

# コアペリフェリ構成を用いた仮想化ネットワーク機能配置の 効率性および安定性の評価

津久井佑樹<sup>†</sup> 荒川 伸一<sup>†</sup> 村田 正幸<sup>†</sup>

<sup>†</sup> 大阪大学 大学院情報科学研究科 〒565-0871 大阪府吹田市山田丘 1-5

E-mail: †{y-tsukui,arakawa,murata}@ist.osaka-u.ac.jp

あらまし ネットワーク機能提供の柔軟性を高めつつ提供コストを削減することを目的として、ネットワーク仮想化 (NFV: Network Functions Virtualization) の導入が進められている。しかし、NFV の運用にあたってはサービスチェーン要求の変化に対して利用者品質の向上や運用コストを削減するための適切な仮想ネットワーク機能 (VNF: Virtual Network Function) の配置を定める VNF 配置問題が残される。本稿ではこの問題へのアプローチとしてコアペリフェリモデルに着目し、コアとなる機能を抽出し各ノードへと配置することによる VNF 配置の効率性と安定性を評価する。サービスチェーン生成モデルを用いた評価と AS トポロジーデータを参考にしたサービスチェーンを用いた評価の結果、コアサイズを 128 (Functions) とした場合、約 24% のサービスチェーン要求が処理可能であることがわかった。また、要求される VNF の偏在性が高い場合はコアの効率性と安定性が高まることがわかった。

キーワード ネットワーク仮想化 (NFV)、仮想ネットワーク機能 (VNF)、コアペリフェリモデル、サービスチェーン、有向グラフ、DC グラフ

## Evaluation of efficiency and stability of VNF placements by extracting a core-peripheral structure

Yuki TSUKUI<sup>†</sup>, Shin'ichi ARAKAWA<sup>†</sup>, and Masayuki MURATA<sup>†</sup>

<sup>†</sup> Graduate School of Information Science and Technology, Osaka University

1-5 Yamadaoka, Suita, Osaka 565-0871 Japan

E-mail: †{y-tsukui,arakawa,murata}@ist.osaka-u.ac.jp

**Abstract** Network function virtualization (NFV) has begun to be introduced with the purpose of increasing the flexibility of providing network functions and reducing the cost of providing them. One of the problems to be solved in NFV is the problem of placing a virtual network function (VNF: Virtual Network Function). In this paper, focusing on a core-periphery structure as an approach to this problem, We evaluate the efficiency and stability when the extracted core function is placed. By the evaluations using service chain based on AS topology data and evaluations using generative model of service chain, about 24% service chain request can be processed when the core size is 128 (Functions). We also found that the efficiency and stability of the core increases as the request rate for VNF is less diverse.

**Key words** Network Function Virtualization (NFV), Virtual Network Function (VNF), Core Periphery model, Service chain, Directed graph, DC graph

### 1. ま え が き

スマートフォンやタブレットの普及を背景に、情報ネットワークと情報ネットワークを利用するアプリケーションやサービスは人々の生活に浸透している [1]。利用者は従来の料金設定で従来通り、あるいはより良いサービスを求め、さらにはサー

ビス利用に伴って通信量が増大する。通信事業者は、通信量の増大に対処しつつ、様々な利用者が求める様々なサービスを効率的に提供することが求められている。利用者にサービスを提供するにあたって、通信事業者は、例えばファイアウォール機能などのネットワーク機能を提供する。一般に、ネットワーク機能は専用ハードウェアを用いて提供されるため、ユーザの需

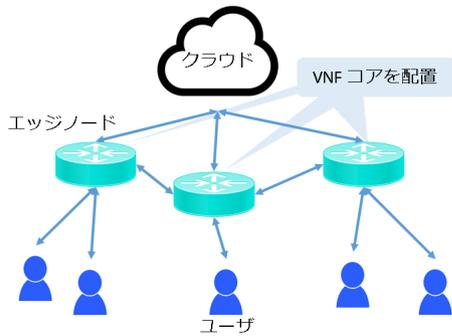


図 1 VNF 配置の例

要の変化に合わせたネットワーク機能の処理能力の調整が困難となる。また、専用ハードウェアの導入コストや運用コストも増大するため、コスト効率性を高めることが課題となる。この問題の解決策として、ネットワーク仮想化 (NFV: Network Functions Virtualization) が注目されている [2,3]。NFV ではこれまで専用のハードウェアで実現されていたネットワーク機能を分離し、汎用サーバ上で動作する仮想ネットワーク機能 (VNF: Virtual Network Function) として実現する。VNF は一つの汎用ハードウェア上で複数のインスタンスを動作させることが可能であるため、通信事業者が抱えるハードウェア数を減らし、設備投資や運用にかかるコストを軽減することが期待できる。また、利用者が所望するネットワーク機能群の提供は、VNF を鎖状に接続したサービスチェーン要求として通信事業者が受け入れ、これを通信事業者が有する汎用ハードウェア上で実行処理することにより行われる。VNF は仮想化されるため、要求されるネットワーク機能の変化に応じて実行される汎用ハードウェアを変更したり、新たな機能を提供する VNF を作成し展開することが可能となる。これにより、ユーザの需要の変化に合わせてネットワーク機能の処理能力を調整可能とする柔軟性が提供可能となる。

NFV によって、ネットワーク機能提供のコスト効率性と柔軟性を高めることが可能となるが、運用にあたっては解決すべき課題がいくつかある。その一つに VNF の配置問題がある [4]。VNF 配置問題は、汎用ハードウェアの数量が与えられるものとして、サービスチェーン要求の変化に対して利用者品質の向上や運用コストを削減するための適切な VNF の配置を求めめる問題である。文献 [5] では MIP (Mixed Integer Programming) を用いて、配置コストの最小化と負荷の分散の観点で最適な VNF 配置手法を示している。しかし、この手法では MIP によって VNF 配置を求めめるため、計算に膨大な時間を必要とする。そのため、サービスチェーン要求の変化に応じて VNF の再配置を行う場合、それに伴う計算時間が問題となる。さらに、ヒューリスティック手法等によって計算時間の問題が解決できると仮定しても、これらの手法によって常に最良の VNF 配置を求めめる結果として、VNF 配置場所の変更が多数必要となる懸念がある。多数の配置変更を必要とする場合、配置変更の過程で生じるネットワーク機能提供の継続性が損なわれる恐れがある。また、変更後の VNF の配置において、一定の性能品質を

確保するためには、すべての VNF において実行品質を担保しなければならない。近年は、ネットワーク機能の仮想化だけではなく、サービスアプリケーションが利用する機能の仮想化についても検討がなされており、すべての VNF に対して実行品質を担保することは困難となる。そこで、効率性を高めつつも適応性を有する VNF 配置を得ることを目的として、コアペリフェリモデル [6] に着目する。コアペリフェリモデルは、システムの環境変化に対する適応性と効率性を説明するモデルの一つであり、システム要素をコアとペリフェリに分類する。コアのシステム要素は効率的かつ安定的に動作し、一方、ペリフェリのシステム要素は環境変化に応じた変化を許容している。これにより、コアペリフェリモデルにもとづくシステム構成では、効率性と柔軟性が両立される [6]。このコアペリフェリモデルを用いて VNF 機能を解釈し、さらには設計に応用することで、効率性と柔軟性を両立する VNF 配置の実現が期待できる。すなわち、VNF 全体の実行効率を高めるのではなく、コアに分類される VNF の実行効率を高めることによってシステムの効率性を高めつつ、機能要求の多様化についてはペリフェリに分類される VNF のみを組み替えることによって VNF 再配置時の変更量の抑制しつつ対処することが期待される。

NFV において、効率的かつ安定的に動作させたい VNF は、頻繁に実行される VNF である。実際に、文献 [7] のシミュレーション結果では、サービスチェーンの要求量の変化に迅速に対応するためには、共通して実行される VNF 集合を保持することが望ましいことが示されている。したがって、頻繁に実行される VNF をコアと解釈し、コアではない VNF をペリフェリと解釈すれば良い。しかし、このような解釈を NFV における VNF 配置問題に応用するには、いくつかの課題がある。一つは、コアサイズに対する動作の効率性である。例えば図 1. のように、ある特定の VNF 集合をコアに分類し、エッジノードに配置するものとする。すべての VNF をコアとしエッジに配置すれば、エッジノードですべてのユーザリクエストが処理できる。しかし、エッジノードのリソース量はコスト観点から有限かつ最小としなければならない、すべての VNF をコアとすることは困難であり、一部の VNF をコアとせざるを得ない。その際に、何種類の VNF をコアとすれば、何割のサービスチェーン要求 (の一部) を処理することが可能となるのかを定量的に明らかにしておく必要がある。もう一つの課題は、コア/ペリフェリの分類の時間軸方向での安定性である。ある時刻のサービスチェーン要求パターンにもとづいて VNF をコアとペリフェリに分類したとしても、サービスチェーン要求が変化の中で、その分類が長期にわたって有効となるとは限らない。

本稿では、NFV における VNF 配置問題を前提とし、コアペリフェリモデルの適用にもとづく VNF の分類と、その分類により得られる効率性と安定性を定量的に明らかにする。ただし、NFV はまだ標準化を策定する段階であり、実測にもとづくサービスチェーン要求パターンを用いた定量的評価を行うことはできない。そこで、まず経年的変化の実測データがある自律システム (AS: Autonomous System) のネットワークを対象とし、コアペリフェリモデルの適用によるコアの効率性および安

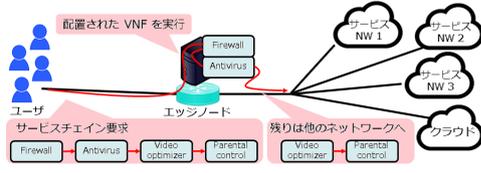


図 2 サービスチェーン要求の例

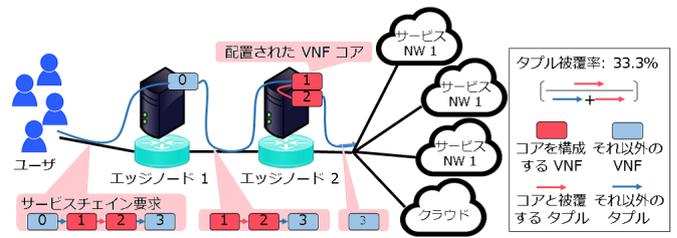


図 3 タプル被覆率の例

定性を評価する。次に、生成モデルにもとづくサービスチェーン要求に対してコアペリフェリモデルにより VNF を分類し、コアの効率性および安定性の知見が有効であることを確認しつつ、定量的な評価を行う。

## 2. 有向グラフを対象としたコア抽出方法

### 2.1 サービスチェーン

NFV ではユーザからのリクエストを、VNF を連鎖させたサービスチェーンとして扱う。ユーザにサービスを提供する際には、図 2.1 のようにサービスチェーンの順序にしたがって VNF を実行する。サービスチェーンにはトラフィックフローの自動化や、実行される VNF の組み換えの柔軟性が高いといった利点がある。

### 2.2 サービスチェーン要求の有向グラフ化

サービスチェーンにおいて、効率性の観点より相互に頻繁に呼び出しあっている VNF の集合をコアとして認識することができる。そこで、ここではサービスチェーンを有向グラフ化し、グラフノードの次数に着目して相互に呼び出しあうコアの抽出を行う。

### 2.3 サブグラフ分解にもとづく VNF コアの抽出

サービスチェーンからコア抽出を行うために、文献 [8] で提案されたマトリクスを用いる。文献 [8] で提案されたマトリクスは、ノードのハブ/オーソリティ性のみではなくノード間の有向辺についても着目することで、コラボレーティブな特徴を有するサブグラフ  $\mathbf{DC}_{k,l}$  を抽出する。マトリクスの定義を以下に示す。

まず、有向グラフを  $D = (V, E)$  とおく。  $V$  は  $D$  の頂点集合であり、  $E$  は  $D$  の辺集合である。非負整数  $k, l$  について  $\delta^{out}(F) \geq k$ ,  $\delta^{in}(F) \geq l$  を満たす  $D$  の最大のサブグラフ  $F$  を  $\mathbf{DC}_{k,l}(D)$  と定義する。ここで、  $\delta^{in}(F)$  は  $F$  のノードの最低入次数、  $\delta^{out}(F)$  は最低出次数である。なお、  $\mathbf{DC}_{0,0}(D) = D$  であることに注意する。

## 3. コアの評価指標

文献 [8] のマトリクスを用いることで、有向グラフ化したサービスチェーンを次数にもとづいてサブグラフに分解することが可能である。本章では得られたサブグラフをコアとしたとき、その効率性と安定性を評価するための指標を定義する。

### 3.1 タプル被覆率

サービスチェーン要求において、隣接 VNF の組み合わせをタプルと呼称する。ここで、図 3 のように、全てのサービスチェーン要求に含まれるタプルのうち、あるグラフを構成する有向辺と被覆するものの割合を、そのグラフのタプル被覆率とする。

タプル被覆率は、そのグラフをコアとしたときに、コアだけで処理可能なサービスチェーン要求の割合を反映しており、タプル被覆率が  $x\%$  であるならば  $x\%$  のサービスチェーン要求を処理できるとみなす。NFV におけるコアの効率性は、コアを物理ネットワークに配置した場合に、その部分だけで確率的にサービスチェーン要求の何 % を処理することができるかで評価する。また、時間経過に伴いサービスチェーン要求に含まれるタプルは追加・削除され変化していく。コアがもつ安定性はサービスチェーン要求が変化した場合に、コアで処理できるサービスチェーン要求の割合がどのように変化するかで評価する。

#### 3.1.1 タプル被覆率の定義

まず、ある時刻  $t$  においてサービスチェーンの有向グラフを分解して得られたサブグラフを  $\mathbf{DC}_{k,l}^t$  とする。また、  $\mathbf{DC}_{k,l}^t$  を構成する有向辺の集合を  $E(\mathbf{DC}_{k,l}^t)$  で表す。一方で、ある時刻  $t$  のサービスチェーンに含まれるタプルの集合を  $L(t)$  とする。なお、  $L(t)$  は重複するタプルを含む。

ここで、時刻  $t_1$  におけるサブグラフ  $\mathbf{DC}_{k,l}^{t_1}$ 、時刻  $t_2$  におけるタプル集合  $L(t_2)$  について、  $L(t_2)$  のうち  $E(\mathbf{DC}_{k,l}^{t_1})$  に含まれるタプルの集合  $LE_{k,l}(t_1, t_2)$  を式 (1) で定義する。

$$LE_{k,l}(t_1, t_2) = \{x | x \in L(t_2), x \in E(\mathbf{DC}_{k,l}^{t_1})\} \quad (1)$$

このとき、時刻  $t_2$  における  $\mathbf{DC}_{k,l}^{t_1}$  のタプル被覆率  $C_{k,l}(t_1, t_2)$  を式 (2) で定義する。

$$C_{k,l}(t_1, t_2) = \frac{|LE_{k,l}(t_1, t_2)|}{|L(t_2)|} \quad (2)$$

#### 3.2 リクエストに追加・削除されたタプルの割合

サービスチェーンの変化に伴い、タプル被覆率がどのように変化するかをみることでコアの安定性を評価する。本節では、サービスチェーンの変化を明らかにするための指標として、ある時刻  $t_1$  から時刻  $t_2$  までの間に追加されたタプルの割合と削除されたタプルの割合を導入する。ある時刻  $t_1$  における全てのサービスチェーン要求またはパステータに含まれるタプルの集合は  $L(t_1)$  で表される。一方で別の時刻  $t_2$  におけるタプル集合は  $L(t_2)$  で表される。このとき、時刻  $t_1$  から時刻  $t_2$  までの間に サービスチェーン要求に追加されたタプルの割合を式 (3) で、削除されたタプルの割合を式 (4) で求める。

$$\frac{|L(t_2) - L(t_1)|}{|L(t_1)|} \quad (3)$$

$$\frac{|L(t_1) - L(t_2)|}{|L(t_1)|} \quad (4)$$

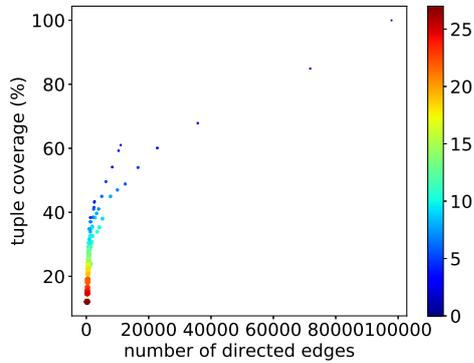


図4 サブグラフを構成する有向辺の数とタプル被覆率

## 4. インターネットの経路データを用いたコア抽出方法の適用性評価

### 4.1 AS パスデータ

本稿では、2. 章で述べたコアの抽出方法と 3. 章で述べた評価指標を NFV のサービスチェーン要求に適用し、抽出したコアの効率性と安定性の評価を行う。しかし、NFV の標準化は策定段階にあるため、実測にもとづくサービスチェーン要求を用いた定量的評価を行うことはできない。そこで、まず実測的なインターネット経路のデータを用いる。インターネット経路データは NFV とは異なるが、情報流の観点では一定の類似性があると考えられる。

インターネットの経路データとして文献 [9] で使用されている AS パスデータを用いる。AS パスデータは RouteViews プロジェクトのサーバで収集された BGP テーブルから抽出した。RouteViews のサーバは、プロジェクト開始からほぼ同一の ISP から BGP テーブルを収集しているため、抽出した AS パスデータは経年での一貫した比較が可能である点で優れている。AS パスデータは "1, 3, 6, 7" のように、トラフィックの通った AS をリストとして表されており、複数のエントリで構成されている。これらのエントリを NFV のサービスチェーン要求に、AS を VNF に相当するものとみなして評価を行う。

### 4.2 コアサイズとタプル被覆率の相関関係

ある時刻の AS パスデータから得られたサブグラフの、その時刻におけるタプル被覆率の導出を行い、コアサイズ (サブグラフを構成する有向辺の数) とタプル被覆率がどのような相関関係にあるかを求める。ここでは、2014 年 1 月 15 日 12 時の AS パスデータを用いている。図 4 はサブグラフを構成する有向辺の数を横軸に、タプル被覆率を縦軸にした散布図である。また、プロットされたポイントのサイズとカラーマップはサブグラフ  $DC_{k,l}$  の  $k+l$  を反映している。図より、各ポイントは  $y = x^{1/2}$  で近似される曲線状にプロットされている。ここから、構成する有向辺が少ないほどタプル被覆率も低いことがわかる。次数  $k+l$  が最も高いサブグラフが最も構成する有向辺の数の少なく、最も低いタプル被覆率を有している。

### 4.3 時間経過に伴う AS パスとタプル被覆率の変化

AS パスデータの変化に伴うタプル被覆率の変化を示す。ここでは  $t=0$  において、構成する有向辺の数と同時刻における

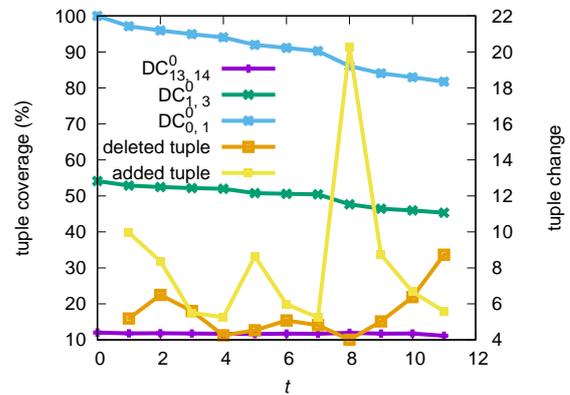


図5 AS パスに追加・削除されたタプルの割合とタプル被覆率の変化

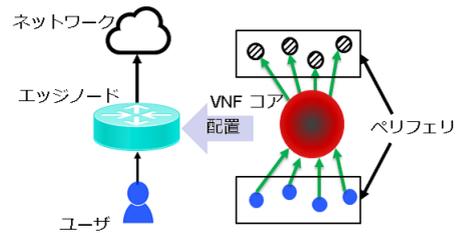


図6 想定する物理ネットワーク環境

タプル被覆率が最も低い  $DC_{13,14}^0$ 、構成する有向辺の数とタプル被覆率が最も高い  $DC_{0,1}^0$ 、タプル被覆率が 50% を超えるもののなかで最も構成する有向辺の数が少ない  $DC_{1,3}^0$  の変化を求める。期間は 2014 年 1 月 15 日 12 時 ( $t=0$ ) から、2014 年 12 月 15 日 12 時 ( $t=11$ ) までの 1 ヶ月ごととしている。図 5 に、各時刻において追加・削除されたタプルの割合とタプル被覆率を示す。図の横軸は時刻  $t$  を示しており、縦軸の左側には各時刻におけるサブグラフのタプル被覆率を、縦軸の右側には 1 ヶ月前から追加・削除されたタプルの割合を示している。図より、最も構成する有向辺の数が少ない  $DC_{13,14}^0$  は他のサブグラフと比べてタプル被覆率の減少が少なく、AS パスの変化に対して安定的であることがよみとれる。特に、約 20% と多くのタプルが追加された 8 月の 9 月の間においても、 $DC_{13,14}^0$  のタプルの減少は少なく、安定的である。

## 5. コア抽出方法にもとづく VNF 配置の効率性及び安定性の評価

### 5.1 ネットワーク構成

NFV の物理ネットワーク環境は、図 6 に示すような他のネットワークに 1 台のエッジノードが接続されているものを想定する。このエッジノードに、2. 章の方法を用いて抽出したコアを配置する。エッジノードには配置できる VNF に上限があるものとする。 $x$  個の VNF を配置したとき、エッジノードで最大  $x(x-1)$  種類のサービスチェーン要求を処理可能となる。また、ここでのコアとは有向グラフを分解して得られたサブグラフの中で、配置できる VNF 数の上限を満たし、最もタプル被覆率の高いものとする。なお、ユーザはエッジノードを介してリクエストを送信するものとし、簡単化のためネットワーク間

表 1 サービスチェイン要求の生成パラメータ: 効率性評価

パラメータ	値
サービスチェイン要求の本数	30000
サービスチェイン要求一本あたりに要求される VNF 数	[10, 20]
VNF の総種類数	30000

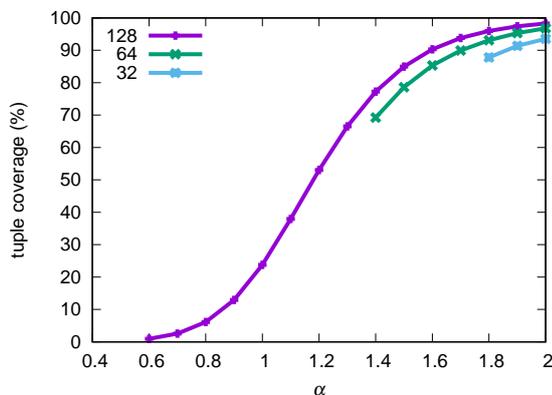


図 7 VNF コアのタプル被覆率

回線の帯域は考慮しないものとする。

## 5.2 サービスチェイン生成モデル

過去の VNF 配置問題を扱う研究では、サービスチェイン要求の生成モデルを導入している。文献 [7], [10] では、サービスチェインを一様な確率で決定する生成モデルを用いている。しかし、実際には要求される VNF には偏在性があり、サービスチェインへの登場頻度の高い VNF と低い VNF があることが予測される。文献 [11] では、配置された VNF で処理できるサービスチェイン要求のヒット率を向上させるために、要求される VNF の偏在性にもとづいた VNF 配置を提案している。それに伴い、要求される VNF の偏在性に基づいたサービスチェイン要求の生成モデルを用いている。ある一つのサービスチェイン要求に含まれる VNF 数を 10 ~ 20 の範囲で一様にランダムで決定し、その後要求される VNF を偏在性のモデルにしたがって決定する。本稿でも同様に、文献 [11] で用いられた生成モデルを使用する。要求される VNF の偏在性はジップの法則を用いてモデル化している。ジップの法則は要素の人気度と出現頻度をモデル化しており、 $j$  番目に人気な要素の出現頻度  $p_j$  は式 (5) で示される。

$$p_j = 1/j^\alpha \sum_{i=1}^N \frac{1}{i^\alpha} \approx 1/j^\alpha \int_1^N \frac{1}{i^\alpha} di = \frac{1-\alpha}{j^\alpha(N^{1-\alpha}-1)} \quad (5)$$

ここで  $N$  は要素の総種類数である。ジップの法則に対して逆関数法を適用することで要求される VNF を決定する。

## 5.3 VNF 配置の効率性および安定性の評価

### 5.3.1 効率性の評価

エッジノードに配置可能な VNF 数の上限を  $x$  とした場合に、VNF コアを配置することでサービスチェイン要求に含まれる VNF 間のリクエストをどの程度処理可能かを明らかにすることにより、VNF コアの効率性を示す。効率性の評価では、表 1 に示すパラメータでサービスチェイン要求を生成した。エッジノードに配置できる VNF 数の上限を 128, 64, 32, ..., 1 に設

表 2 サービスチェイン要求の生成パラメータ: 安定性評価

パラメータ	値
サービスチェイン要求の本数	30000
サービスチェイン要求一本あたりに要求される VNF 数	[10, 20]
VNF の総種類数	30000
$\alpha$	1.0

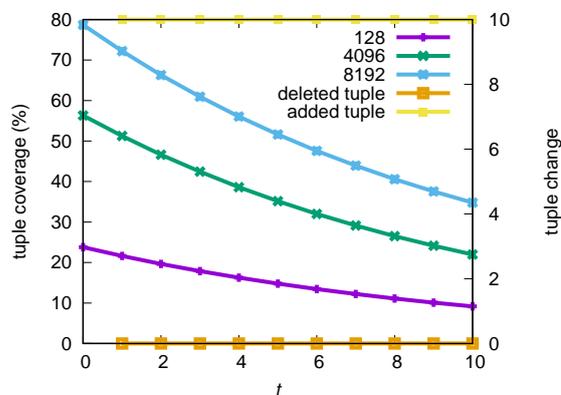


図 8 時間変化に伴う VNF コアのタプル被覆率の変化

定し、これらの配置上限を満たす VNF コアを抽出してタプル被覆率を導出する。

図 7 に、各サービスチェイン要求から抽出した VNF コアのタプル被覆率を示す。図の横軸は各サービスチェイン要求の生成時に用いたパラメータ  $\alpha$  を [0.1, 2.0] の範囲で 0.1 ずつ示している。縦軸にはタプル被覆率を示している。なお、タプル被覆率時には 10 回ずつサービスチェイン要求を生成しそれらの平均をとっている。ただし、エッジノードに配置可能な VNF 数の上限を  $x$  としたとき、5 回以上コアの抽出が行えなければ、そのパラメータのサービスチェイン要求からはコアの抽出が行えないとみなす。図より、VNF コアのタプル被覆率は  $\alpha = 0.1$  に近づくとき 0% に収束し、 $\alpha = 2.0$  に近づくとき 100% に収束する。また、 $\alpha$  が高まるほど、エッジノードに配置可能な VNF 数の上限がより少ない場合でも VNF コアの抽出が可能になる。VNF コアについて、VNF 数の上限が 128 以下の場合  $\alpha = 0.6$  から、64 以下の場合  $\alpha = 1.4$  から、32 以下の場合  $\alpha = 1.8$  から抽出されていることが確認できる。これは  $\alpha$  が高まると要求される VNF の偏在性が高くなり、より少数の VNF に呼び出しが集中するためである。要求される VNF の偏在性が高いほど、より少ないエッジノードのリソースでより多くのリクエストを処理できるようになり、VNF コアは高い効率性を示す。要求される VNF の偏在性が低い場合により多くのリクエストを処理できるようにするためには、エッジノードに配置可能な VNF 数の上限を増やす等の設備投資が必要となる。

### 5.3.2 安定性の評価

安定性の評価では、まず時刻  $t = 0$  において表 2 に示すパラメータで生成したサービスチェイン要求から VNF コアを抽出し、エッジノードに配置するものとする。時間が 1 ステップ進むごとに、 $|L(t)|$  の 10% 分のサービスチェイン要求を追加生成

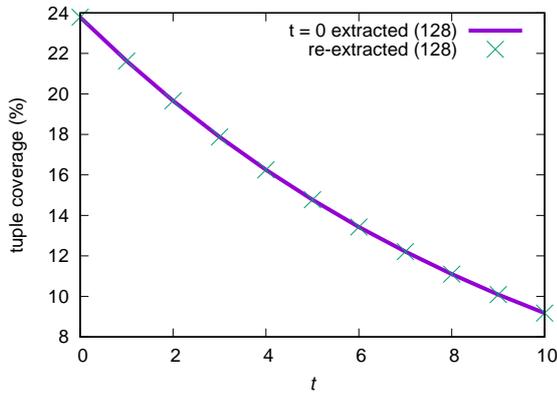


図9  $t = 0$  で抽出した VNF コアと再抽出した VNF コアの比較

する。このとき、追加生成するサービスチェーンは5.2章で述べた生成モデルと異なり、要求される VNF を一様な確率で決定する。図8に時刻  $t = 0$  において抽出した VNF コアのタプル被覆率の時間変化を示す。図の横軸には時刻  $t$  を、縦軸にはタプル被覆率を示している。なお、VNF 数の配置上限は 128, 4096, 8192 である。図より、10 ステップでのタプル被覆率の減少量を比べた場合、エッジノードに配置できる VNF 数の上限が 8192 のときは約 54%、4096 のときは約 40%、128 のときは約 15% 減少している。これは、追加生成されたサービスチェーン要求がランダムに決定されていることから VNF の偏在性が下がり、タプル被覆率が減少したと考えられる。

ここで、サービスチェーン要求の変化にあわせて VNF コアを抽出しなおしエッジノードに再配置を行うことを考える。図9に、配置上限を 128 とし各時刻のサービスチェーン要求から新たに抽出した VNF コアのタプル被覆率を示す。図の曲線は  $t = 0$  において抽出した VNF コアのタプル被覆率の変化を示し、ポイントは各時刻においてサービスチェーン要求から再抽出した VNF コアのタプル被覆率を示している。横軸には時刻  $t$  を、縦軸にはタプル被覆率を示している。図より、 $t = 0$  の VNF コアと各時刻で新たに抽出した VNF コアとの間で、タプル被覆率の差が少ないことがわかる。したがって、エッジノードに配置可能な VNF 数の上限を 128 に設定したとき、サービスチェーン要求の変化に合わせてコアを再抽出しエッジノードに再配置してもタプル被覆率はあまり改善しない。

限定した VNF をコアとすることで、サービスチェーン要求を安定的に処理することができる。しかし、一定のタプル被覆率を維持するためには、配置可能な VNF 数の上限を増やすといった設備投資を含めた VNF 配置変更の頻度をあげるといった工夫を要する。

## 6. おわりに

本稿では VNF の配置問題に焦点をあて、NFV にコアペリフェリモデルを適用することで VNF を物理ネットワークに配置した際の効率性と安定性の向上を図った。NFV から VNF コアを抽出するために、サービスチェーン要求を有向グラフに変換し、サブグラフ分解メトリクスを用いた。サービスチェ

イン要求の標準化は策定段階にあり実測データを用いた評価を行えないため、AS パスデータを用いてコア抽出の適用性の評価をまず行った。その後、ジップの法則にもとづいた生成モデルを用いてサービスチェーン要求を生成し、VNF コアの効率性と安定性を明らかにした。ただし、効率性と安定性はともに VNF の配置リソースとサービスチェーン要求の影響を受ける。要求される VNF の偏在性が高い場合は配置する VNF コアを少数に限定しても高い効率性と安定性を期待することができる。対して、要求される VNF の偏在性が低くリクエストがランダムに分散する場合は、配置リソースを増やしつつ配置変更の頻度をあげるといった工夫が必要となる。今後の課題としては、異なる生成モデルを用いてサービスチェーン要求を生成した場合に抽出したコアの効率性と安定性を明らかにし本稿で得られた結果との比較を行うことや、他の VNF 配置手法と比較を行うことがあげられる。

謝辞 本研究の一部は NICT 委託研究「未来を創る新たなネットワーク基盤技術に関する研究開発」によるものである。ここに記して謝意を示す。

## 文 献

- [1] A. Al-Fuqaha, M. Guizani, M. Mohammadi, M. Aledhari, and M. Ayyash, "Internet of Things: A survey on enabling technologies, protocols, and applications," *IEEE Communications Surveys & Tutorials*, vol. 17, pp. 2347–2376, Jun. 2015.
- [2] "Network Functions Virtualisation Introductory White Paper." ETSI, Oct. 2012.
- [3] "Network Functions Virtualisation - White Paper #3." ETSI, Oct. 2014.
- [4] B. Han, V. Gopalakrishnan, L. Ji, and S. Lee, "Network function virtualization: Challenges and opportunities for innovations," *IEEE Communications Magazine*, vol. 53, pp. 90–97, Feb. 2015.
- [5] N. M. K. Chowdhury, M. R. Rahman, and R. Boutaba, "Virtual network embedding with coordinated node and link mapping," in *Proceedings of IEEE INFOCOM 2009*, pp. 783–791, Apr. 2009.
- [6] P. Csermely, A. London, L.-Y. Wu, and B. Uzzi, "Structure and dynamics of core/periphery networks," *Journal of Complex Networks*, vol. 1, pp. 93–123, Oct. 2013.
- [7] M. Otokura, K. Leibnitz, T. Shimokawa, and M. Murata, "Evolutionary core-periphery structure and its application to network function virtualization," *IEICE Nonlinear Theory and Its Applications*, vol. 7, pp. 202–216, Apr. 2016.
- [8] C. Giatsidis, D. M. Thilikos, and M. Vazirgiannis, "D-cores: measuring collaboration of directed graphs based on degeneracy," *Knowledge and Information Systems*, vol. 35, pp. 311–343, May 2013.
- [9] Y. Nakata, S. Arakawa, and M. Murata, "Analyzing the evolution and the future of the internet topology focusing on flow hierarchy," *Journal of Computer Networks and Communications*, vol. 113, pp. 13–18, Jan. 2015.
- [10] W. Rankothge, J. Ma, F. Le, A. Russo, and J. Lobo, "Towards making network function virtualization a cloud computing service," in *Proceedings of IFIP/IEEE International Symposium on Integrated Network Management (IM)*, pp. 89–97, May 2015.
- [11] Y. Nam, S. Song, and J.-M. Chung, "Clustered NFV service chaining optimization in mobile edge clouds," *IEEE Communications Letters*, vol. 21, pp. 350–353, Oct. 2017.