

SDN/NFV ソリューションの研究開発・実証・商用化と今後の展望*

岩田 淳^{†a)} 下西 英之^{††b)} 小林 正好^{†††c)}

SDN/NFV Solution Research, Trial, Commercialization and Its Future Vision*

Atsushi IWATA^{†a)}, Hideyuki SHIMONISHI^{††b)}, and Masayoshi KOBAYASHI^{†††c)}

あらまし 本論文では、ネットワーク業界に20年に1度と言われる革新をもたらしたSDN, NFVにおける研究開発・実証・商用化と本領域の将来の発展の方向性について述べる。現在のインターネットは自律分散制御方式を採用し、個々のネットワーク装置に機能が埋め込まれているため、機能追加・変更が困難な上、装置の機能肥大化を招いている。またデータセンタ等の仮想化が進む環境では、頻繁に生じるサーバやストレージの構成変更迅速に追従するネットワーク構築・運用が求められるが、自律分散制御に起因する遅延や振る舞いの予測困難性により、迅速、確実なネットワーク変更ができないという課題がある。本課題に対し、筆者らは論理集中型のプログラマブルな制御方式によるネットワークの設計・構築・運用(SDN)と最小限の標準的枠組み(OpenFlow)とによる解決策を提唱し、更にネットワーク装置の仮想化(NFV)と組み合わせ、ネットワークサービスでの高機能化と柔軟性の実証・実用化に成功した。更に、IoTなど実世界のデータをセンシング・解析・最適化する際の広域分散データ解析プラットフォームへの本技術の適用へ向け、将来を展望する。

キーワード OpenFlow, SDN, NFV, プログラマブル, 設計

1. ま え が き

グローバルな成長ビジネス領域であるクラウド基盤領域、クラウドサービス領域、テレコム領域において、それぞれ新たな成長を目指した動きが活発に行われている[1]~[4]。

クラウド基盤領域では、Googleがルータ・スイッチベンダの装置を購入する対応では、(1)12~15ヶ月ごとに帯域が倍に伸びるデマンドに対しコスト対性能比に追従できない、(2)分散型プロトコルに対応する設定・運用対応では運用コストを低減できない、といっ

た課題に直面し、市販の通信デバイスを活用して専用スイッチを自ら開発し、データセンタ内(Jupiter [1]), データセンタ間(B4 [2])で用いる装置コストを低減し、集中型制御形態によりネットワーク利用率を限界まで向上、運用コストの大幅削減を実運用中である。

クラウドサービス領域では、FacebookがOpen Compute Project (OCP) [3]を立ち上げ、データセンタ内で用いるラックマウントサーバ並びにTop-of-Rack Switch (TOR)の設計をOpen化・革新を牽引し、それぞれ性能・消費電力・制御性を大幅に向上させている。更に、Telecom Infra Project (TIP) [4]では、テレコムオペレータのワイヤレス・バックボーンインフラをOpen化し、低価格、大容量、最適な運用を可能とする試み(OpenCellular, Open Optical Packet Transport)が2016年3月から開始されている。

テレコム領域の例では、AT&T Domain 2.0が、古いテレコム気質を変え、標準化[5],[6], Open Source Software (OSS) 団体[7]~[10]に深く入りこみ、Open Innovationによりテレコムインフラを革新するプロジェクトを推進中である。

このように、サービスを運用する際のネットワーク装置のコスト削減、運用性向上を徹底的に追求する動き

[†] NEC アメリカ, 米国

Optical IP Development Division, NEC Corporation of America, 3151 Jay Street, Suite 110, Santa Clara, CA 95054, USA

^{††} 日本電気株式会社システムプラットフォーム研究所, 川崎市 System Platform Research Laboratories, NEC Corporation, Kawasaki-shi, 211-8666 Japan

^{†††} 日本電気株式会社技術イノベーション戦略本部, 東京都 Corporate Technology Division, NEC Corporation, Tokyo, 108-8001 Japan

a) E-mail: Atsushi.Iwata@necam.com

b) E-mail: h-shimonishi@cd.jp.nec.com

c) E-mail: m-kobayashi@eo.jp.nec.com

* 本論文は、システム開発・ソフトウェア開発論文である。

DOI:10.14923/transcomj.2016SHI0004

が各所で活発化しており、その動きを牽引しているのが、Software Defined Networking (SDN), Network Function Virtualization (NFV) 技術である。

SDN の代表的方式である OpenFlow [11], [12] は、2007 年頃から Stanford 大学, UC Berkeley 大学で研究開始以来、SIGCOMM, NSDI 等の国際学会, GENI [13] 等の研究コミュニティにて、新たなネットワーク制御の研究開発・トライアルが進められ、Open Networking Foundation (ONF) で仕様が標準化 [12], 各社から企業, データセンター, クラウド, テレコムキャリア向けの OpenFlow 対応の商用製品が提供されている。

OpenFlow は、従来のルータ・スイッチ等のネットワークノード機能を C-plane (ネットワーク制御機能) と U-plane (データ転送機能) とに分離できるようにインタフェースをオープン化し、C-plane 部分にネットワーク制御ソフトウェアを自由に構築可能にする基盤技術である。これにより、従来のルータ・スイッチではできなかった新たなネットワーク制御、例えば集中型・Fabric 型での最適ネットワーク制御や Agile なネットワーク設計・構築・運用が可能となり、ネットワーク運用者の新たなニーズへ迅速に対応可能である。

例えば、Google の B4 [2] では BGP/ISIS の既存プロトコルと OpenFlow プロトコルを統合運用して WAN 環境へ適用し、従来技術だと 30–40% のネットワーク利用率のところを 100% に近い利用率まで引き上げに成功している。また、Facebook の OCP では Wedge スイッチ内部に Data Plane Abstraction 層を設け OpenNSL [14], OF-DPA [15] 等データプレーンを Open に制御する機構を設けており、OF-DPA が OpenFlow 向け制御機構となる。

NFV の業界としての動きは、2012 年 11 月に、BT, DT, FT, Telefonica, AT&T, Verizon, NTT, KDDI 他のテレコムオペレータの提案により、ETSI Industry Specification Group (ISG) において Network Function Virtualization (NFV) WG [6] を発足したのが始まりである。2013–2014 年の 2 年間で NFV の Architecture, Requirement について各種仕様を Release 1 として発行。その後 2015–2016 年の Release 2, 2016 年以後の Release 3 と継続中である。

NFV の背景は、従来キャリア網内にある様々なネットワークアプライアンス装置がハードウェアベースの専用装置で、サービス設計・構築・運用が固定的かつ柔軟性がなく、これらの装置を仮想化技術により

ソフトウェアベースに置き換え、サービスの変化に合わせた柔軟なインフラの設計・構築・運用による CAPEX(設備コスト)/OPEX(運用コスト) 削減を実現するニーズである。ユースケースは、モバイルコア仮想化の Virtual Evolved Packet Core (vEPC), 無線アクセス仮想化の Virtualized Radio Access Network (vRAN), 固定アクセス仮想化の Virtual Customer Premise Equipment (vCPE) 等があり、2016 年現在 41 種類の Proofs-of-Concept (PoC) の実装、トライアルが継続的に行われている。

例えば、2016 年現在 AT&T は ON.Lab にてテレコムキャリア局舎仮想化プロジェクト Central Office Re-Architected as a DataCenter (CORD) [7] を牽引し、ネットワーク装置の仮想化機能部 (VNF), システム全体のオーケストレーション (XOS), SDN によるネットワーク制御 (ONOS [10]) との統合により、固定アクセス網向けの仮想化である Residential-CORD (R-CORD) [7] のフィールド実験を継続中である。

本論文では、ネットワーク業界を大きく動かした SDN/NFV 領域に対して筆者らが取り組んできた研究開発、実証、商用化について、どのように課題を克服して実用化したかを述べ、現在更なる発展を目指し鋭意取り組んでいる研究領域について述べる。以下、**2. 3.** にて、それぞれ SDN, NFV における市場ニーズ・技術トレンド、研究開発・実証、商用化へ向けた取り組みを述べ、**4.** にて、最新の SDN/NFV 研究動向を俯瞰し将来の展望を述べる。

2. SDN 研究開発・実証へ向けた取り組み

2.1 市場ニーズ・技術トレンド

筆者らは、OpenFlow/SDN の黎明期の 2007 年から、米国 Stanford 大学と Open Innovation 型共同研究体制を組み、SDN コンセプトの策定と中核となる OpenFlow 技術の研究開発・標準化・実証を行った。2007 年頃に Stanford 大学で開始された Clean Slate Program と呼ぶ研究プログラムの中に要素研究テーマとして OpenFlow [11] が立ち上げられた。

OpenFlow/SDN の原点は Ethane [16] でのスイッチの Data plane と Control plane を分離し、Campus network 向け Access Control を実現した方式である。OpenFlow [11] は Ethane 方式を一般化し、Data plane と Control plane を分離し、Control plane 側に多様なネットワーク制御、マネジメント制御機能を具備できるようなネットワーク装置の Abstraction

機能、複数のネットワーク装置の Network map 機能をもった Network OS を提供することで、新たなネットワークのイノベーション創出を目指した。2007年当時は、米国での FIND, GENI, 欧州の FP7, FIRE, 日本の NWGN, JGN2plus など研究プログラムとそれをフィールド実証するプログラムとの両面で、世界中で新たなネットワーク研究へ向けた動きが活発化していた時代である。筆者らはその中で、OpenFlow の考え方に新たなネットワークのパラダイムシフトの可能性を見出し、Stanford 大学に研究員を常駐させ、Stanford 大学-NEC での共同研究体制を開始した。業界にパラダイムシフトを起こすには、最終的には業界が賛同する Open Platform を生み出し、その上での新たなイノベーションの機運を起こす必要があり、その実現は Silicon Valley を中心に Open Innovation にて業界をリードするのが得策という判断であった。

2.2 Open Innovation による Agile 研究開発

OpenFlow/SDN のベースとなる Flow-based networking を実現するため、Broadcom 社の Ethernet switch ASIC のアーキテクチャを参照し、市販のチップで実装可能なフローテーブルとその制御機構を考慮した OpenFlow 仕様を設計した。Stanford 大学での POC, Campus Trial, 全米トライアルと順次規模を拡張し、機能面・性能面・運用面の観点で仕様を繰り返し更新し、2009年に最新版の OpenFlow 1.0 [17] 策定に成功した。その後業界として立ち上げた標準化団体 ONF へ仕様策定を引き継ぎ、現在 OpenFlow 仕様は 1.5.1 [12] へ拡張されている。

第1段階の POC フェーズでは、Stanford, NICT JGN2plus にそれぞれ複数台の OpenFlow Switch を配備し、L2 Tunneling でつないだ実験網を構築し、全体をコントロールする OpenFlow Controller を配備した。基本的な OpenFlow 動作の実証や、太平洋をまたいだ VM Migration のデモなどを実現した。本活動で、NEC は世界初の OpenFlow スイッチハードウェア試作機を完成させ実験加速に貢献した。第2段階の Campus Trial フェーズでは、キャンパス内の生活用トラヒックと、複数の OpenFlow 実験用のトラヒックを混在させた環境を構築するため、FlowVisor と呼ぶネットワークの Slicing 機能を開発し導入した。それぞれの Slice に異なる OpenFlow Controller を置き、多様な研究 (Plug-N-Serve, OpenRoads, Aggregation, OpenPipes) を単一インフラで同時に実験・実証することでその有効性を実証した。

第3段階の全米トライアルでは、GENI [13] の基盤を活用し Stanford, Georgia Tech, Princeton, Rutgers 大学等北米8大学、バックボーンとして Internet2, NLR を活用した実験網が整備され、コンピューティング基盤である PlanetLab とネットワーク基盤である OpenFlow ネットワークを統合した環境構築に成功した。既存網が併存する全米トライアル網構築において、機能・性能の拡充要件など知見を多く得ることができた。これと並行し日本での NICT JGN2plus, JGN-X, ヨーロッパでの FP7 OFELIA, ブラジルでの CPqD でのトライアルなど全世界にトライアルが拡大した。本トライアルを通し、実用レベルの利用を目指した商用化向け要件の獲得に成功した。それぞれの段階での詳細は論文 [18] を参照されたい。筆者らは本トライアルで仕様設計、実装、デバッグ、運用等の主要な役割を果たすとともに、Software-Defined なシステムが構築可能なことを世界で初めて実証し、ソフトウェアベンダが主導する革新的なインフラ構築への業界変革への道筋を示すのに成功した。

2.3 SDN の差異化領域の研究開発

上述の Stanford 大学との共同研究を遂行しながら、NEC 内での研究開発は以下のように OpenFlow/SDN Open Platform 上での差異化領域にフォーカスして実施した。大きく分類すると、SDN コントローラ基盤に関する研究 (第1, 2, 3世代)、並びに SDN 向けネットワーク制御方式に関する研究とに分類できる。

(1) SDN コントローラ基盤方式 (第1世代)

第1世代はコントローラアーキテクチャを模索した時代である。当時は OpenFlow/SDN コントローラとして理想のアーキテクチャ像が業界として確立されていない時代で、Stanford 大学でのコントローラの実験も活用しつつアーキテクチャ視点での研究開発を実施した。Stanford で活用していた OpenFlow Controller NOX [19] での課題を分析し、Monolithic なアーキテクチャの課題 (システム再起動なしにはコントローラ上のアプリケーションの追加・削除が不可) の克服とソフトウェア開発者に開発容易な環境の提供の両者を目的とし Network OS 型のアーキテクチャを実現するため、以下の四つの課題を解決するコントローラ基盤の方式策定・設計・実装評価の研究 [20] を実施し、JGN2plus の広域 OpenFlow 網でその有効性を実証した。本アーキテクチャは、現在 SDN コントローラの OSS Community である OpenDaylight [8], ONOS [10] などでも採用されている考え方である。

(a) ソフトウェアモジュールの Modularity

ソフトウェアモジュールを自由に組み合わせ可能なモジュール構造へ変える必要性

(b) 多様なネットワーク装置対応の抽象化技術

仮想化基盤 FlowVisor [21] が OpenFlow スイッチのみの抽象化しか対応できず、OpenFlow 対応機器以外のノードも抽象化する制御の必要性

(c) ネットワークの仮想化と分離化制御

上記抽象化技術をベースに、OpenFlow 対応機器以外のノード群をまとめて仮想化・分離化の制御の必要性

(d) 大規模網まで対応するスケラビリティ

NOX は単一コントローラでスケールが原理上不可能で、キャリアクラスの広域・大規模網への適用にスループット、遅延特性を満足する分散型コントローラの必要性。Nicira 社が中心に開発した ONIX [22] と呼ぶ分散型コントローラは、全てのデータのステート管理を共通に実現したためステート管理が肥大化する課題があり、アプリケーションのデータの特徴ごとにステート管理を最適化する分散コントローラの必要性

(2) SDN コントローラ基盤方式 (第 2 世代)

第 2 世代は、コントローラの Open 化により Third Party Developer に新たなユースケース開発を拡大し、ユースケースごとにどのコントローラ方式を用いるのが最適になるのか、キラーユースケースは何かの分析を始めた時代である。このとき OpenFlow Controller の Open Source 化を行い開発をすすめる活動 Trema [23] を実施した。Trema は第 1 世代のアーキテクチャの中でソフトウェアのモジュラリティ、開発環境向上に特化して開発し、C と Ruby にて開発環境を提供した。Wakame などクラウドサービス事業者での採用など、コミュニティ拡大に成功した。

更に、2012 年に NEC Biglobe 向けに、Trema ベースで分散コントローラを開発するプロジェクトを実施した。NEC Biglobe のクラウドサービスの要件から、Web 3 層アーキテクチャに基づく OpenFlow プロトコル終端プレゼンテーション層、ネットワークスイッチ制御処理を行うロジック層、制御情報の保存・複数ワーク間での共有を行うデータ層から構成されるアーキテクチャを考案し、設計・開発を実施した。VXLAN のオーバーレイ仮想レイヤ 2 ネットワークを提供する 1000 台を超える仮想スイッチ (Open vSwitch) の管理、10,000 を超える仮想ネットワークの生成管理、10,000 を超えるエンドホストを仮想ネットワークに接続管理、等の

要件を満たすコントローラとして実証に成功した [24]。本コントローラは NEC Biglobe の商用クラウドサービスで採用された [25]。当時の Nicira 社 (現 VMware 社) の NSX Virtualized Solution [26] は、ONIX ベースの分散型コントローラで VXLAN オーバレイによる仮想ネットワークを提供したが、単体コントローラが分散型の仕組みをとらなくても Web 3 層モデルによりシステムとして必要に応じてコントローラをスケールさせることにより、VXLAN オーバレイの仮想ネットワークを商用化できることを示した [24]。

(3) SDN コントローラ基盤方式 (第 3 世代)

第 3 世代はテレコムキャリア網への適用を想定し、データセンターや広域網を含めたヘテロな網の混在環境でのコントローラによる、統合的な制御に不可欠なネットワークの抽象化モデルを追求した時代である。従来の網の制御システムは、特定のレイヤに特化したモデルと制御方法を採用しているため、サービス要求の進化に合わせて最適な統合制御システムを開発することが困難だった点を課題として捉え、レイヤ非依存のネットワークモデル・抽象化とモデルベースでの新たな制御方法を考案し、マルチレイヤ、マルチドメインの連携制御を容易に実現できる統合制御基盤 OdenOS [27] を開発した。OdenOS では、さまざまなネットワークをトポロジーとフローに抽象化したモデルで表現し、物理ネットワークを抽象化したネットワークオブジェクトに対して操作 (Federator, Slicer, Aggregator, Linklayerizer などの制御オペレータ) を行うことで物理ネットワークを制御する方式を設計した。本方式をパケット・光トランスポート網のマルチレイヤ制御や、OpenFlow 網と VXLAN 網の連携制御へ適用した。本技術は、NTT、NTT コミュニケーションズ、NEC、富士通、日立の 5 者で共同で進めた広域ネットワークインフラの SDN 化を目指す世界初の研究開発プロジェクト O3 Project にて実証に成功した [28], [29]。

(4) SDN 向けネットワーク制御方式

前述した SDN コントローラ基盤上には、SDN 向け各種ネットワーク制御・監視アプリケーションが動作する。従来の自律分散型のプロトコルとは異なる多様なネットワーク制御・監視の研究開発を行ったうち三つを紹介する。

(a) 高速フローテーブル Verification 制御

SDN にてユーザを収容する仮想ネットワーク間でのトラフィックの完全な Isolation の保証、コントロー

ラ・スイッチのネットワーク状態が一時的に不安定になった際のネットワーク状態の完全性の確認等、ネットワーク Verification 制御に関する研究開発を実施した。ネットワーク内の各スイッチに設定されたフローテーブルの状態を確認して、Reachability の可否、Isolation の完全性、Loop が存在しない事の証明などネットワークの Configuration 状態を確認するのが主目的である。本領域は多くの研究機関が取り組んでおり、HSA, VeriFlow, NetPlumber など Graph-based approach, FlowChecker, Ant eater, Flower, DNA など Logic-based approach がある [30]。一般にネットワーク規模が大きくなると Verification には多大な時間がかかるが、上記の中で最も高速な Stanford の Header space analysis (HSA) 方式でも、フロー数を R とすると $O(R^3)$ の時間がかかる課題がある。NEC ではそれに対して HSA 方式を改善し、 $O(R^2)$ の時間に処理を高速化した Back-trace Header space analysis (B-HSA) 方式を提案。性能評価で有効性を示すとともに、方式の完全性を証明した [30]。

(b) マルチパス経路制御・QoS トラヒック制御

Fat-Tree 型のデータセンタネットワークでは、サーバ間 (East-West) のトラヒックが転送される経路は複数の経路が存在し、かつダイナミックに状況変化が起こるため、一般のマルチパスルーティングではその動的変化に追従できない。リンク利用率を考慮した k-shortest path 方式では計算量が大きくなるため、本課題に対して、計算量を抑えつつ動的なリンク利用率に高速に追従するマルチパス経路制御方式 [31] の研究開発を実施した。都市工学の行動理論で広く用いられている非集計ロジットモデルを援用した Multinomial Logit Based (MLB) ルーティング方式に着目し、フローベースルーティング方式へ拡張した。MLB はリンクコストを基に重み行列を算出し確率的にトラヒックを転送する仕組みで、重み行列の計算は一度の逆行列の演算のみで行える特徴から、計算量を大幅に低減でき、フロールーティングへの対応、End-to-End QoS への対応方式の具体化と有効性を実証した。

(c) ネットワーク統合監視・デバッグ開発環境

SDN コントローラの開発・運用時には、コントローラの内部状態、コントローラによって制御されるスイッチの状態、及びネットワークに接続された端末の状態を統合的かつ効率的に収集する機構が標準化されておらずデバッグ・障害解析が難しい。本課題に着目し、コントローラ開発・運用時に種々のエンティティ

の状態を統合的に収集・解析する機構の研究開発 [32] を実施した。従来は、ネットワークのログを収集しログを統合することで、運用者が装置の知識と想定イベントとの差を認識することで異常を検知するといった比較的静的な変化への対応は可能だったが、SDN のようにネットワークの状態が動的に変化し、各装置が多様なイベントを出す環境への対応は不可能であった。本問題に対して、種々のエンティティが発生する種々の状態・状態変化イベントを共通の方式でカプセル化し、それをイベント通知としてイベントコレクタに送信し、イベントコレクタ側で時刻に基づき整列して、単一インタフェースで多様なイベントを表示、デバッグを可能とするシステムを提案した。Trema 上に実装を行い、従来方式に比べて高速かつ低負荷で種々のエンティティからの状態・状態変化を収集・分析できることを実証した。

2.4 SDN ソリューション商用化

前述の Stanford での Open Innovation による Agile 研究開発、SDN の差異化領域の研究開発等の結果を活用し、2011 年にネットワーク構築・運用ニーズの強いデータセンタ・企業・公共向けに、OpenFlow/SDN をベースにしたネットワーク仮想化ソリューション ProgrammableFlow (スイッチ (PF5000 シリーズ) とコントローラ (PF6800 シリーズ) から構成) を世界で初めて製品化・商用提供開始した [33]。ネットワーク装置を抽象化して制御する ProgrammableFlow コントローラ (SDN コントローラの NEC 商用版) により、トポロジー可視化、セキュアなネットワーク分割 (仮想ネットワーク (Virtual Tenant Network) 機能)、通信経路の負荷分散 (マルチパス経路制御)、アプライアンス連携 (フローベースサービスチェイニング) 等をネットワークの集中制御により提供 [33] し、多くのユーザへの商用展開に成功した。

キャリアデータセンタでは、多拠点のデータセンタをもつ事業者が本製品を採用し、拠点間の動的なリソース配備により効率的運用をしている [34]。企業網では、部門ごとに構築・拡張した物理ネットワークを柔軟性とセキュリティを両立させながら共有化し、構築・運用コストを低減した [35]。公共網では、一つの物理ネットワークを仮想化により多様なシステムで共有し、広域ミッションクリティカルな環境下でネットワーク統合運用を実現した [36]。OpenFlow/SDN は、ネットワークの設計・構築・運用の概念を変え、大幅なコスト低減を作り出し、運用者の投資やサービス形

態へ大きな革新をもたらしている。

2.5 SDN 研究開発の発展的取り組み

これまで、上述の SDN コントローラ基盤方式、ネットワーク制御方式の研究開発は、Homogeneous な SDN 装置での実現としての研究開発とデータセンタ・企業・公共向けの商用化適用を主に進めていた。現在はこれらの成果を更に発展させ、テレコムキャリア網を対象とした Transport SDN の実現方式の研究開発中である。第 3 世代の SDN コントローラ基盤方式の研究開発で学んだマルチレイヤ、マルチドメインでのヘテロな網の混在環境での SDN 実現方式を発展させ、レイヤ非依存のネットワークモデル・抽象化の考え方を既存網の NMS/EMS システムと SDN 装置を混在させながら実現する方式に関して、ONF 標準化の枠組みである T-API (TAPI [37]) も取り込んだ形で実現方式を検討し、マルチベンダでの相互接続実験等を通して方式の拡張方式を検証中である。

3. NFV へ向けた取り組み

3.1 市場ニーズ・技術トレンド

従来、キャリアネットワーク内で用いられる各種アプリケーション装置 (BRAS, SBC, EPC, IMS, NAT, Firewall, DPI 等) は専用装置で運用されていた。しかしながら、昨今のスマートフォンによるトラフィックパターンの集中、あるいは震災などのネットワーク障害復旧の際の急激な負荷の集中の状況に対応するために、専用装置を最大負荷に併せて配備する必要があり過剰に設備を導入せざるを得ず、投資コストが大きくなる課題があった。また Long-Tail の多様なサービスを Small-Start で立ち上げ運用する際に、Small-Start で小さく初めつつ需要に応じてリソースを段階的に増やしていくといったニーズには対応できなかった。本課題を解決するためには、急激な負荷変動に動的に追従したり、Small-Start から需要に応じてリソースを追従させる等インフラの柔軟なリソース割り当てを実現するニーズが高まっている。

Network Function Virtualization (NFV) とは、従来専用装置によって提供されていたネットワーク機能・サービスを、クラウドコンピューティング及び仮想化技術を活用し、汎用サーバ・スイッチ等の汎用装置を用いて提供しようとする仕組みである。NFV は、ETSI-NFV において通信オペレータを中心に標準化が進められるとともに、OPNFV など OSS コミュニティにて POC による実証が進められている。NFV の

実用化により、通信オペレータは設備コストの低減による CAPEX の低減、IT 領域で適用されている自動化技術の導入による OPEX 低減、新規ネットワークサービスの迅速な開始、サービスに対する投資利益率の改善などを期待できる。

3.2 NFV 研究開発へ向けた取り組み

NFV は大きくネットワーク機能を提供する仮想化ソフトウェア Virtualized Network Function (VNF)、VNF の実行基盤となるプロセッシング、ストレージ、ネットワーク資源を提供する Network Function Virtualization Infrastructure (NFVI)、物理・論理資源のオーケストレーション (配備、設定、管理の自動化)、ライフサイクル管理を実行する NFV Management and Orchestration (MANO) とから構成され、クラウドサービスの要件と共通の Layer2/3 仮想ネットワーク提供、スケーラビリティ対応、オンデマンドプロビジョニングなどの要件に加えて、信頼性・可用性・性能の保証等のキャリアインフラの要件が加えられており、仮想化基盤でのキャリア要件をいかに実現するかが研究テーマとなる。

従来は、NFV 環境で多様なサービスを統合する際に、性能の見積もりやサイジングを行うための人手をかける設計が必要となり、多大な時間がかかる問題がある。そのため信頼性・可用性・性能の保証等のキャリアインフラ要件の設計の自動化へ向けた研究開発を実施した。大きく分類するとサービス性能推定・モデル化、論理物理網対応-最適配備設計、自動更改型プロビジョニングの研究の 3 領域を実施している。

(1) サービス性能推定・モデル化

(i) NFV サービスは複数の機能 (Firewall, Load balancer, web サーバ, AP サーバ, DB 等) を結合して構成されるが、それぞれの機能である VNF は一般にブラックボックスであり、VNF の動作予測への対応、(ii) NFV のサービス内の機能の入れ替え (リソース割当の変更, Scale Out/In 等による構成変更, WAN accelerator を追加するなど新機能の追加・削除, 帯域制御・経路制御の変更) による動的性能変化のモデル化とが研究課題である。NEC では、離散イベントモデルである Colored timed Petri Net (CTPN) ベースのシミュレーションモデルと VNF の統計的性能モデルとを組み合わせた新たな方式 [38] を提案している。本方式により、期待される SLA (遅延やスループット) を入力すると、NFV の性能見積もりを実行できるサービス性能設計・管理システムを開発し、性

能予測精度を極めて高く実現できるのを実証した。

(2) 論理物理網対応-最適配備設計

サービス性能推定・モデル化をベースに、ユーザのサービス定義（機能要件、性能品質要件、制約条件）に対して論理資源の見積もり（仮想マシン数、論理リンク数、論理トポロジー他）を行った後、実際に物理資源上への配置（余剰計算資源、余剰ネットワーク資源、物理トポロジー、SLA 制約確保）設計を行い、資源設定、運用を行う。その際、特に論理物理網対応-最適配備設計部分、具体的には複数の VNF を連結して一つのサービスを構成するサービスチェイニング向けのリソース配置最適化が研究課題である。コンピューティングとネットワークのリソースを一つのグラフとして統合モデル化し、整数線形計画問題に定式化。サービスチェーン全体でネットワーク性能保証や信頼性の向上をシミュレーションによる評価により実証した [39]。

(3) 更改型 Template ベースプロビジョニング

最後の資源設定、運用フェーズでは、効率的な構築・変更・運用作業が必要となる。既に稼働中の VNF 環境に新たに VNF を追加・削除したり、それに併せてチェイニングを変更する場合、現在実行中のサービスに影響を与えずに（無瞬断で）変更手順を実行する Consistent Update 作業は、手順の導出、間違えのない実行など従来完全な自動化は困難で、人を介した作業が必要だった。Configuration Management Tool (CMT) ツール群 (Puppet, Chef, Ansible, PowerShell DSC 等) や Template-based Provisioning (TBP) ツール群 (OpenStack heat, Oasis TOSCA 等) の構築自動化手法があるが、事前に定義される部品の再利用性が低く、実際の利用時には人手をかけた多くの定義作業が残る点が課題だった。CMT や TBP の利点を活かしつつより簡潔に所望のシステム構成の定義を可能とし、Consistent Update を実現する Configuration-Oriented Planning (COP) 方式の研究開発を実施した。性能評価により有効性を実証した [40]。

3.3 NFV 研究実証へ向けた取り組み

NFV と SDN の研究実証として、NTT ドコモ、NEC、富士通とで総務省の委託研究を活用し、大規模災害などで携帯電話の通話やデータ通信が大規模かつ集中して発生した場合に起こる、疎通率が低下する輻輳状態の解決へ向け取り組んだ。具体的には、モバイルコア網に対しサーバ仮想化技術を用いた NFV 化 (vEPC) と SDN 化 (モバイル優先制御) とを組み合わせ

せ、通信サービスの処理リソースを動的かつ迅速に変更可能であることを実証した [41]。

vEPC では輻輳を検知した場合、動画などのリッチコンテンツサービス用の処理リソースを削減し、音声やメールなどの基本サービスに優先的に処理リソースを割り当てる動的な変更制御と、それに向けた vEPC のリソース制御を実現した [41]。SDN では、大規模災害時には、音声通話やメールといった重要通信を優先処理したり、障害時に迂回経路を設定するためにトラヒックの優先度に応じた経路制御を行う必要があり、3GPP 標準では対応が不可能だった移動通信網内で基本サービスのフローを識別し、フローの優先度に応じた帯域の割り当てと経路制御を実証した [41]。本実証実験を仙台と東京の遠隔の 2 地点のインフラを用いて現実に近い形で行うことで、vEPC の商用化が加速され、ETSI-NFV の標準化の場でも本実証が先端的な例として高く評価された。

3.4 NFV ソリューション商用化

前述の NFV の研究開発・実証の取り組み成果を活用し、2014 年にはネットワーク機能仮想化 NFV 関連の製品、VNF としてのモバイルコア製品 vEPC [42]、固定アクセス仮想化製品 vCPE、運用管理システム MANO [43] の商用化に成功した。サーバ上の制御ソフトでキャリアグレードのネットワーク機能・品質の提供、並びに高性能化の提供を実現し、国内外のキャリアとトライアル実証に成功した。基幹ネットワークへの SDN/NFV コンセプト適用の潮流を醸成し、SDN/NFV は安心安全・効率化を支える社会インフラ基盤の実現技術として広く受け入れられている。

3.5 NFV 研究開発の発展的取り組み

これまで、上述の 3 領域の NFV 研究開発は、地理的に同一局舎内・同一データセンタ内での性能推定、設計、運用管理システムとしての有効性を実証してきた。現在はこれらの成果を更に発展させ、サービスとして Internet-of-Things (IoT) のような多様なアプリケーションが動作する環境を想定し、NFV の処理系が広域・地理的に分散された Edge-Cloud Computing の形態をとる環境での性能推定、設計、運用管理システムとしての実現方式を研究開発中である。取り扱うデータの特성에依りて処理能力が変わる IoT のモデル化の困難性と、広域・地理的分散での NFV のモデル化と動作予測、地理的分散下での対障害時の信頼性の担保を行う設計、運用方式といった観点で、研究開発を鋭意推進中である。

4. SDN/NFV の今後の展望

4.1 SDN/NFV の世界の最新技術潮流

SDN/NFV 領域での世界の最新技術潮流を分析すると、以下の四つの技術のトレンドが見て取れる。

(1) Disaggregation / Re-Aggregation による Open API の進化

主にキャンパス、データセンタでのイノベーションでスタートした SDN の本質は装置の Disaggregation であり、C-plane - D-plane の分離から業界のパラダイム変化を促進した。この流れはキャリア領域の無線基地局、モバイルバックホール・フロントホール、モバイルコア、固定アクセス、光パケットトランスポート等全領域に波及し、Disaggregation をベースに低コスト化、運用容易化、マルチベンダ対応容易化へ向けた動きが加速中である。

また SDN の初期段階は Disaggregation することで Open API が OpenFlow として規定されたが、現在の市場ではこの API の Boundary が変化し、より使いやすく市場を広げる形で進化しつつあることも見逃せない。例えば Facebook のデータセンタで活用され、OCP で標準化されている Wedge シリーズスイッチは、ソフトウェア層に FBOSS Agent を置き、その配下に OpenFlow ベースの SDK である OF-DPA [15] に加え、一般のネットワークインタフェースの SDK である OpenNSL [14] 等が配備され、複数の API を統合したより制御可能な範囲を拡張したモデルが展開された。ある意味、Diaggregation したあとに新たな Re-Aggregation による API の最適化が行われているとみなせる。つまり、SDN 制御というコンセプトは維持しつつも、業界として Disaggregation - Re-Aggregation を繰り返すことにより徐々に今後方式が洗練されていく流れが汲み取れる。キャリア領域での Disaggregation の流れは始まったばかりだが、同様の動きが予想され今後の動向は注視すべきである。

(2) Domain Specific Language (DSL) による SDN 向け Policy 記述手法の進化

SDN の運用の簡便さを求める技術トレンドである。SDN によりネットワーク状態を収集・観測・制御が容易になったが、複雑な Policy による制御を行ったり、きめ細かな制御をやろうとすると、SDN Controller 上の制御プログラムの Configuration 量が増加する課題がある。当然ながら Configuration 量が増えると設定ミスをおかすリスクも増える。本課題に対して、Cor-

nell 大学では、Frenetic [44]、NetKAT らの研究を進め、それを更に進化させた形で Princeton 大学では、SDX [45]、Propane、SNAP [46] など DSL の Policy Language の研究を進めている。Frenetic、NetKAT は SDN の Policy Language として SDN の初期段階に提案され、その後 SDX は Internet Exchange での BGP ポリシーへの適用、Propane はマルチドメインの BGP ポリシーへの適用、SNAP は SDN において Dataplane を Stateful に管理するきめ細かな制御への適用を行う。いずれの方式も、ネットワーク全体を一つの Fabric とみなして上位レベルの運用 Policy を短いコード量で記述すると、Compiler がネットワークグラフにおける経路割当、ステート割当、ステート依存性解析、Policy の完全性の解析をした上でデバイスへの Configuration を矛盾なく行う。今後はこのような DSL の進展により、SDN コントローラ上の制御プログラムを簡便にミスなく記述が可能となる動きが加速することが期待される。

(3) Data Plane の Programmability の進化

SDN の Dataplane 制御はデータパケットのヘッダごとに規定された動作を定義する仕様 (Match-Action) となっているため、新規のプロトコルが追加されると新たに定義を追加・標準化する必要があり、柔軟性に課題があった。Stanford 大学・Barefoot Networks らが提唱している P4 [47] 言語では、(i) Reconfigurability: Dataplane のヘッダの Parsing と処理を再定義可能、(ii) Protocol independence: プロトコルに依存しない SDN 動作 (Match-Action) を定義可能、(iii) Target independence: Compiler がハードウェア特性を吸収し、ハードウェアには非依存でパケット処理機能を定義可能、とを提供している。ある意味、Dataplane というハードウェア、データに近いレイヤにおける DSL により、プロトコル依存性をなくす SDN を実現する方式である。P4 Consortium [48] も立ち上がり、P4 Developer 向けの様々な Hands-on Training がなされ、かつ P4 に対応した商用版高速 Programmable ASIC が Barefoot Networks 社から近々提供開始となっている状況である。本チップの登場により、従来困難であった Application-specific なカスタマイズをユーザが自由に定義したネットワーク制御が可能となる世界が加速されると予想される。

(4) NFV 処理系の Open 化

現在の NFV 市場の製品は第一世代製品で、ハードウェアプライアンスをソフトウェア化、仮想化した段

階で、旧来型のハードウェアを置換するレベルが多い。そのため、NFVの制御インターフェースがベンダごとにばらばらでマルチベンダでのNFVの統合管理が困難で、ベンダごとのNFV管理システムを置いた複雑な管理システムになりがちである。また、一般にはNFVはNFVでの管理形態、SDNはSDNでの管理形態で分離しているため、NFVのリソース管理のみNFVが行い、NFV間での通信の最適化のみSDNが行うといった処理の統合レベルに留まっていることが多い。本来は、異なるNFV間の接続、過負荷の検出、負荷分散、動的スケールリング (Scale-in/out)、冗長制御などNFVのプロセッシングに注目したベース部分は共通的な枠組みで取り扱え、特殊な機能に特化した処理部分をVNFとして実装といった枠組みが望ましい。これは、SDNとNFV融合型のProcessing Frameworkと捉えられ、VNFの新たなDisaggregationによるプロセッシングと制御の分離・共通化の方向性である。本手法に関する研究が、OpenNF [49]、E2 [50]等 で実施され、試作評価で有効性が確認されている。今後はNFV処理系のOpen化と共通フレームワーク化の世界への移行が期待される。

4.2 SDN/NFV 研究開発の発展的取り組み

上記のようなSDN/NFVの世界の最新技術潮流も捉えつつ、著者らは、交通や社会インフラなど様々な社会ソリューションプラットフォームの実現に向け、SDN/NFVの活用を進めている。ここでは、交通向けにはエッジサーバでの折り返しによる低レイテンシ処理を目指し、社会インフラ向けにはデバイス-エッジクラウドまで一貫して広域にセキュリティを担保を目指すなど、様々な要件にあわせてNFVの処理系が広域・地理的に分散されたEdge-Cloud Computingの形態をとる。図1は、提案するアーキテクチャを示す。5G無線アクセスからトランスポートネットワーク、クラウド上のコンピューティング資源まで統一された物理インフラ上に、様々なサービス要件をもつネットワーク機能とエッジ処理機能を複数構成する。これにより、サービス要求条件に応じた最適な仮想インフラを自由に構築し、迅速な新サービス開通を実現する。本領域においては、著者らは下記のような技術を含めてSDN/NFV研究開発の発展的な取り組みを進めている。

(1) NFVの処理系が広域分散した環境でのEnd-to-endサービスオーケストレーションに向けて、DSLによるSDN向けPolicy記述手法、NFV処理系の

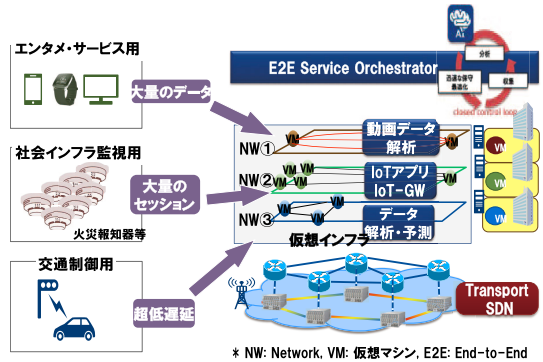


図1 SDN/NFVを活用した広域・地理的分散 Edge Cloud Computing

Fig.1 Geographically distributed wide-area edge cloud computing based on SDN/NFV.

Open化の適応型の研究開発。IoT業界では Sensing / Data Analytics / Dynamic Actuation というデータの特性に応じたアプリケーション配備問題の観点、SDN/NFV業界ではネットワーク・ITインフラ (IaaS) の動的設定という観点でそれぞれ研究開発が進められているが、我々は両者を統合した最適化を図ることで、IoT+SDN/NFVにおける Platform as a Service (PaaS) モデルでの課題を抽出し研究開発を進めている。

(2) 小型センサーなどのM2Mデバイスから発生する大量のセッションを収容するための制御信号削減技術の研究開発。M2Mデバイスの通信頻度に応じてネットワークとの接続時間を動的に変更し、発信時の制御信号量を1/10に削減を可能とする。

(3) 超低遅延化を実現するためのモバイルネットワーク基地局スケジューリング技術の研究開発。エッジサーバにおいてトラフィックの分析を行いアプリケーションの動作状況を推測し、これに基づいて無線基地局スケジューラの制御パラメータを動的に最適化する。これにより、自動車と歩行者が混在する環境下において、衝突回避の通信要件である100msec以内の通信成功率を従来の約50%から95%まで向上を可能とする。

5. むすび

筆者らは、SDNコンセプトを提唱し、中核となるSDN/OpenFlow + NFV研究開発を推進し、研究開発した技術を世界に先駆けて商用化し、論理集中型プログラマブル制御方式によるSoftware-Definedなシステム構築を実現した。これによりネットワークシステム概念を変革し、運用者の投資やサービス形態、利用形態に大きな革新を与える業界の進化に貢献したと感じ

ている。現在でも SDN/NFV は業界として進化を継続しており、今後は Disaggregation/Re-Aggregation による Open API の進化、DSL による SDN 向け Policy 記述手法の進化、Data Plane の Programmability の進化、NFV 処理系の Open 化が有機的に結合されていく世界となっていくことが期待される。また、それらプラットフォームを活用するユーザ視点、アプリケーション視点に立つと、昨今の IoT, AI を取り込んだクラウド型、フォグ型での多様な社会ソリューションは、まさに SDN/NFV 型のプロセッシングとネットワークワーキングが広域に分散された環境で構築されると予想される。これにより、新たなアプリケーションのイノベーションをクラウド、フォグの位置には依存せず自由に開発、構築、運用ができる世界へ向かっていくと期待できる。この世界の実現には、SDN/NFV の真の融合が必須になると確信しており、本領域への業界としての取り組みがますます重要となると考える。

文 献

- [1] A. Singh, J. Ong, A. Agarwal, G. Anderson, A. Armistead, R. Bannon, S. Boving, G. Desai, B. Felderman, P. Germano, A. Kanagala, J. Provost, J. Simmons, E. Tanda, J. Wanderer, U. Hölzle, S. Stuart, and A. Vahdat, "Jupiter Rising: A decade of clos topologies and centralized control in Google's datacenter network," SIGCOMM, pp.183–197, Aug. 2015.
- [2] S. Jain, A. Kumar, S. Mandal, J. Ong, L. Poutievski, A. Singh, S. Venkata, J. Wanderer, J. Zhou, M. Zhu, J. Zolla, U. Hölzle, S. Stuart, and A. Vahdat, "B4: Experience with a globally-deployed software defined WAN," SIGCOMM, pp.3–14, Aug. 2013.
- [3] <http://www.opencompute.org/>
- [4] <https://telecominfraproject.com/>
- [5] <https://www.opennetworking.org/>
- [6] <http://www.etsi.org/technologies-clusters/technologies/nfv>
- [7] <http://opencord.org/>
- [8] <https://www.opendaylight.org/>
- [9] <https://www.opnfv.org/>
- [10] <http://onosproject.org/>
- [11] N. McKeown, T. Anderson, H. Balakrishnan, G. Parulkar, L. Peterson, J. Rexford, S. Shenker, and J. Turner, "OpenFlow: Enabling innovation in campus networks," SIGCOMM CCR, vol.38, no.2, pp.69–74, April 2008.
- [12] <https://www.opennetworking.org/sdn-resources/technical-library#tech-spec>
- [13] <https://www.geni.net/>
- [14] <https://www.broadcom.com/products/ethernet-communication-and-switching/switching/openns>
- [15] <https://www.broadcom.com/products/Switching/Software-Defined-Networking-Solutions/OF-DPA-Software>
- [16] M. Casado, M.J. Freedman, J. Pettit, J. Luo, N. McKeown, and S. Shenker, "Ethane: Taking control of the enterprise," SIGCOMM, pp.1–12, Oct. 2007.
- [17] B. Pfaff, B. Heller, D. Talayco, D. Erickson, G. Gibb, G. Appenzeller, J. Tourrilhes, J. Pettit, K.K. Yap, M. Casado, M. Kobayashi, N. McKeown, P. Balland, R. Price, R. Sherwood, and Y. Yiakoumis, "OpenFlow switch specification version 1.0.0," <http://archive.openflow.org/documents/openflow-spec-v1.0.0.pdf>, Dec. 2009.
- [18] M. Kobayashi, S. Seetharaman, G. Parulkar, G. Appenzeller, J. Little, J. van Reijendam, P. Weissmann, and N. McKeown, "Maturing of OpenFlow and software-defined networking through deployments," Comput. Netw., vol.61, pp.151–175, March 2014.
- [19] <http://www.noxrepo.org>
- [20] H. Shimonishi, S. Ishii, L. Sun, and Y. Kanaumi, "Architecture implementation, and experiments of programmable network using OpenFlow," IEICE Trans. Commun., vol.E94-B, no.10, pp.2715–2722, Oct. 2011.
- [21] R. Sherwood, G. Gibb, and M. Kobayashi, "Carving research slices out of your production networks with OpenFlow," SIGCOMM CCR, vol.40, no.1, pp.129–130, Jan. 2010.
- [22] T. Koponen, M. Casado, N. Gude, J. Stribling, L. Poutievski, M. Zhu, R. Ramanathan, Y. Iwata, H. Inoue, T. Hama, and S. Shenker, "Onix: A distributed control platform for large-scale production networks," OSDI, pp.351–364, Oct. 2010.
- [23] <https://github.com/trema/trema>
- [24] 千葉靖伸, 須堯一志, "大規模 SDN ネットワークを実現する OpenFlow コントローラアーキテクチャ," NEC 技報, vol.66, no.2, pp.40–43, Feb. 2014.
- [25] <http://www.biglobe.co.jp/en/press/2013/0422.html>
- [26] <http://www.vmware.com/products/nsx.html>
- [27] Y. Izawa and K. Suzuki, "Multi-layer and multi-domain network orchestration and provision of virtual views to users," COMPSAC, vol.2, pp.20–25, June 2016.
- [28] <http://www.ntt.com/content/dam/nttcom/hq/jp/about-us/press-releases/pdf/2016/20160310.pdf>
- [29] A. Iwata, "Innovation for network businesses by the world's first SDN WAN technologies - O3 project," SDN/MPLS, http://www.isocore.com/sdn-mpls/technical_sessions.htm, Nov. 2014.
- [30] T. Tonouchi, S. Yamazaki, Y. Yakuwa, and N. Tomizawa, "A fast method of verifying network routing with back-trace header space analysis," IM, pp.225–233, May 2015.
- [31] 篠原悠介, 逸身勇人, 千葉靖伸, 下西英之, "マルチパス

- ルーティングにおける通信品質を考慮に入れた経路選択方法,” 信学技報, CQ2012-6, April 2012.
- [32] Y. Chiba and H. Shimonishi, “Network debugger: A unified tool for diagnosing network controlling applications,” WTC2012, March 2012.
- [33] <http://jpn.nec.com/univerge/pflow/>
- [34] <http://cloud.watch.impress.co.jp/docs/news/539491.html>
- [35] http://jpn.nec.com/press/201402/20140224_01.html
- [36] http://jpn.nec.com/press/201405/20140507_02.html
- [37] “Functional Requirements for Transport API,” ONF TR-527, June 2016.
- [38] S. Koizumi and M. Fujiwaka, “SDN + cloud integrated control with statistical analysis and discrete event simulation,” ICOIN, pp.289–294, Jan. 2015.
- [39] 逸身勇人, 篠原悠介, 千葉靖伸, 下西英之, “サービスチェイニングに向けたリソース最適配置手法の提案,” 信学技報, NS2013-245, March 2014.
- [40] M. Nakanoya, T. Kuroda, and A. Kitano, “Automated change planning for differential update IT systems with state constraint,” EDOCW, pp.1–9, Sept. 2016.
- [41] T. Shimizu, T. Nakamura, S. Iwashina, W. Takita, A. Iwata, M. Kiuchi, Y. Kubota, and M. Ohhashi, “An experimental evaluation for dynamic virtualized networking resource control over an evolved packet core network,” IEEE R10-HTC, TS10, 2013.
- [42] 塚越 努, 佐伯修一, 寺田純平, 砂川佑太, 島本裕志, 井口智仁, “vEPC におけるユーザプレーン制御の実現,” NEC 技報, vol.68, no.3, pp.23–26, March 2016.
- [43] 新井智也, 極楽寺淳一, 大平麻代, “ネットワークのインテリジェントな運用管理を実現する MANO 技術,” NEC 技報, vol.68, no.3, pp.19–22, March 2016.
- [44] N. Foster, R. Harrison, M. Freedman, C. Monsanto, J. Rexford, A. Story, and D. Walker, “Frenetic: A network programming language,” ICFP, pp.279–291, Sept. 2011.
- [45] A. Gupta, L. Vanbever, M. Shahbaz, S. Donovan, B. Schlinker, N. Feamster, J. Rexford, S. Shenker, R. Clark, and E. Katz-Basset, “SDX: A software defined internet exchange,” SIGCOMM, pp.551–562, Aug. 2014.
- [46] M. Arashloo, Y. Koral, M. Greenberg, J. Rexford, and D. Walker, “SNAP: Stateful network-wide abstractions for packet processing,” SIGCOMM, 29043, Aug. 2016.
- [47] P. Bosshart, D. Daly, G. Gibb, M. Izzard, N. McKeown, J. Rexford, C. Schlesinger, D. Talayco, A. Vahdat, G. Varghese, and D. Walker, “P4: Programming protocol-independent packet processors,” SIGCOMM, pp.88–95, Aug. 2016.
- [48] <http://p4.org/>
- [49] A. Gember-Jacobson, R. Viswanathan, C. Prakash, R. Grandl, J. Khalid, S. Das, and A. Akella, “OpenNF: Enabling innovation in network function

control,” SIGCOMM, pp.163–174, Aug. 2014.

- [50] S. Palkar, C. Lan, S. Han, K. Jang, A. Panda, S. Ratnasamy, L. Rizzo, and S. Shenker, “E2: A framework for NFV applications,” SOSP, pp.121–136, Oct. 2015.

(平成 28 年 12 月 13 日受付, 29 年 3 月 15 日再受付, 6 月 7 日早期公開)



岩田 淳 (正員)

1988 年東京大学工学部電気工学科卒業, 1990 年同大学大学院修士課程修了。同年 NEC 入社。以来, ATM, IP, MPLS, コンテンツ配信, 広域イーサネット, メトロ基幹, SDN/NFV, IoT/Edge Computing におけるシステム開発に従事。1997 年 University of California, Los Angeles 客員研究員。2001 年東京大学工学部電気工科学博士受領。2009 年 NEC IP ネットワーク事業部にて SDN 製品開発に従事。2011 年以後クラウドシステム研究所, 情報・ナレッジ研究所にて SDN/NFV 研究開発統括・所長代理業務に従事。2014 年以後 NEC Corporation of America にて SDN/NFV/IoT/Edge Computing 領域の研究開発・ビジネス開発にて Director 業務従事。2012 年フジサンケイビジネスアイ第 26 回先端技術大賞産経新聞社賞受賞, 2013 年科学技術と経済の会第一回技術経営・イノベーション賞受賞。2014 年電子情報通信学会業績賞受賞。著書「広域イーサネット技術概論」(電子情報通信学会) 2005 年, 「マスタリング TCP/IP OpenFlow 編」(オーム社) 2013 年。



下西 英之 (正員)

1996 年大阪大学大学院博士前期課程修了, 同年 NEC 入社, 現在同社システムプラットフォーム研究所部長業務に従事。2002 年大阪大学大学院博士後期課程修了。2003 年 UCLA computer science department 客員研究員。同社システムプラットフォーム研究所にて Software-Defined Networking (SDN), OpenFlow, Network Functions Virtualization (NFV), モバイルコアネットワーク及び IoT 関連のネットワーク技術の研究開発に従事。2012 年フジサンケイビジネスアイ第 26 回先端技術大賞産経新聞社賞受賞, 2013 年科学技術と経済の会第一回技術経営・イノベーション賞受賞, 2014 年電子情報通信学会業績賞受賞。



小林 正好

1995年京都大学工学部数理工学科卒業、1997年京都大学大学院工学研究科応用システム科学専攻修了。同年 NEC 入社。2003-2004年、及び、2007-2012年まで米 Stanford 大学にて客員研究員。2012-2014年まで米 Open Networking Laboratory に駐在。Stanford 大学にて OpenFlow/SDN の黎明期から研究開発に携わり、実証実験、標準化、SDN を活用した無線ネットワーク制御、SDN コントローラの研究開発に従事。現在、NEC 技術イノベーション戦略本部シニアエキスパート業務に従事。2013年科学技術と経済の会第一回技術経営・イノベーション賞受賞。2014年電子情報通信学会業績賞受賞。2016年第64回電気科学技術奨励賞受賞。