

6 ネットワークの効率的な資源分配を目指す研究開発

6 *Research and Development of Technology for Effective Resource Allocation in Networks*

6-1 仮想ネットワークの自動構築制御技術

6-1 *Automatic Construction and Control of Virtual Networks*

宮澤高也 Ved P. Kafle

Takaya MIYAZAWA and Ved P. KAFLE

将来のサービス要求の多様化及び通信量の時間変動に迅速に対応するためには、仮想ネットワークの構築・制御を、なるべく人手を介さず、自動化させる仕組みが必要である。本稿では、特に、通信量及び資源利用状況の変動に対応もしくは予測し、各サービス機能を提供する計算機資源を動的に調整してサービス品質の維持を実現するための、我々の研究開発について紹介する。

To deal with diversification of service requirements and time variation of traffic in future networks, we inevitably require technologies to automatically and agilely construct and control virtual networks with reducing manual procedures as much as possible. In this paper, we introduce our research and development on dynamic adjustments of computational resources to provide sufficient quality of services in response to analyses of network traffic and resource usage state.

1 まえがき

今後の5G・IoT時代では、多種多様な機器がネットワークにつながり、サービス要求がますます多様化していくと同時に、ネットワーク上を流れる通信量が増加し、かつモバイル通信の普及も相まって時々刻々と変動することが予想される。情報を流通するためのネットワーク資源及び計算機資源は有限であり、様々な通信サービスに対して適時適切に資源を配分し、サービス品質を維持する必要がある。しかしながら、既存の人手によるネットワークの構築・制御では、ユーザごとに要求が異なるサービスを迅速に提供したり、障害発生や通信量変動等に即応したりすることは極めて困難である。そこで、サービス要求の多様化に対応するためのネットワーク仮想化技術やネットワーク機能仮想化技術が注目されており、さらには、状況の変化に即応するために仮想ネットワーク構築制御を自動化させる仕組みが必要である。現在、国内外において、仮想ネットワークや仮想ネットワーク機能の運用管理・制御の自動化に関する研究・議論が活発に行われている。例えば、欧州の標準化機構であるETSI (European Telecommunications Standards Institute) は、2017年12月に、ネットワーク及びサービスの管理の自動化に関するISG (Industry Specification Group)

であるZSM (Zero touch network and Service Management) を立ち上げた [1]。

我々は、仮想ネットワーク構築制御において、AI等も活用し、通信量及び資源利用状況の変動に応じて各サービス機能を提供する計算機資源量の調整を自動的に行うための技術の研究開発を行っている。本稿では、我々が研究開発を行っている以下の3種類の自動的な計算機資源調整技術について紹介する。

- サービスを提供する各仮想マシン内の計算機資源の動的増減 (Vertical Scaling)
- 各サービスネットワーク内の仮想マシン数の動的増減 (Horizontal Scaling)
- 上記2手法を活用した、サービスネットワーク間の動的な計算機資源調停・機能移行 (Internetwork Scaling)

2 多様なサービス要求に対応するための要素技術

2.1 ネットワーク及びネットワーク機能の仮想化

近年、ネットワークやサーバ及びそれらのネットワーク機能 (NF) の仮想化技術の研究が盛んに行われている [2]-[5]。仮想ネットワークオペレータ (VNO: Virtual Network Operator) からの資源要求に応じて、

物理インフラ提供者 (InP: Infrastructure Provider) が各 VN に対して異なる物理資源、仮想 NF (vNF)、もしくはフロースペースを割り当てることで、VN 同士を分離できる。通信ビジネス市場においては、例えば、モバイル VNO (MVNO) が既に、他のキャリアから物理回線及びサーバを借りて VN サービスを提供している。同様に、NF の仮想化 (NFV: Network Function Virtualization) によって、複数の vNFs を共通の汎用サーバで構成された基盤上でソフトウェアで実装可能となり、CAPEX/OPEX を低減できる。vNF としては、例えば、ファイウォール、アドレス変換 (NAT: Network Address Translation)、パケット検査 (DPI: Deep Packet Inspection)、メディア変換、映像解析、コンテンツキャッシュ、ネットワークコーディング、認証機能等があげられる。ネットワーク仮想化及び NFV は、限られた物理ネットワーク資源や計算機資源を有効活用し、ユーザごとに要求が異なるサービスを提供するための有望な技術である。

2.2 サービス機能チェイニング

複数の vNFs によるトラフィックのネットワーク内処理を実現する技術として、サービス機能チェイニング (SFC: Service Function Chaining) が注目されている [6]-[11]。SFC とは、各 VN において、必要な複数の vNFs を分散配置し、特定の経路に沿って転送されるパケットに対して適切な順番でネットワーク機能処理を施すための仮想的なサービス機能チェーンを構築する技術である。仮想ネットワーク上で、柔軟な vNFs の動的配置や、ユーザ要求に応じて動的に特定の vNF の追加または削除が可能である。SFC アーキテクチャの基本設計について、IETF が RFC 7665 及び RFC 8300 を発行しており [6][7]、さらに、ソフトウェア定義ネットワーク (SDN: Software Defined Network) 基盤及び NFV 基盤に基づいた SFC アーキテクチャが、ETSI GS NFV-EVE 005 V1.1.1 において提案されている [8]。

2.3 VN 及びサービス機能チェーンの構築

VNO は、アプリケーションサービス提供者からの仮想ネットワーク構築要求を受信した際に、QoS 要求レベルとネットワーク状況等を基に、エッジ・コア・データセンタで構成される物理ネットワーク上に VN を構築する [10]-[15]。VN 構築の際、論理トポロジ作成及び仮想的なネットワーク資源・計算機資源の選択が行われ、VNO が InP に対して物理資源を要求する。InP は、図 1 に示すような、仮想資源から物理資源へのマッピング処理 (VN エンベディング) を行い、その結果を基に、VNO に対して物理資源を提供する。

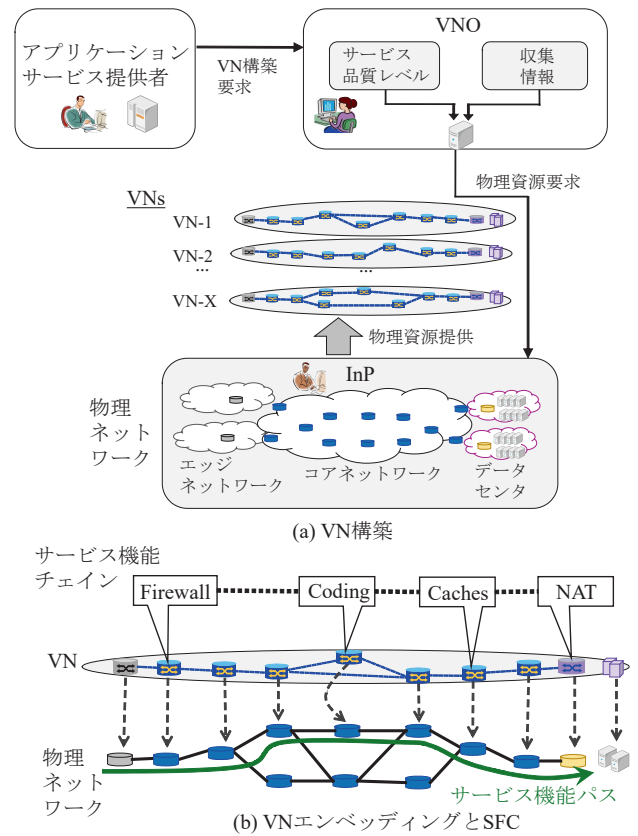


図 1 VN 及びサービス機能チェーンの構築

各 VN 上では、複数のノード上に様々な vNFs を分散配置する SFC を行うことで、伝送パケットに対して複数ノードでネットワーク内処理を施すためのサービス機能チェーンを仮想的に構築できる。そして、パケットの転送経路であるサービス機能パスが決定され、初期 VN 構築のための各種機器設定が行われる。

2.4 自動化の必要性

現在の手動によるネットワーク構築及び再構成は通常、長期間 (2 週間程度) を要することが多く [16]、VN 上のサービス機能チェーンの構築も同様の期間を要することが考えられる。したがって、ネットワーク設定時間の短縮による迅速なサービス提供、さらには人為ミス等を回避するためには、VN 及びサービス機能チェーンの構築制御の自動化が必須である。また、IoT アプリケーションの急速な普及に伴い、データ生成機器やサービス品質 (QoS) 要求が今後ますます多様化するため、時々刻々と変動する通信トラフィック及び様々な QoS 要求の双方を認知して、動的かつ自動で VN を再構成する仕組みが必要である。その際、InP は、各 vNF に対して割り当てられた計算機資源 (例: CPU) の量を適時適切かつ迅速に動的増減することで、SFC 基盤上でトラフィックを効率的に処理することができる。

表1に示すように、リソース制御の自動化技術は、主に、抽象化(abstraction)、配分(allocation)、調整(adjustment)及び調停(arbitration)の4つの機能要素で構成され、それらによって、VNを、トラフィック量や処理負荷、リソース状況等の変動に適応させること(adaptation)が可能となる。3では、計算機資源の自動調整及び自動調停による、VNの3種類の制御手法(Vertical scaling, Horizontal scaling, Internet-network scaling)について紹介する。

表1 VNの資源制御自動化の機能要素

Functions	Description
抽象化 (abstraction)	不均一で分散された仮想資源を簡易な形式で表現し、簡易なインターフェースで情報開示
配分 (allocation)	資源の選択、予約、割当て
調整 (adjustment)	各サービス用VN内部における、ネットワーク環境変動に応じた資源の追加、削除、増減、移行等
調停 (arbitration)	複数のサービスVNsに対して、有限な資源を適時適切に動的配分

3 計算機資源の自動調整技術

3.1 Vertical Scaling

本節では、ディレクトリサービスの機能に対する仮想計算機資源の動的配分のための仕組みについて記述する。我々は、IoTデバイスのプロファイル情報をレコードとして蓄積するためのIoTディレクトリサービスについて研究開発を行っている[17]-[20]。

3.1.1 IoTディレクトリサービス

IoTディレクトリシステムによって、多様な属性を含む膨大な数のレコードを蓄積でき、かつ自動運転のようなIoTアプリケーションの通信遅延に対する厳しい要求を満たすような高速な情報検索や情報更新が可能である。IoTディレクトリサービスに対するレコードの検索要求時にIoTアプリケーションクライアント間の通信遅延を低減するために、レコードが、遠くの位置にあるディレクトリサーバから、IoTアプリケーションクライアントの近くにありオンデマンドに仮想計算機資源上に生成されるキャッシュサーバにコピーされる。IoTディレクトリサービスにおいて、トラフィック量変動しても、優れた資源利用効率を得つつ所望の通信遅延性能を維持するために、動的な計算機資源配分が必要である。キャッシュサーバは、主に、資源利用状況測定と計算機資源調整の2つの機能を有する。測定ユニットは、トラフィック量や資源利用

状況等を定期的に測定し、コントローラに測定データを提供する。同様に、計算機資源調整ユニットは、コントローラからの要求に基づき、計算機資源の割当量の増減を実行する。

3.1.2 各サービスにおける、通信遅延に基づく動的な計算機資源調整

コントローラ内の分析装置は、測定データと性能要求を入力として、多変量の閾値ベースの動的資源調整アルゴリズムを実行し、瞬時に、必要な計算機資源量を導出する。このアルゴリズムは、図2に示されている、通信遅延と資源利用の関係性のロジックに基づいたものである。資源利用状況としては、“Low”、“Desirable”、“High”の3種類に分類され、資源利用率が“Desirable”の範囲になるように、計算機資源の割当量が動的に調整される。一般に、資源利用率が高いほど検索のレイテンシが大きくなり、一方、資源利用率が低いほど検索のレイテンシが小さくなるが資源利用の効率が悪くなる。前述のアルゴリズムでは、各サービスの品質(通信遅延等)に対する要求に依存して、レイテンシの最大許容値(L_{max})が固定であり、一方で、レイテンシの最小値(L_{min})及び“Desirable”の範囲を決定するための高低の閾値(U_h と U_l)は、当該アルゴリズムによって動的に調整される。

3.1.3 評価

図3に、(a)経過時間(単位:秒)に対する検索レイテンシの平均値(単位:ミリ秒)とトラフィック負荷及び(b)経過時間に対するCPU割当率(単位:%)とCPU利用率(単位:%)を示す[12][20]。CPU割当率は、物理CPU資源量に対して、サーバに割り当てるCPU資源量の割合であり、CPU利用率は、CPU割当量に対して、実際に使用しているCPU資源量の割当てである。図から、動的資源調整アルゴリズムによって、トラフィック負荷が増えても、CPU割当率を調整する

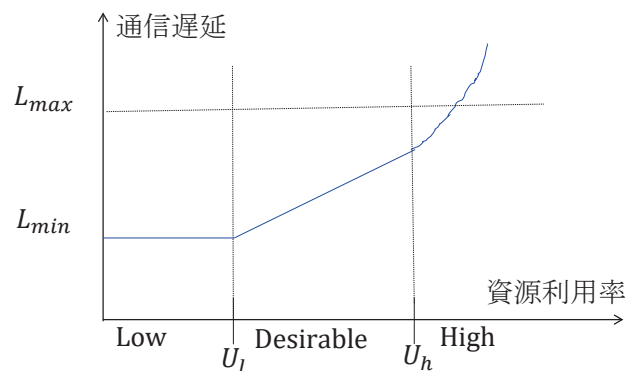


図2 通信遅延 vs 資源利用率

ことによって、10 ミリ秒未満のレイテンシを維持できることが分かる。

3.2 Horizontal Scaling

本節では、VN 基盤において、トラフィック変動時に、サービスを提供する仮想マシンの数を動的に調整する自律資源調整機構 (ARCA: Autonomic Resource Control Architecture) について説明する [21]-[24]。我々が研究開発を行っている ARCA では、資源利用や障害等の情報だけでなく、ネットワーク外部のイベント等の発生を外部イベント検出器によって検知し、その結果も利用して、サービスを提供するのに必要な仮想マシン数を推定し、自律的な計算機資源制御を行う。同時に、閾値ベースの制御アルゴリズムも有しており、トラフィック負荷の計測結果を分析した結果に基づく動的資源調整を、予測ベースの制御調整に優先して行う。

3.2.1 ARCA のフレームワーク

図 4 に、ARCA のコンポーネントとワークフローを示す。ARCA の制御システムは、収集、分析、決定、実行の 4 つの機能を含む。収集機能では、VN 基盤及び外部イベント検出器から得られる監視データを集める。分析機能では、複数の監視データ間を相互に比較し、VN 基盤のトラフィック負荷状況や、非日常ケース

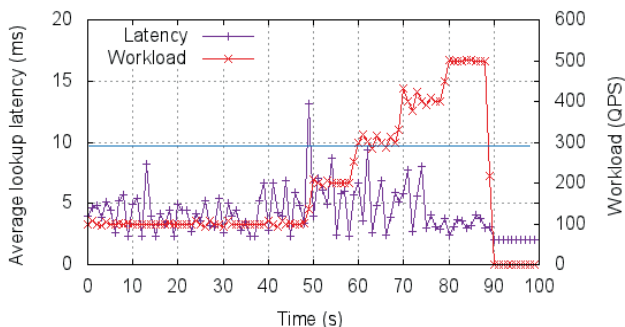
(例: 各種イベントや障害) の発生を認知する。決定機能では、現在のトラフィック負荷または今後の負荷変動予測を基に、計算機資源調整の内容を決定する。実行機能では、決定された制御内容に従って、VN 基盤を制御する。

3.2.2 各サービス用に必要な仮想マシン数の推定

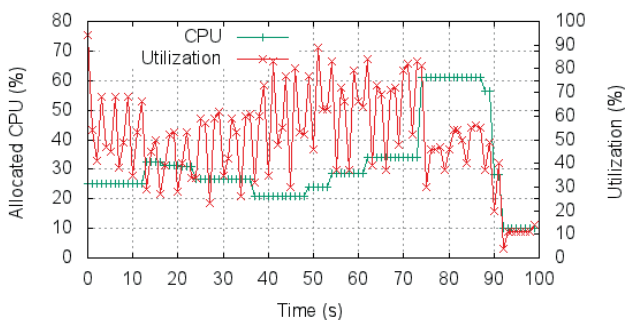
決定機能においては、資源制御に要する時間に起因する遅延発生をなるべく最小化する必要があり、機械学習等を活用して、事前に必要な計算機資源量を推定する仕組みが有効である。収集機能では、例えばカルマンフィルタ等を適用して、無関係なデータを分析対象外とした上で、重要なデータを、分析機能に送る。分析機能では、イベント間の相関、VN 基盤の現在及び今後の予測状況を認知し、決定機能に通知する。決定機能では、例えば機械学習の 1 技法であるサポートベクター回帰 (SVR: Support Vector Regression) 等を適用し、今後必要となる仮想マシン数を推定する [25]。完全な予測は不可能なことから、トラフィック負荷の測定結果を基に閾値ベースでリアクティブに制御する機構も用いて、計算機資源量の再調整を行う。必要な仮想マシン数の推定においては、資源利用効率等を考慮して必要最小限の数を推定するが、前述の非日常ケースの発生等も考慮し、状況に応じて推定ポリシーを適応的に変更可能とする。実行機能では、VN 基盤の仮想環境を直接的に構築制御するシステムに対して、仮想マシン数の調整を要求する。

3.2.3 評価

機械学習ライブラリである SciKit-Learn (<http://scikit-learn.org>) の SVR 機能を使って、必要な仮想マ



(a) 検索レイテンシ及びトラフィック負荷



(b) CPU割当率及びCPU利用率

図 3 検索レイテンシ平均値、トラフィック負荷、CPU 割当率及び CPU 利用率

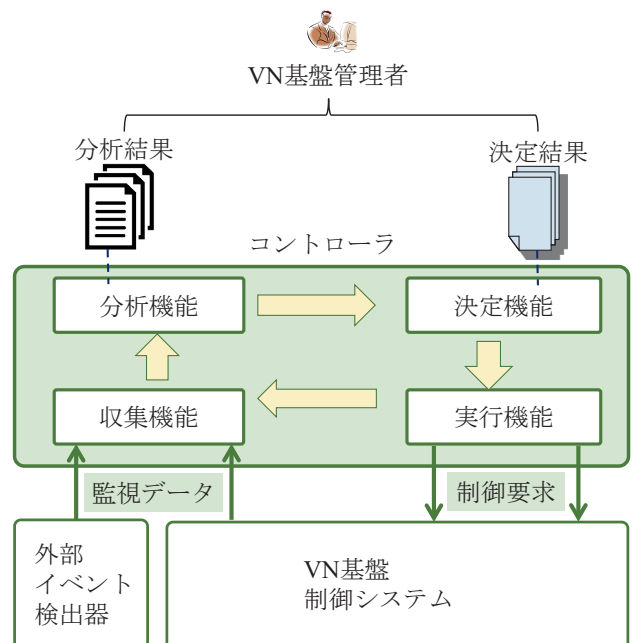


図 4 ARCA のコンポーネントとワークフロー

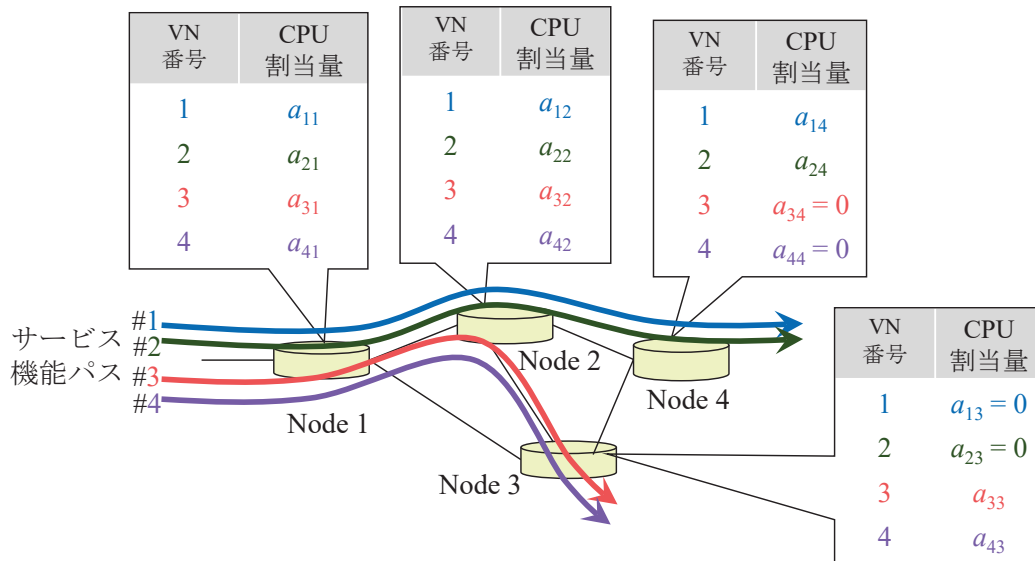


図5 複数 VNs とサービス機能チェーンの構築及び CPU 割当てのイメージ

シン数を事前に推定する提案方式を実装した。その上で、資源に対する需要が急増するような非日常ケースのイベントも発生させて、リアクティブに計算機資源調整する方式と、サービス要求の棄却率について定量的に比較した [23]。その結果、リアクティブ方式でサービス要求が約 1 % 棄却する状況において、事前に計算機資源需要予測を行う提案方式は、各サービスに対して計算機資源を約 15 % 余分に割り当てることによって、サービス要求の棄却率をほぼ回避できることが分かった。

3.3 Internetwork Scaling

本節では、共通の物理ネットワーク上で、複数のサービスネットワーク (VNs) 間で計算機資源を共有するための動的資源調停技術について説明する [10][11]。さらに、各ネットワークノード内の複数サービス間資源調停を実行してもサービス品質維持ができない場合において、ネットワーク機能を別のネットワークノードに動的に移行する技術についても説明する。各 VN 上では、SFC によって、複数の vNFs から構成されるサービス機能チェーンが構築される。各 vNF を動かす仮想マシンにおける計算機資源 (例: CPU) は、トラフィック負荷や資源利用状況によって、動的に調整 (増加もしくは減少) する必要がある。さらに、データプレーンにおけるトポロジ再構成やスイッチ切替等は時間を要求することが多い [26] ので、サービスを維持するため、なるべくデータプレーン上の経路 (データを転送するリンクと物理スイッチ) を変更することなく、計算機資源を調整することが望ましい。

図 5 に、複数 VNs とサービス機能チェーンの構築及び CPU 割当てのイメージを示す。4 つのネットワー

クノードで構成されたネットワーク上で、4 つの VNs が構築され、各ネットワークノード内でサービス機能を実行するための物理計算機資源は、複数の仮想マシンを作成することで共有される。多様なサービス要求が混在する環境を想定し、VNs は各々、QoS 要求が異なる。例えば、高 QoS (最高優先)、中程度 QoS、低 QoS (ベストエフォート) の 3 レベルへの分類が考えられる。この例では、各 VN は、複数のネットワークノードをまたいで、1 つのサービス機能チェーンを構築する。例えば、VN 1 の例では、ネットワークノード 1、2、4 をまたいでサービス機能チェーン (パス #1) を構築し、各ノードで異なる NF を動かす。VN x に対して、ネットワークノード y で割り当てられている CPU 資源量を、 a_{xy} とする。

3.3.1 同一ネットワークノード内の複数 VNs 間の動的資源調停

動的資源調停プロセスでは、各ネットワークノードにおいて、ある VN に割り当てられた仮想マシンの CPU 使用量が増加し、あらかじめ設定された閾値 (θ_h) を超えた時に、別の VN に割り当てられた仮想マシンで、CPU 使用量が小さく他 VN に資源譲与可能な CPU 資源があった場合に、後者の CPU 資源量を減らし、前者の CPU 資源量を同量だけ増やすことで、調停を実現する。ここで、我々の提案方式では、QoS 要求レベルが高いほど、優先的にサービス品質を維持すべく CPU 資源が配分される。資源譲与可能か否かを判定するための閾値 (θ_l) はあらかじめ設定され、各 VN の仮想マシンの CPU 使用量が、 θ_l 以上 θ_h 以下となるように、複数 VNs 間で資源調停がなされ、前述の a_{xy} の値が調整される。閾値については、制御ポリシーに基づいて、柔軟に変更可能である。図 6 に、資源

調停の例を示す。各ネットワークノードでは、どのVNも使っていない(未使用の)CPU資源はないものとする。ネットワークノード1では、VN 1のCPU使用量が大きくCPU資源増加が必要で、VN 4のCPU使用量が小さく資源譲与可能と想定する。VN 1はCPU資源量 α だけ増やす必要があり、VN 4では譲与可能なCPU資源量が α 以上の場合、VN 4のCPU資源量を α だけ減らすことによって、VN 1のCPU資源量を α 増やす。もしVN 4の譲与可能資源量が α 未満だった場合は、他のVNの譲与可能資源も探索し、CPU資源が余っていたら、他VNからも譲与される。ネットワークノード2では、VN 2のCPU使用量が大きくCPU資源増加が必要で、VN 3のCPU使用量が小さく資源譲与可能と想定する。ネットワークノード1のケースと同様、VN 2はCPU資源量 β だけ増やす必要があり、VN 3では譲与可能なCPU資源量が β 以上の場合、VN 3のCPU資源量を β だけ減らすことによって、VN 2のCPU資源量を β 増やす。

CPU資源量の調整は、各VNのQoS要求レベルに依存する。前述のCPU割当量 a_{xy} に関して、時間 t における割当量を $a_{xy}(t)$ と表す。一方で、時間 t におけるCPU使用量を $u_{xy}(t)$ と表す。時間 t におけるVN x のネットワークノード y での資源利用状況は、下記のとおりである。

- *Saturation*: $u_{xy}(t) = a_{xy}(t)$
- *Ideal*: $(u_{xy}(t) + m(q)) = a_{xy}(t)$
- *High*: $(u_{xy}(t) + m(q)) > a_{xy}(t)$ and $u_{xy}(t) < a_{xy}(t)$
- *Low*: $(u_{xy}(t) + m(q)) < a_{xy}(t)$

ここで、 $m(q)$ はQoS要求レベル q のVNs に対するマージン値である。我々の提案方式では、QoS要求レベルが高いVNのCPU飽和発生をなるべく回避するために、QoS要求レベルが高いほど $m(q)$ の値を高く設定するといった傾斜配分を行う [10][11]。なお、 $m(q)$ の値は、当該VNにおいては、どのノードにおいても同じ値である。各VN及び各ネットワークノードにおいて、上記を基に、資源利用状況の判定がなされる。判定の結果、「High」の場合、そのVNにおいては、 $(u_{xy}(t) - a_{xy}(t) + m(q))$ だけCPU資源量を増加する必要がある。「Low」の場合、そのVNは、 $(a_{xy}(t) - u_{xy}(t) - m(q))$ だけ、他の資源逼迫中のVNsにCPU資源を譲与可能である。

前述のとおり、QoS要求レベルが高いほど優先的にサービス品質を維持すべくCPU資源量が増加されるが、一方で、他のVNで譲与可能なCPU資源が複数ある場合に、どのVNのCPU資源を減らすかという課題がある。QoS要求レベルが低いVNから優先的に減らす、同じQoSレベルのVNsの中でどの

CPU資源を減らすかについては、最も簡易な方法は、ランダムに資源譲与するVNを決定する手法(ランダム選択法)である。しかしながら、ランダム選択法では、譲与可能なCPU資源量が小さい複数のVNsから譲与する可能性があり、CPU資源量増減の調整回数が増えてしまう問題がある。そこで、我々の提案方式では、同じQoSレベルのVNsの中では、譲与可能資源量の最も大きいVNからCPU資源を減らす手法(決定的選択法)を用いる [10][11]。

3.3.2 各VNにおける複数ネットワークノード間の動的機能移行

3.3.1で説明したネットワークノード内のVNs間資源調停を実行してもなお、CPU飽和状態のVNが存在する場合、当該VNにおいて、ネットワークノード間でNFの移行(マイグレーション)を行う。ただし、我々の提案方式では、前述のとおり、まずはデータプレーン上の経路(データを転送するリンクと物理スイッチ)を変更することなく、機能マイグレーションする。また、どの移行先候補ノードにおいても十分な計算機資源が無い場合は、NF移行は行わない。図7に、NF移行の例を示す。この例では、ネットワークノード1において、VN 1がCPU飽和状態である。ネットワークノード2に十分な計算機資源があることから、VN 1のNFがネットワークノード2に移行され、その分、VN 1の計算機資源量を増加する。NF移行においては、まず初めに、移行先ノードにおいて当該VNのために必要な計算機資源が確保された上で、コントローラからNFオフローディングの要求が送られ、NFが起動する。次に、移行元ノードにおいてコントローラからの要求に従ってNFが停止し、仮想マシンの計算機資源が解放される。

3.3.3 評価

図8は、1ノード内のCPU飽和発生回数について、固定資源割当(Static)、各ノード内の資源調停のみ実

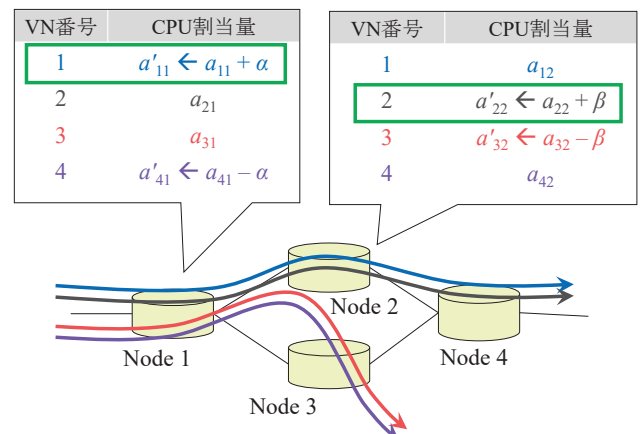


図6 各ネットワークノードにおけるVNs間の資源調停のイメージ

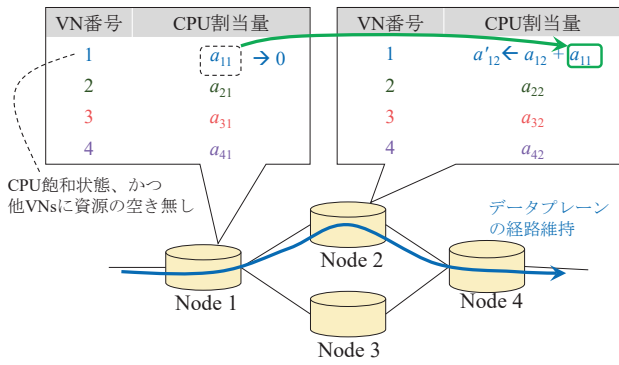


図7 各VNにおけるネットワークノード間の機能移行のイメージ

行する方式 (Dynamic-1) 及びノード内資源調停とノード間機能移行の双方を実行する方式 (Dynamic-2) の計3方式を、シミュレーションにより定量的に比較した結果である。シミュレーションでは、CPU使用量の変動に伴う資源調整を500回試行した際に、何回CPU飽和状態が発生したかを導出した。VNs数を10とし、VN 1、2、3が高QoSレベル ($q=1$)、VN 4、5、6が中QoSレベル ($q=2$)、VN 7、8、9、10が低QoSレベル ($q=3$) とする。Dynamic方式において、 $(m(1) - m(2))$ 及び $(m(2) - m(3))$ を d とし、 d の値を変更させて、5種類の特性を導出した。各種パラメータ等の前提条件や設定については、文献 [10][11] に詳述されている。Dynamic方式は、Static方式と比較して、時間変動するCPU使用状況に応じてNFへのCPU割当量を動的に調整することによって、CPU飽和発生回数を低減できている。さらに、Dynamic-2は、全QoS要求レベルにおいて、Dynamic-1よりも、CPU飽和発生回数を低減できている。また、提案する資源傾斜配分法によって、QoS差別化が可能であり、かつ、より高いQoSを要求するVNsの特性を改善できている。

4 まとめ

5G・IoT時代において、今後、ユーザのサービス要求の多様化により、通信量がますます増加し、かつ時々刻々と変動することが予想される。運用・管理データの増大やシステム複雑化、ネットワーク設計や再構成等の複雑化に対処するためには、機械学習等を活用し、各サービスを提供するための仮想ネットワークやサービス機能チェーンの構築・制御(特に計算機資源調整)・運用管理の自動化及び高速化を図ることで、迅速なサービス提供や早期障害復旧、通信量変動への適応化を実現する必要がある。本稿では、仮想ネットワークの構築制御における自動化技術に関する研究開

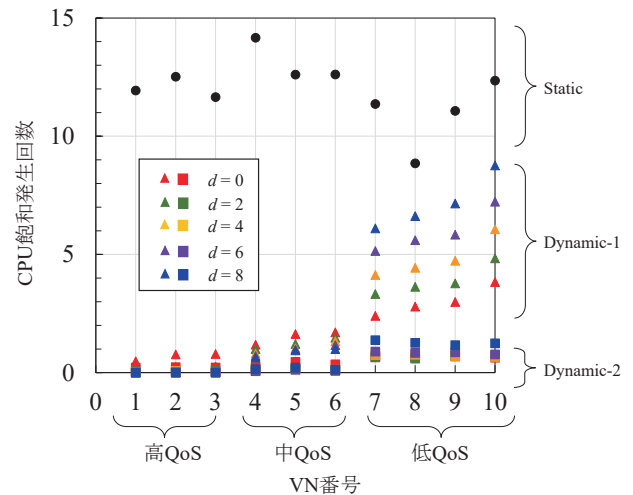


図8 CPU飽和発生回数に関する評価結果

発成果の一部を紹介した。今後、例えば、サービス機能チェーンの計算機資源の調整制御においては、いかに、機械学習等を駆使した高度自動化によって、サービス品質の維持向上や制御処理の高速化を実現するか、といった課題が挙げられる。

謝辞

本研究は、原井洋明総合テストベッド研究開発推進センター長、地引昌弘主任研究員、Pedro Martinez-Julia研究員、福島裕介研究員、Abu Hena Al Mukhtadir研究員、平山孝弘研究員、藤川賢治主任研究員、小針康永研究技術員らと実施した。また、研究開発にご協力いただいた関係各位に感謝する。

【参考文献】

- 1 <https://www.etsi.org/technologies-clusters/technologies/zero-touch-network-service-management>.
- 2 R. Sherwood, G. Gibb, K. Yap, G. Appenzeller, M. Casado, N. McKeown, and G. Parulkar, "FlowVisor: A Network Virtualization Layer," OpenFlow Switch Consortium, Technical Report, Oct. 2009.
- 3 M. Chowdhury, M. R. Rahman, and R. Boutaba, "ViNEYard: Virtual Network Embedding Algorithms With Coordinated Node and Link Mapping," IEEE/ACM Trans. on Networking, vol.20, no.1, pp.206-219, Feb. 2012.
- 4 B. Han, V. Gopalakrishnan, L. Ji, and S. Lee, "Network Function Virtualization: Challenges and Opportunities for Innovations," IEEE Communications Magazine, vol.53, issue 2, pp.90-97, Feb. 2015.
- 5 T. Taleb, B. Mada, M. Corici, A. Nakao, and H. Flinck, "PERMIT: Network Slicing for Personalized 5G Mobile Telecommunications," IEEE Communications Magazine, vol.55, no.5, pp.88-93, May 2017.
- 6 IETF RFC 7665 "Service Function Chaining (SFC) architecture," Oct. 2015.
- 7 IETF RFC 8300 "Network Service Header (NSH)," Jan. 2018.
- 8 ETSI GS NFV-EVE 005 V1.1.1 "Network Functions Virtualisation (NFV); Ecosystem; Report on SDN Usage in NFV Architectural Framework", Dec. 2015.
- 9 A. M. Medhat, T. Taleb, A. Elmangoush, G. A. Carella, S. Covaci, and T. Magedanz, "Service Function Chaining in Next Generation Networks: State of the Art and Research Challenges," IEEE Communications Magazine, vol.55, issue 2, pp.216-223, Feb. 2017.

6 ネットワークの効率的な資源配分を目指す研究開発

- 10 宮澤高也, 地引昌弘, ベド カフレ, 原井洋明, "ネットワーク仮想化におけるサービス機能チェーンに沿った自動資源調停機構," 電子情報通信学会, 信学技報, vol.117, no.303, NS2017-124, pp.73-78, 2017年11月.
- 11 T. Miyazawa, M. Jibiki, V.P. Kafle, and H. Harai, "Autonomic Resource Arbitration and Service-continuable Network Function Migration along Service Function Chains," IEEE/IFIP Network Operations and Management Symposium (NOMS2018), Taipei, Taiwan, pp.1-9, April 2018.
- 12 V. P. Kafle, Y. Fukushima, P. Martinez-Julia, T. Miyazawa, and H. Harai, "Adaptive Virtual Network Slices for Diverse IoT Services," IEEE Communications Standards Magazine, Sept., 2018.
- 13 K. Fujikawa, V. P. Kafle, P. Martinez-Julia, A. H. A. Muktadir, and H. Harai, "Automatic Construction of Name-Bound Virtual Networks for IoT," IEEE 41st Annual Computer Software and Applications Conference (COMPSAC), Turin, Italy, pp.529-537, July 2017.
- 14 A. H. A. Muktadir, T. Miyazawa, P. Martinez-Julia, V. P. Kafle, and H. Harai, "Multi-QoS Compliant Virtual Resource Recommendation Scheme," 電子情報通信学会ネットワークシステム研究会信学技報, vol.117, no.303, NS2017-125, pp.79-84, 2017年11月.
- 15 T. Miyazawa, V. P. Kafle, and H. Harai, "Reinforcement Learning Based Dynamic Resource Migration for Virtual Networks," Proc. of 15th IFIP/IEEE International Symposium on Integrated Network Management (IM 2017), Lisbon, Portugal, pp.428-434, May 2017.
- 16 K. Katsuura, M. Miyauchi, T. Numazaki, Y. Kurogouchi, Y. Satoh, and T. Koseki, "IaaS Automated Operations Management Solutions That Improve Virtual Environment Efficiency," NEC Technical Journal, vol.8, no.2, pp.29-32, April 2014.
- 17 ベド カフレ, 福島裕介, ペドロ マルティネスフリア, 原井洋明, "将来IoTアプリケーションのためのディレクトリサービスの設計と実装," 電子情報通信学会, 信学技報, vol.116, no.485, IN2016-135, pp.227-232, 2017年3月.
- 18 V. P. Kafle, Y. Fukushima, P. Martinez-Julia, and H. Harai, "Scalable Directory Service for IoT Applications," IEEE Commun. Standards Mag., vol.1, no.3, pp.58-65, Sept. 2017.
- 19 ベド カフレ, 福島裕介, ペドロ マルティネスフリア, 原井洋明, "IoTディレクトリサービスのための資源割当と動的調整," 電子情報通信学会, 信学技報, vol.117, no.262, NS2017-90, pp.7-12, 2017年10月.
- 20 ベド カフレ, 福島裕介, ペドロ マルティネスフリア, 原井洋明, "IoTディレクトリサービスのための資源動的調整," 電子情報通信学会, 信学技報, vol.117, no.459, NS2017-222, pp.305-310, 2018年3月.
- 21 P. Martinez-Julia, V. P. Kafle, and H. Harai, "Achieving the autonomic adaptation of resources in virtualized network environments," Conference on Innovation in Clouds, Internet and Networks (ICIN2017), Paris, France, pp.52-59, Feb. 2017.
- 22 Pedro Martinez-Julia, Ved P. Kafle, and Hiroaki Harai, "Adapting OpenStack-Based Virtual Computer and Network Systems to Dynamic Demands," 電子情報通信学会, 信学技報, vol.117, no.204, NS2017-73, pp.13-18, 2017年9月.
- 23 P. Martinez-Julia, V. P. Kafle, and H. Harai, "Anticipating Minimum Resource Needed to Avoid Service Disruption of Emergency Support Systems," Conference on Innovation in Clouds, Internet and Networks (ICIN2018), Paris, France, pp.1-8, Feb. 2018.
- 24 P. Martinez-Julia, V. P. Kafle, and H. Harai, "Exploiting External Events for Resource Adaptation in Virtual Computer and Network Systems," IEEE Transactions on Network and Service Management, vol.15, issue 2, pp.555-566, June 2018.
- 25 U. Thissen, R. van Brakel, A.P. de Weijer, W.J. Melssen, and L.M.C. Buydens, "Using Support Vector Machines for Time Series Prediction," Chemometrics and Intelligent Laboratory Systems, vol.69, no.1-2, pp.35-49, Nov. 2003.
- 26 "Cisco ONS 15454 DWDM Network Configuration Guide, Release 10.x.x," Chapter: Node Reference, Jan. 2018.



宮澤高也 (みやざわ たかや)

ネットワークシステム研究所
ネットワーク基盤研究室
主任研究員
博士(工学)
ネットワーク制御、ネットワークシステム、
ネットワーク管理



Ved P. Kafle (べど かふれ)

ネットワークシステム研究所
ネットワーク基盤研究室
研究マネージャー
博士(情報学)
ネットワークアーキテクチャ、ネットワーク
システム、モバイルネットワーク