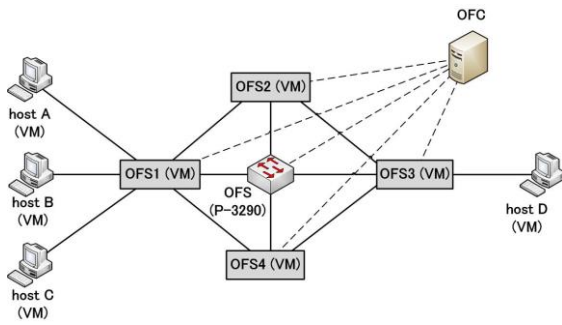


図2 測定環境

表1 測定 PC 仕様

OS	Ubuntu 12.04.1 LTS (32 bit)
CPU	Intel® Core TMi5 CPU650@ 3.20GHz × 4
Memory	4.0 GB
NIC	Intel Gigabit CT Desktop Adapter
OpenvSwitch	Ver 2.0.0

4.2 測定方法

性能測定は、iperf 3^[7]を用いて、hostA から hostD への TCP 通信の通信速度を測定した。また、通信遅延を起こす要因を作るために、hostB から hostD, hostC から hostD への TCP 通信も同時に行う。性能測定のシナリオは以下のとおりである。

測定開始 0 sec から 60 sec までは、hostA から hostD への通信のみが行われる。続いて、60 sec 以降は、hostB から hostD へ通信を開始する。そして、120 sec 以降、hostC から hostD への通信を開始する。このため、図 2 のネットワーク上には、最終的に 3つの通信が並行して行われる。このとき、提案手法の動的経路変更の判断周期は 60 sec とし、測定回数は 10 回とする。

また、比較のために、既存方式を用いた測定も行った。具体的には、図 2 のネットワークから OFC を取り除いた環境における測定を行った。このとき、ネットワーク上の各 OFS は、単一経路のみで通信を行う従来の L2 スイッチと同様の動作を行うため、提案手法における通信トラフィックを考慮した経路制御は行わない。また、既存方式の経路設定は、host (A, B, C) → OFS1 → OFS (p-3290) → OFS3 → host D とする。

4.3 測定結果と考察

性能測定の結果を図 3 に、スループットの平均値と標準偏差を表 2 に示す。

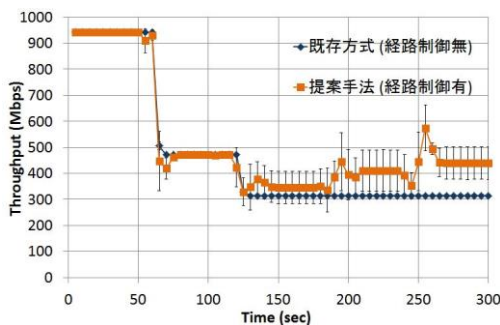


図3 測定結果

表2 スループットに関する比較

手法	平均値 (Mbps)	標準偏差
既存方式 (経路制御無)	471.640	1.958
提案手法 (経路制御有)	517.433	35.475

図 3 より、既存方式と提案手法では、共に 60 sec と 120 sec 時に大幅なスループットの低下が確認された。この理由は、通信遅延要因である hostB と hostC の通信によるものと考えられる。

また、既存方式では、120 sec 以降のスループットが 314 Mbps に収束していることが分かる。これは、単一経路に通信トラフィックが集中し、有効帯域が約 1/3 に減少した状況であると考えられる。

これに対して、提案手法では、120 sec 以降からスループットの増加が見られる。この理由として、通信トラフィックの変動を考慮した経路制御が行われ、通信帯域に余裕のある経路である OFS2 と OFS4 が選択されたためと考える。また、動的経路変更の周期である 60 sec に合わせて、一時的なスループットの低下が見られる。この要因は、OFC による経路切替処理の際に発生する、各 OFS に対するフローエントリの更新処理によるオーバーヘッドによるものと考えられる。

また表 2 より、提案手法の適用によって、スループットの平均値が既存方式と比較して約 9 % 向上していることが分かる。しかし、両方式の標準偏差を比較すると、既存方式に対して提案手法のスループットが不安定である。

このため、通信速度の向上に加えて、安定性を維持するための手法の検討が必要とされる。

5. まとめ

本研究では、DAQ システムを構成するネットワークに対する、OpenFlow 技術を用いた通信効率の向上を目的としている。そのため、トラフィック量の変動に応じて通信経路を動的に制御する手法の提案、性能評価を行った。

性能評価の結果、提案手法の適用による、約 9 % の平均スループットの向上が確認された。しかし、既存方式と比較してスループットのばらつきが大きく、安定性が低いことが分かった。

今後の課題として、通信速度の向上と安定性を維持することのできる手法の検討が挙げられる。また、より大規模なネットワーク環境における有用性の検証も必要である。

参考文献

- [1] 山木戸啓亮, 長坂康史, 「大規模データ収集システムにおける DCB を用いたネットワーク QoS の向上に関する研究」, FIT2014 第 13 回情報科学技術フォーラム 講演論文集 第 4 分冊, pp.119-120 (2014).
- [2] 沖恭志, 長坂康史, 「大規模データ収集システムにおける通信の効率化に関する研究」, 広島工業大学紀要 研究編, Vol.47, pp.127-131 (2013).
- [3] The OpenFlow Switch Consortium, <http://www.openflow.org/>, (2015/01/15 参照)
- [4] 安井雄一郎, 「大規模最短路問題におけるダイクストラ法の高速度化」, 中央大学大学院研究年報 理工学研究科篇, 第 39 号 (2009).
- [5] OpenvSwitch, <http://openvswitch.org/>, (2015/01/15 参照)
- [6] trema-edge, <https://github.com/trema/trema-edge>, (2015/01/16 参照)
- [7] iperf3, <https://code.google.com/p/iperf/>, (2015/01/16 参照)