

アプリケーション識別型トラフィック制御システムの開発 Development of a System for Application-Aware Traffic Control

式町 龍聖¹⁾ 神屋 郁子²⁾ 下川 俊彦¹⁾
Ryusei Shikimachi Yuko Kamiya Toshihiko Shimokawa

1 研究背景

近年、アプリケーションの多様化に伴って、アプリケーションのネットワーク要件が多様化している。例えば、クラウドゲーミングサービスでは、クライアントとサーバ間の許容遅延や必要な帯域幅が詳しく明記されている。また、動画配信サービスでは、クライアントで再生される動画の解像度によって必要な帯域幅が異なっている。このように、アプリケーションごとに異なるネットワーク要件が存在するにもかかわらず、現在はそれらを考慮したトラフィック制御は十分に行われていない。

その背景には、ネットワークにおけるルーティングが、パケットの宛先 IP アドレスのみに基づいて経路を決定・転送しているという仕組みがある。このため、パケットがどのアプリケーションに由来するかは考慮されず、すべて一律に扱われてしまう。その結果、各アプリケーションのネットワーク要件を満たせず、サービス品質が低下する可能性がある。

2 本研究の目的と解決策

本研究の目的は、アプリケーションごとのネットワーク要件を考慮してトラフィック制御することで、アプリケーションが提供するサービス品質の低下を防ぐことである。

この目標達成のための解決策を述べる。トラフィック制御には、セグメントルーティング技術の1つである SRv6[1]を用いる。SRv6では、送信元がパケットに経路情報を付与することで、パケット転送する経路を柔軟に制御できる。付与される経路情報は、Segment Routing Header(SRH)という IPv6 拡張ヘッダである。ネットワークの中継地点を表す Segment Identifier(SID) のリストなど、セグメントルーティングに必要な情報が格納されている。アプリケーションごとに SRH を付与することで、アプリケーションのネットワーク要件を考慮したトラフィック制御が可能となる。

このようなトラフィック制御を実現するためには、利用されているアプリケーションを識別する必要がある。近年では、機械学習を用いてリアルタイムにアプリケーション識別する手法 [2] も研究されている。本研究では、クライアントのアクセス先ホスト名を用いることでアプリケーションを識別する手法を提案する。

3 関連研究

アプリケーションごとの通信特性に応じた経路制御を実現する研究として、杉浦らによる Acar システムが挙げられる [3]。この研究では、SRv6 を用いた分散型の動的経路制御方式を提案しており、ネットワーク内のスイッチに配置されたエージェントが各リンクの帯域使用状況を収集し、アプリケーションごとの帯域幅要求に応じた最適な経路を選定する仕組みを実装している。

Acar は、SRv6 の特性を活かして、経路の変更をネッ

トワークの入口スイッチのみで完結させることで、OpenFlow に代表される中央集権型の SDN 方式に見られるスケーラビリティの課題やコントローラ依存の問題を克服している。評価実験では、ECMP と比較して最大 23% のスループット向上を達成しており、その有効性が示されている。

一方で、Acar ではアプリケーションの識別機構は搭載されておらず、通信フローに対してはあらかじめユーザが指定した帯域幅情報をもとに経路が制御されている。そのため、通信フローの内容に基づいて自動的にアプリケーションを識別し、経路制御に反映するような仕組みは含まれていない。

そこで本研究では、アプリケーション識別機構と連携し、アプリケーションの要件に応じた経路制御を SRv6 の仕組みを活用して実現するトラフィック制御システムを開発する。

4 システムの設計方針

提案するアプリケーション識別型トラフィック制御システム(以下、本システム)をどのようなネットワーク環境に適用し、どのような設計方針に基づいて構成するのかについて述べる。

4.1 適用環境

本システムは、ISP のネットワークにおける導入を想定している。ISP ネットワークでは、多様なアプリケーショントラフィックが混在しており、ネットワーク要件に応じたトラフィック制御の必要性が高まっている。本システムを適用することで、アプリケーションごとの要件に応じたトラフィック制御が可能となり、ネットワーク全体の効率的な運用や、各アプリケーションに対するサービス品質の向上が期待される。

4.2 システムの前提条件

本システムは、以下のような前提条件に基づいて設計している。

- クライアントは、システムで利用するフルリゾルバを用いて名前解決し、本システムがそのリクエストと応答結果の情報を取得できる。
- 各ホスト名は一意に特定のアプリケーションと対応しており、1つのホスト名に複数のアプリケーションが混在していない。
- アプリケーションのネットワーク要件(必要帯域、許容遅延など)が、事前に把握または定義されている。

2項目の前提条件について補足する。一般に、1つのホストで複数のアプリケーションが提供されることもある。近年はアプリケーションごとにホスト名(サブドメイン)を分ける構成が見られる。例えば、mail.google.com や calendar.google.com のようにサービス単位で専用のホスト名が割り当てられる。この傾向を踏まえ、本システムでは1ホスト名が1アプリケーションに対応すると仮

1) 九州産業大学 Kyushu Sangyo University

2) 福岡女子大学 Fukuoka Women's University

定し、ホスト名を識別子として用いる。

5 システム構成

本システムは、“フルリゾルバ”、“アプリケーション識別サブシステム”、“経路制御サブシステム”、“モニタリングサブシステム”の4つの主要コンポーネントから構成される。システムの構成を図1に示す。

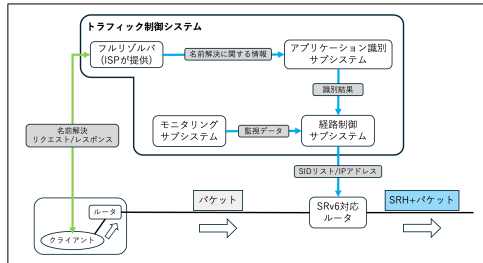


図1 システムの構成図

5.1 フルリゾルバ

クライアントが利用するISPのフルリゾルバを用いる。本システムはこのフルリゾルバから、クライアントの名前解決リクエストおよびその応答結果を取得し、ホスト名に対応するIPアドレスを把握する。

5.2 アプリケーション識別サブシステム

アプリケーション識別サブシステムは、フルリゾルバのキャッシュ情報を用いてアプリケーションの識別する。以下の3つのモジュールから構成される。アプリケーション識別サブシステムの構成を図2に示す。

- ホスト名抽出モジュール
- アプリケーションプロファイルデータベース
- アプリケーション識別モジュール

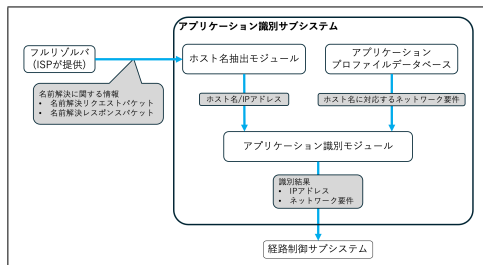


図2 アプリケーション識別サブシステムの構成図

ホスト名抽出モジュールは、専用フルリゾルバから名前解決に関する情報を取得し、そこからホスト名とそれに対応するIPアドレスを抽出する。アプリケーションプロファイルデータベースは、ホスト名とアプリケーション名、ネットワーク要件の対応関係を保持するデータベースである。アプリケーション識別モジュールは、抽出されたホスト名をもとにアプリケーションを識別する。そして、ホスト名に対応するIPアドレスとネットワーク要件を経路制御サブシステムに提供する。

5.3 経路制御サブシステム

経路制御サブシステムは、アプリケーションの識別結果とネットワークの監視データをもとに、アプリケーションにとって最適な経路を選定・管理する機能を担う。本サブシステムは以下の2つのモジュールで構成される。経路制御サブシステムの構成を図3に示す。

- SIDリスト管理モジュール
- SIDリストデータベース

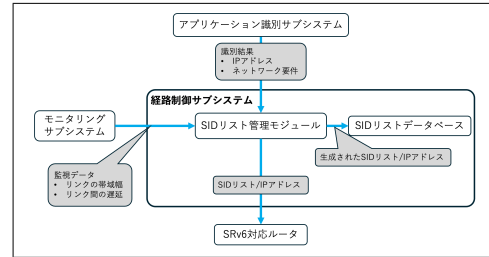


図3 経路制御サブシステムの構成図

SIDリスト管理モジュールは、アプリケーション識別サブシステムから、ホスト名に対応するIPアドレスとネットワーク要件を取得する。これにモニタリングサブシステムから得た監視データを加味することで、アプリケーションにとって最適なSIDのリストを生成する。

生成されたSIDリストは、対応するIPアドレスとともにSIDリストデータベースへ記録される。SIDリスト管理モジュールは、データベースに保持されている過去の情報と比較し、SIDリストに変更があった場合、SIDリストとIPアドレスを通知する。

5.4 モニタリングサブシステム

モニタリングサブシステムは、トラフィック制御対象のネットワークで利用されている機器から、得られる監視データ（帯域使用率、遅延など）を定期的に収集し、内部のデータベースに蓄積する。また、経路制御サブシステムからのリクエストに応じて蓄積した監視データを提供する。

6 まとめ

本研究では、アプリケーションごとに異なるネットワーク要件が存在するにもかかわらず、それらを考慮したトラフィック制御が十分に行われていないという課題に着目した。

これらの課題を解決するために、クライアントが利用するフルリゾルバのキャッシュ情報を活用し、アクセス先ホスト名を用いてアプリケーションを識別する手法を提案した。さらに、本システムではSRv6の仕組みを利用し、識別されたアプリケーションのネットワーク要件とネットワークの状態に基づいて、最適な経路（SIDリスト）を選定しパケットに付与することで、アプリケーションごとの要件に応じた柔軟なトラフィック制御を実現する。

今後の展望としては、提案システムの実装と評価を行い、実ネットワーク上での適用性や、アプリケーションごとの通信品質の改善効果を検証する。

参考文献

- [1] C. Filsfils, et al. Segment Routing over IPv6 (SRv6) Network Programming. RFC 8986, February 2021.
- [2] 岩井貴充, 中尾彰宏. 機械学習によるモバイルアプリケーション判定の検証方法と問題点. 電子情報通信学会技術研究報告; 信学技報, Vol. 119, No. 92, pp. 29–34, 2019.
- [3] 杉浦智基. Acar: Srv6を用いたアプリケーションの通信特性を考慮した通信経路制御システム. 2022.