

## 特別研究報告

題目

脳の認知機能モデルを用いた複数ネットワークスライスへのサービス  
品質を考慮した資源割り当て制御

指導教員

村田 正幸 教授

報告者

安 世民

2019/02/19

大阪大学 基礎工学部 情報科学科

脳の認知機能モデルを用いた複数ネットワークスライスへのサービス品質を考慮した資源  
割り当て制御

安 世民

内容梗概

スマートフォンやタブレットといった高機能なモバイル端末の普及やInternet of Things(IoT)デバイスといった機器の登場とともに、ネットワークを介して提供されるサービスも多様化している。このような多種多様なサービスを収容する方法として、ネットワークスライシング技術の研究が進められている。

スライス化されたネットワークでは、各ネットワークスライスへの資源割り当てが重要な課題となる。スライスへの資源割当は、現在、各スライスに発生しているトラフィック量に合わせて行うのではなく、近い将来に発生するトラフィック量に合わせて行うことが必要となる。将来発生するトラフィック量に合わせて資源割当を行うことにより、トラフィック量の変化に先駆けて資源割当を行うことができ、資源不足による通信性能の劣化を防ぐことができる。

各ネットワークスライスが収容するトラフィックは、人の行動などの現実世界の状況の影響を受ける。そのため、スライスへの動的な資源割当といったネットワーク制御に用いるトラフィック量の予測を行うにあたり、現実世界のセンシングデータは有用な情報となる。我々の研究グループでは、人の脳が曖昧・不完全な観測情報をもとに認知を行う仕組みをモデル化した Bayesian Attractor Model にもとづいた状況認知を行うことにより、現実世界のセンシングデータ、トラフィック量の観測情報の両方を用いる。

本報告では、Bayesian Attractor Model にもとづいた状態認知を、各スライスにおけるトラフィック量の予測・制御に用いるのみではなく、ネットワークスライス間の資源割当にも適用する。各スライスにおけるトラフィック変動のリスクについては、当該スライスに関連する現実世界のセンシングデータとトラフィック量の観測情報をもとに、把握することができる。しかしながら、ネットワークスライス間の資源割り当てを行うためには、各ネットワークにおける資源不足のリスクを把握するのみではなく、全ネットワークスライスの状況を把握することが必要となる。しかし、いずれかのネットワークスライスの状況の把握に有用な情報をすべて収集すると、資源割当の制御を行うコントローラが収集する情報が膨大となり、状態の把握に必要な情報の収集が困難となる。

そのため、本報告では、各スライスにおける状態判断は各スライスに対応する制御を行うコントローラで行い、スライス間の資源割当を行うコントローラでは、各スライスにおける判断結果を収集した上で、ネットワーク全体の情報を判断し、資源割当の制御を行うといった階層的な判断を行う。本報告では、提案する階層的な判断にもとづくネットワークスライス制御を実装し、その動作を確認する。その結果、階層的な認知により、資源割当を行うコントローラにおいて、ネットワーク全体の状況を把握し、各スライスが求める通信性能を維持できるように資源割当を行うことができることを示している。

本報告では、提案手法を適用することで、全時間帯において各スライスでの資源不足が発生しないことを確認した。

## 主な用語

ネットワークスライス、Bayesian Attractor Model、モバイルネットワーク、バックボーンネットワーク、ゲートウェイ

## 目次

<b>1</b>	<b>はじめに</b>	<b>6</b>
<b>2</b>	<b>関連研究</b>	<b>8</b>
2.1	ネットワークスライシング技術 . . . . .	8
2.2	脳の認知機能モデル . . . . .	9
<b>3</b>	<b>脳の認知機能モデルを用いた複数ネットワークスライスへの資源配分手法</b>	<b>11</b>
3.1	概要 . . . . .	11
3.2	スライスコントローラ . . . . .	13
3.2.1	観測情報と認知対象の状態 . . . . .	14
3.2.2	認知 . . . . .	14
3.2.3	制御 . . . . .	16
3.3	資源配分コントローラ . . . . .	16
3.3.1	観測情報と認知対象の状態 . . . . .	16
3.3.2	認知 . . . . .	17
3.3.3	制御 . . . . .	17
<b>4</b>	<b>動作確認</b>	<b>18</b>
4.1	評価シナリオ . . . . .	18
4.1.1	ネットワーク . . . . .	18
4.1.2	ネットワークスライス . . . . .	19
4.1.3	トラフィック量と現実世界の観測情報 . . . . .	19
4.1.4	各コントローラにおける制御 . . . . .	20
4.2	評価結果 . . . . .	22
<b>5</b>	<b>まとめ</b>	<b>29</b>
	謝辞	30
	参考文献	31

## 目 次

1	本手法におけるスライス及びコントローラ構成 . . . . .	12
2	スライスコントローラ . . . . .	13
3	資源配分コントローラ . . . . .	16
4	想定するネットワークの概略 . . . . .	18
5	スライス A の発生トラヒック量 . . . . .	21
6	スライス B の発生トラヒック量 . . . . .	21
7	スライス A の選択枝の確信度遷移 . . . . .	23
8	スライス B の選択枝の確信度遷移 . . . . .	23
9	資源配分コントローラを選択枝の確信度遷移 . . . . .	25
10	各リンクの資源量遷移 . . . . .	26

## 表 目 次

1	評価に用いた Bayesian Attractor Model のパラメータ . . . . .	22
2	スライス A 選択枝別予測トラヒック量 . . . . .	24
3	スライス B 選択枝別予測トラヒック量 . . . . .	24
4	選択枝別スライス A 配分資源量の候補値 . . . . .	25
5	選択枝別スライス B 配分資源量の候補値 . . . . .	27
6	選択枝別スライス C 配分資源量の候補値 . . . . .	27
7	収容できなかったトラヒック量 . . . . .	28

## 1 はじめに

スマートフォンやタブレットといった高機能なモバイル端末の普及や Internet of Things (IoT) デバイスといった機器の登場とともに、ネットワークを介して提供されるサービスも多様化している。ネットワークには、各サービスが要求する品質を満たすように、各サービスのトラフィックを収容することが求められている。このような多種多様なサービスを収容する方法として、ネットワークスライシング技術の研究が進められている [2-4]。ネットワークスライシング技術を用いることにより、ネットワークを仮想的に分割し、ネットワークスライスを構築することができる。このネットワークスライスを各サービスに対して構築することにより、各サービスに応じた仮想的なネットワークを柔軟に構成し、運用することが可能となる。

スライス化されたネットワークでは、各ネットワークスライスへの資源割り当てが重要な課題となる。各スライスに収容されるトラフィック量は時々刻々変化をする。その結果、当初各スライスに割り当てた資源では不足し、発生する通信に対して、十分な通信性能を維持することができなくなることもありうる。この場合、余剰な資源が割り当てられているスライスが存在すれば、そのスライスから資源がひっ迫しているスライスへ資源を割り当てることにより、資源不足を解消することにより、必要な通信性能を維持することができる。

このような動的なスライスへの資源割当は、現在、各スライスに発生しているトラフィック量に合わせて行うのではなく、近い将来に発生するトラフィック量に合わせて行うことが必要となる。将来発生するトラフィック量に合わせて資源割当を行うことにより、トラフィック量の変化に先駆けて資源割当を行うことができ、資源不足による通信性能の劣化を防ぐことができる。

上記の制御において、将来のトラフィック量は、制御時点で得られた情報から予測することにより得られる。従来、トラフィック予測は、トラフィックの時間変動をモデル化しておき、過去に観測されたトラフィック量をもとに当該モデルのパラメータを定め、将来のトラフィック予測に用いる形で行われてきた [5-8]。しかしながら、過去のトラフィック量の観測値をもとにした将来のトラフィック予測では、予測に用いた時系列モデルに従わない変動が発生した場合には、大きな予測誤差を生じる。

各ネットワークスライスが収容するトラフィックは、人の行動などの現実世界の状況の影響を受ける。例えば、モバイルネットワークを対象とすると、人が密集するエリアでは輻輳発生リスクが増加するなど、人の行動の影響を受けることが考えられる。そのため、スライスへの動的な資源割当といったネットワーク制御に用いるトラフィック量の予測を行うにあたり、現実世界のセンシングデータは有用な情報となる。しかしながら、実世界のセンシングデータを含む、観測情報と将来とのトラフィック量との関係は一般的なモデル化が困難である。

我々の研究グループでは、人の脳が曖昧・不完全な観測情報をもとに認知を行う仕組みを

モデル化した Bayesian Attractor Model にもとづいた状況認知を行うことにより、現実世界のセンシングデータ、トラヒック量の観測情報の両方を用いて、将来発生する資源不足のリスクを把握する手法に関する検討を進めている。文献 [9] では、Bayesian Attractor Model にもとづいた状況認知を、モバイルネットワークにおける特定のエリアから流入するトラヒック量が増加するリスクを事前に把握・資源割当制御へ適用した。本手法では、各時刻で観測可能な情報とその後発生する予測対象エリアのトラヒック量について事前に取得した情報をもとに、認知対象となる選択肢と、その選択肢に対応する将来の流入トラヒック量の対応をあらかじめつけておく。そして、実際に制御する際には、各時刻で取得できる観測情報をもとに、現在の状況がいずれの選択肢に合致しているのかを認知することにより、将来のトラヒック量増加のリスクを把握し、制御に用いる。これにより、各時刻において得られる観測情報と将来のトラヒック量との関係を正確にモデル化できない場合であっても、各時刻に観測された情報をもとに、逐次状態を判断し、制御に用いることができる。

本報告では、Bayesian Attractor Model にもとづいた状態認知を、各スライスにおけるトラヒック量の予測・制御に用いるのみではなく、ネットワークスライス間の資源割当にも適用する。各スライスにおけるトラヒック変動のリスクについては、当該スライスに関連する現実世界のセンシングデータとトラヒック量の観測情報をもとに、把握することができる。しかしながら、ネットワークスライス間の資源割り当てを行うためには、各ネットワークにおける資源不足のリスクを把握するのみではなく、全ネットワークスライスの状況を把握することが必要となる。しかし、いずれかのネットワークスライスの状況の把握に有用な情報をすべて収集すると、資源割当の制御を行うコントローラが収集する情報が膨大となり、状態の把握に必要な情報の収集が困難となる。そのため、本報告では、各スライスにおける状態判断は各スライスに対応する制御を行うコントローラで行い、スライス間の資源割当を行うコントローラでは、各スライスにおける判断結果を収集した上で、ネットワーク全体の情報を判断し、資源割当の制御を行うといった階層的な判断を行う。

本報告では、提案する階層的な判断にもとづくネットワークスライス制御を実装し、その動作を確認する。その結果、階層的な認知により、資源割当を行うコントローラにおいて、ネットワーク全体の状況を把握し、各スライスが求める通信性能を維持できるように資源割当を行うことができることを示す。

本報告の構成は以下の通りである。2章で関連研究について述べ、3章で提案する脳の認知機能モデルを用いた複数ネットワークスライスへの資源割当制御について述べる。4章で脳の認知機能モデルを用いた資源割当制御について、シミュレーションにより動作確認を行い、5章でまとめと今後の課題について述べる。

## 2 関連研究

### 2.1 ネットワークスライシング技術

ネットワーク上の資源を仮想化する技術に関する研究が進められている。

Software Defined Networking (SDN) と呼ばれる技術では、物理ネットワークをコントローラで集中管理することにより、柔軟なネットワーク制御を実現する [10]。また、ネットワーク機能についても仮想化する Network Function Virtualization (NFV) に関する検討も進められている [11]。本技術を用いることにより、従来は、専用ハードウェアと一体化したアプライアンスとして提供されていたネットワーク機能を汎用的なコンピュータ上で動作させることができるようになる。NFV を用いた場合、従来の専用アプライアンスでネットワーク機能を提供する場合と異なり、いずれのネットワーク機能も、ネットワーク上に設置された汎用的なコンピュータのいずれで動作させることも可能となり、汎用コンピュータで動作中の仮想マシンを別の汎用コンピュータに異動させる Live Migration 技術と組み合わせることにより、ネットワーク機能をネットワーク内の任意の場所に異動させることができる。そのため、ネットワークの状況に応じて、各ネットワーク機能が必要とされる通信が多い箇所に異動させたり、需要の少ない時間帯には、ネットワーク機能を集約し、多数の機器をスリープさせるといった制御を行うことができる。

ネットワークの仮想化技術を用い、単一の物理ネットワーク上に複数のネットワークスライスを構成する技術に関する研究が進められている。文献 [12] では、物理ネットワークを複数の仮想ネットワークに分割することによるネットワーク仮想化を実現する SDN コントローラを提案している。本手法では、物理ネットワークの帯域や各スイッチの資源を分割したネットワークスライスを構成することができ、各ネットワークスライスに対応したコントローラを配置される。各コントローラは、自身に割り当てられたスイッチや資源のみを操作することができ、独立した運用を行うことが可能となる。

また、ネットワーク中に存在する計算機資源も仮想的に分割することは可能である。Infrastructure as a Service (IaaS) と呼ばれる種類のサービスでは、データセンター事業者等が持つ計算資源を仮想的に分割し、顧客に提供している [13]。近年では、従来型のデータセンターのみならず、よりユーザの近くに配置された計算リソースであるエッジコンピューティングリソースも、各サービスに分割することにより、共通のリソースを用いて、複数のサービスを収容することに関する研究も進められている [14]。

さらに、無線ネットワーク資源についても仮想化を行う手法が検討されている [15]。無線ネットワークの仮想化では、周波数帯域を仮想的に分割する spectrum-level slicing や、ネットワークレベルで分割する network-level slicing の検討が進められている。特に、network-level slicing では、RAN (Radio Access Network) における各種基地局等が利用する周波数

帯域の一部および、該当 RAN と接続するバックボーン回線や、CN (Core Network) が利用する帯域幅の一部を、仮想的に分割する。これにより、単一のモバイルネットワークを用いて、要求が異なり、異なるポリシーで運用される複数のネットワークを構成することができる。

このように、近年は、様々なネットワーク仮想化技術が存在し、有線・無線にかかわらず、単一の物理ネットワークを分割したネットワークスライスを構成することが可能となっている。ネットワークをスライス化して運用することにより、要求が異なるネットワークや運用方針が異なる複数のネットワークを単一の物理ネットワーク上に構築することができ、サービスに合わせた仮想ネットワークスライスを構成するなど、柔軟なネットワーク運用が可能となる。

しかしながら、スライス化されたネットワークにおいては、スライス間の資源割当てが重要な課題となる。各スライスに対する需要は時々刻々変化するため、ある時点において適切にスライス間の資源割当てを行ったとしても、時間の経過とともに、需要は変化し、需要が当初より増加し必要な通信性能を維持することができないスライスが生じることも考えられる。本報告では、各時刻において、ネットワークの状況を把握し、スライス間の資源割当てを動的に決定する技術について検討を行う。

## 2.2 脳の認知機能モデル

本報告では、現実世界のセンシング情報を生かしながら、時々刻々変化する状況を把握し、把握した状況に合わせてネットワークを制御することを目標とした検討を行う。ただし、現実世界の情報と、制御に必要な将来の需要の関係は曖昧であり、正確なモデル化は困難である。そこで、本報告では、曖昧かつ不完全な情報をもとに逐次判断を行っている代表的なものとして考えられる人の脳が認知を行うプロセスのモデルをもとに、状況の判断を行うことを検討する。

Bayesian Attractor Model [16] は、脳の認知機能のモデルの一つであり、観測情報を入力とし、与えられた意思決定状態から一つを選ぶ意思決定モデルである。本モデルでは、現実世界を観測し、抽象化することにより、特徴ベクトル  $x_t$  を得る。そして、特徴ベクトル  $x_t$  に基づいて意思決定状態  $z_t$  を更新する。BAM が行う状況判断は、意思決定状態  $z_t$  によって決定される。事前に用意した  $S$  個の選択肢 (以後アトラクターと呼ぶ) に対応した状態値  $\phi_1, \dots, \phi_i$  のいずれかに状態  $z_t$  が到達した時、すなわち、 $z_t = \phi_i$  となったとき、 $i$  番目のアトラクターが意思決定の結果となる。この時、状態更新にはベイズ推定の考え方が用いられており、 $z_t$  は一点として更新されるのではなく、観測値の不確かさ、およびダイナミクスの不確かさを反映した確率分布  $P(z_t)$  として更新される。また、 $z_t$  の値は確率を伴って表現さ

れているため、 $z_t = \phi_i$  の判定には確率的な意思決定が導入される。

$z_t$  の更新は、生成モデルをもとに行われる。文献 [16] では、以下の式で生成モデルを構成した。

$$z_t - z_{t-\Delta_t} = \Delta_t f(z_{t-\Delta_t}) + \sqrt{\Delta_t} w_t \quad (1)$$

$$x_t = M\sigma(z_t) + v_t \quad (2)$$

ここで、 $f(z)$  はホップフィールドダイナミクス、 $w_t$ 、 $v_t$  はノイズ項である。 $M = [\mu_1, \dots, \mu_N]$  であり、 $\mu_i$  は、事前に用意された各アトラクターの状態値  $\phi_i$  に対応する観測値である。ベイズ推定を用いて、上の生成モデルを逆推定することで、観測値  $x_t$  から意思決定状態  $z_t$  を更新することができる。ただし、生成モデルは非線形状態空間モデルであり、厳密なベイズ推定は困難である。そのため、文献 [16] では、UKF を用いて近似計算を行っている。

上述の状態推定によって得られる  $z_t$  は、一点としての  $z_t$  でなく、 $z_t$  の事後確率  $P(z_t|x_t)$  である。そのため、意思決定状態がどの選択肢  $\phi_i$  にあるかの判別は確率を伴って行われる。具体的には、しきい値  $\lambda$  を導入し、 $z_t = \phi_i$  における確率密度が  $P(z_t = \phi_i) > \lambda$  であるような選択肢  $i$  を意思決定の結果として選択する。また、このような  $i$  が存在しない場合には、意思決定が終わっていない段階にあると判断される。

我々の研究グループでは、Bayesian Attractor Model にもとづいた状況認知により、現実世界のセンシングデータ、トラヒック量の観測情報の両方を用いて、将来発生する資源不足のリスクを把握する手法に関する検討を進めている。文献 [9] では、Bayesian Attractor Model にもとづいた状況認知を、モバイルネットワークにおける特定のエリアから流入するトラヒック量が増加するリスクを事前に把握・資源割当制御へ適用した。本手法では、各時刻で観測可能な情報とその後発生する予測対象エリアのトラヒック量について事前に取得した情報をもとに、認知対象となる選択肢と、その選択肢に対応する将来の流入トラヒック量の対応をあらかじめつけておく。そして、実際に制御する際には、各時刻で取得できる観測情報をもとに、現在の状況がいずれの選択肢に合致しているのかを認知することにより、将来のトラヒック量増加のリスクを把握し、制御に用いる。これにより、各時刻において得られる観測情報と将来のトラヒック量との関係を正確にモデル化できない場合であっても、各時刻に観測された情報をもとに、逐次状態を判断し、制御に用いることができる。

本報告では、Bayesian Attractor Model にもとづいた状態認知を、各スライスにおけるトラヒック量の予測・制御に用いるのみではなく、ネットワークスライス間の資源割当に適用する。

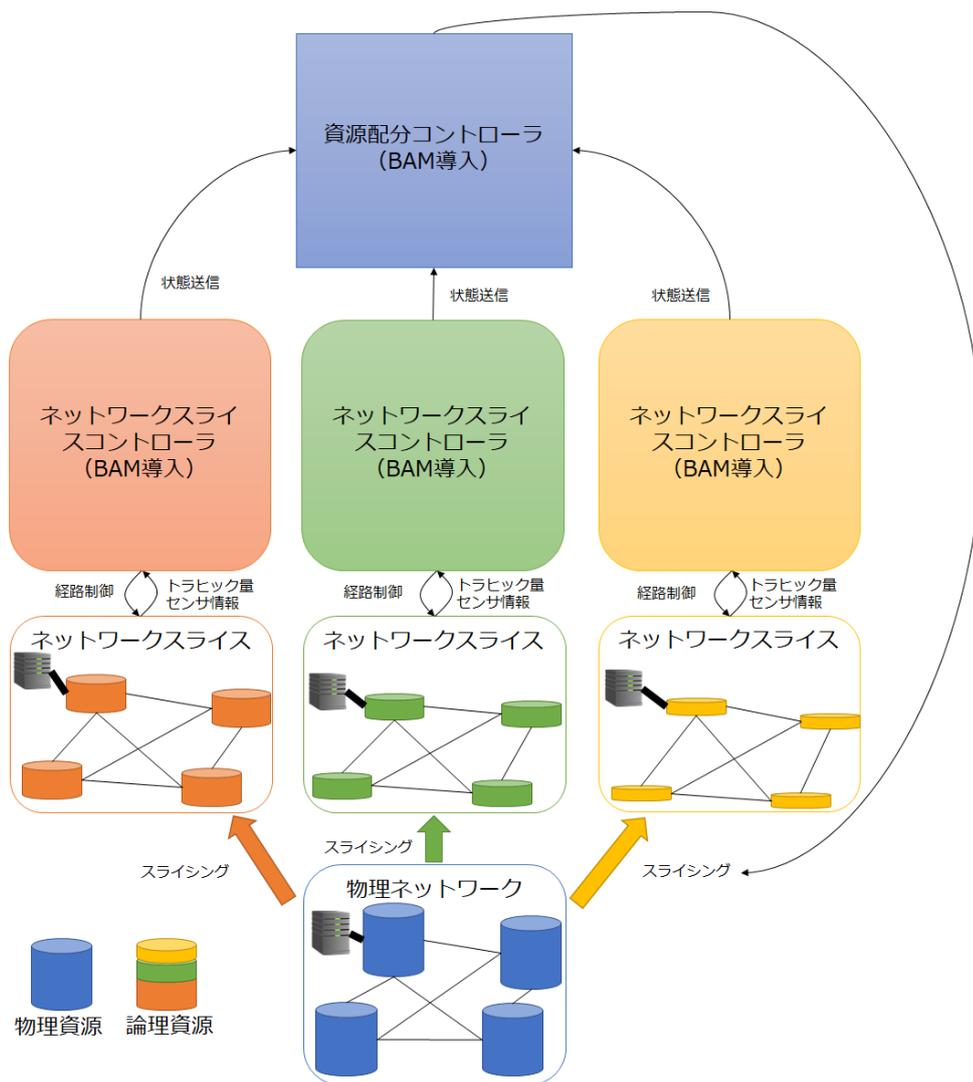
### 3 脳の認知機能モデルを用いた複数ネットワークスライスへの資源配分手法

#### 3.1 概要

本報告では、有線・無線のリンク、エッジコンピューティングで用いられる RAM、CPU、ストレージといった計算機資源を各スライスに分配し、各スライスは分配された資源を用いて当該スライスのサービスを提供する環境を考える。本環境において、各スライスの需要は時々刻々変化する。この変化に対応し、必要なサービス品質を維持するためには、各スライス内において、割当られた資源を用い、経路や機能配置等を状況に合わせて変更することと合わせて、資源がひっ迫しているスライスへ余剰資源が存在しているスライスから資源を再配分することが必要となる。

本報告では、各ネットワークスライス内における経路や機能配置等は、ネットワークスライスに対して設置されるネットワークスライスコントローラが行い、ネットワークスライスの資源配分は、資源配分コントローラによって行う。図1に本手法における構成を示す。ネットワークスライスコントローラでは、対応するネットワークスライス内の近い将来発生する需要を把握し、その需要に対応するように、ネットワーク機能の再配置や経路制御を行う。近い将来発生する需要の予測は、従来、需要の時間変動をモデル化しておき、過去に観測された需要をもとに当該モデルのパラメータを定め、将来の需要予測に用いる形で行われてきた。しかしながら、過去の需要の観測値をもとにした将来の需要予測では、予測に用いた時系列モデルに従わない変動が発生した場合には、大きな予測誤差を生じる。一方、各ネットワークスライスの需要は、現実世界の状況の影響を受ける。例えば、モバイルネットワークを対象とすると、人が密集するエリアでは輻輳発生リスクが増加するなど、人の行動の影響を受けることが考えられる。そのため、本報告では、スライスコントローラは、観測された需要のみではなく、当該スライスのサービスに関連し、現実世界からセンシング可能なデータも収集した上で、将来、当該スライスの各地点で発生する需要を判断する。ただし、現実世界からセンシング可能な情報も含む、各時刻で収集された情報と、将来の需要増加のリスクの対応は曖昧である。そこで、本報告では、スライスコントローラにおける将来発生する需要の把握は、曖昧・不完全な情報から逐次認知を行っている脳の認知機能モデルにもとづいた手法を適用する。すなわち、事前に観測された情報をもとに、認知対象となる選択肢を定め、各選択肢に対応する将来の需要を対応を定める。そして、各時刻において観測された情報をもとに、認知状態を更新することを繰り返すことにより、現在の状況がいずれの選択肢に該当するかを判断する。また、ネットワークスライスコントローラでは、各選択肢と当該スライスに割り当てられた資源の組み合わせに対して、適切な経路や機能配置をあら

図 1: 本手法におけるスライス及びコントローラ構成



かじめ計算して保持しておき、現在の状況がいずれの選択肢に該当するかを認知すると、認知された状態に対する制御を行う。

資源配分コントローラでは、全ネットワークスライスの状態を把握した上で、各スライスに割り当てる資源を決定する。全ネットワークスライスの状態の把握には、各スライスコントローラにおける認知状態を収集し、用いる。これにより、資源配分コントローラでは、実世界のセンシング状況を直接収集したり、解析することなく、それらの情報が反映された各スライス内の状況に関する情報を取得することができる。

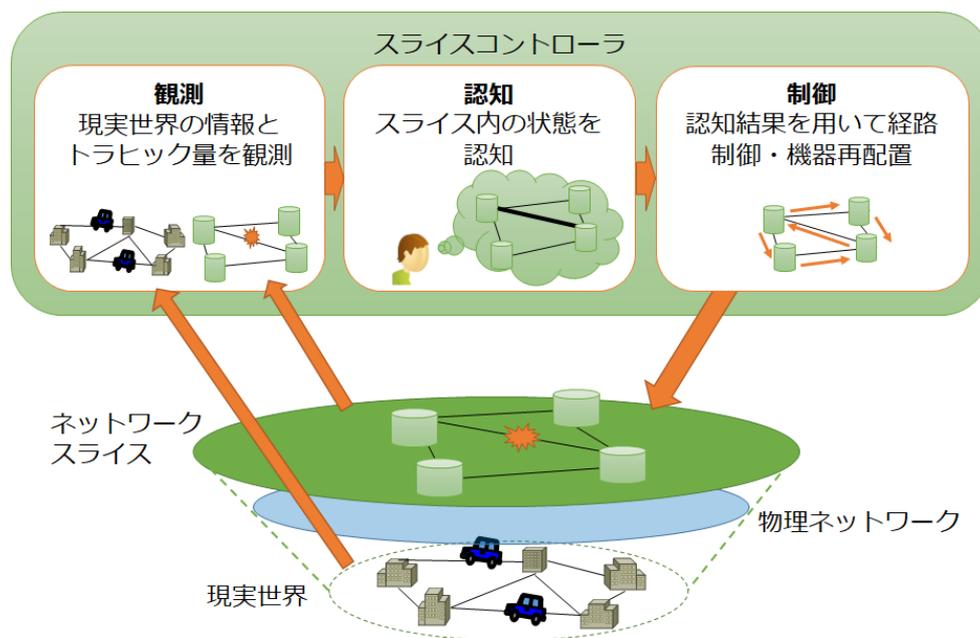
資源配分コントローラでは、あらかじめ観測やシミュレーション等により、資源配分コントローラが観測する情報と、その情報が得られた際の適切な各スライスへの資源配分の対応を計算しておき、計算された資源配分の各候補に対応する状態を、判断における選択肢とする。そして、各時刻において、各スライスコントローラの認知状態を収集し、資源配分コントローラにおける認知状態を更新し、対応する資源配分の制御を行うことを繰り返す。

以降、スライスコントローラ、資源配分コントローラについて、認知対象の選択肢、観測情報を用いた認知の方法、認知後の制御について述べる。

### 3.2 スライスコントローラ

本提案手法における、スライスコントローラの構図を図2に示す。

図 2: スライスコントローラ



### 3.2.1 観測情報と認知対象の状態

各スライスコントローラは、各時刻で得られる観測情報をもとに、将来、当該スライスで発生する需要を判断する。すなわち、各スライスコントローラでは、現在の状態が、近い将来、どのような需要を発生する可能性があるのかの認知を行う。

本報告では、事前に観測された観測情報、需要の情報をもとに、あらかじめ、現在の状況と将来発生しうる需要の対応関係を求め、その関係をもとに、各時刻の状況をあらかじめ分類、分類された各状態に対応する将来発生しうる需要の情報を別途保持する。これにより、各時刻においては、得られた情報をもとに、現在の状況がいずれの状況に合致するのかを判断することにより、将来の需要を判別することができる。

以降、観測情報と認知対象となる状態、また、各状態に対応する将来発生しうる需要の定め方について述べる。本報告では、各時刻  $t$  に観測可能な現実世界でセンシングされた情報と観測された通信需要を表すベクトル  $m_t$  と、時刻  $t+i$  における通信需要  $D_{t+i}$  の組み合わせを事前の観測情報から得ているものとする。まず、得られた情報をもとに、観測情報  $m_t$  が類似するクラスタを構成する。本クラスタは、観測情報が類似しているため、その時点における状況としては類似した状態と考えることができる。しかしながら、類似した状態であっても、将来発生する需要は、一致するとは限らない。本報告におけるスライスコントローラの制御においては、将来、需要が増加するリスクの有無を判別することができれば、需要が増加するリスクが高くなると、当該需要の増加に備えた制御を行うことができる。そのため、本報告では、需要増加のリスクの有無が判断できるように、クラスタ  $C$  について、対応する予測需要  $\hat{D}_C = \hat{d}_{C,1}, \hat{d}_{C,2}, \dots, \hat{d}_{C,k}$  を定める。

$$\hat{d}_{C,k} = \left\lceil \frac{\max_{t \in C} \max_{0 \leq i \leq H} d_{t+i,k}}{U} \right\rceil U$$

ただし、 $\hat{d}_{C,k}$  はクラスタ  $C$  に対応する需要の地点  $k$  に対応する需要、 $H$  は予測対象区間の長さ、 $d_{t+i,k}$  は時刻  $t+i$  における地点  $k$  の観測された需要であり、 $U$  は需要を離散化する際の単位である。その後、 $\hat{D}_C$  が同一であるクラスタを束ね、構成されたクラスタを認知対象とする。これにより、スライスコントローラでは、将来発生しうる需要に対応した認知対象を定めることができ、各時刻において、現在の状況が、いずれのクラスタに該当する状態かを認知することにより、現在の状態が、将来、需要が増大するリスクが高い状態であるのかを把握することが可能となる。

### 3.2.2 認知

脳の認知のモデルでは、逐次観測情報が得られる度に、観測情報を抽象化した特徴ベクトルを取得し、生成モデルをもとに認知の状態を更新することにより、現在の状況の認知を

行う。

以降、本手順における抽象化の方法、生成モデルを用いた認知状態の更新方法について述べる。

**抽象化** 本報告では、得られた観測情報  $m_t$  を抽象化した観測ベクトル  $x_t$  を生成し、Bayesian Attractor Model にもとづく認知の入力として用いる。観測ベクトル  $x_t$  の生成方法は、得られた観測値をそのまま用いるといった方法や、機械学習を用いる方法など、任意の方法を用いることができるが、本報告では、過去に経験し、前節の方法でクラスタリングし、認知対象の選択肢のうち、いずれの選択肢に該当するのかが定められた情報を蓄積した上で、 $k$  近傍法にもとづいて抽象化を行う。

$k$  近傍法にもとづく抽象化では、認知の選択肢数を  $S$  とすると、次元数  $S$  のベクトル  $x_t = (x_{t,1}, x_{t,2}, \dots, x_{t,S})$  を以下の手順により生成する。まず、新たに得られた観測データ  $m_t$  と、これまでに経験・蓄積した観測データ  $m_i$  を比較し、新たに得られた観測データ  $m_t$  から最も近い過去の観測データを  $k$  個選択する。近い観測データを選ぶ際の距離の定義としては任意の定義を用いることができるが、本報告では、 $m_t$  の各要素をこれまでに蓄積された観測データ中の最大値が1となるように正規化した上で、ユークリッド距離を求めることにより、定める。そして、 $x_{t,i}$  を選択された  $k$  個のデータのうち、 $i$  番目の選択肢に対応するデータの割合として定める。これにより、各観測データが得られる度に、当該観測データが、どの選択肢に近いのかを示すベクトル  $x_t$  を得ることができる。

**状態の更新** 本報告では、文献 [9] と同じく、以下の生成モデルを用いる。意思決定状態  $z_t$  及び観測値  $x_t$  の生成モデルを持つ。

$$z_t - z_{t-\Delta t} = \Delta t f(z_{t-\Delta t}) + \sqrt{\Delta t} w_t \quad (3)$$

$$x_t = M\sigma(z_t) + v_t \quad (4)$$

ここで、 $f(z)$  はホップフィールドダイナミクス、 $w_t$ 、 $v_t$  はノイズ項である。 $M$  は、選択肢に対応する抽象化後の観測値を結合した行列であり、本報告では、 $S \times S$  の単位行列となる。

認知状態を更新する際には、ベイズ推定を用いて、上の生成モデルを逆推定することで、抽象化された観測値  $x_t$  から意思決定状態  $z_t$  を更新することができる。ただし、生成モデルは非線形状態空間モデルであり、厳密なベイズ推定は困難であるため、本稿では、粒子フィルタによる近似計算により求める。

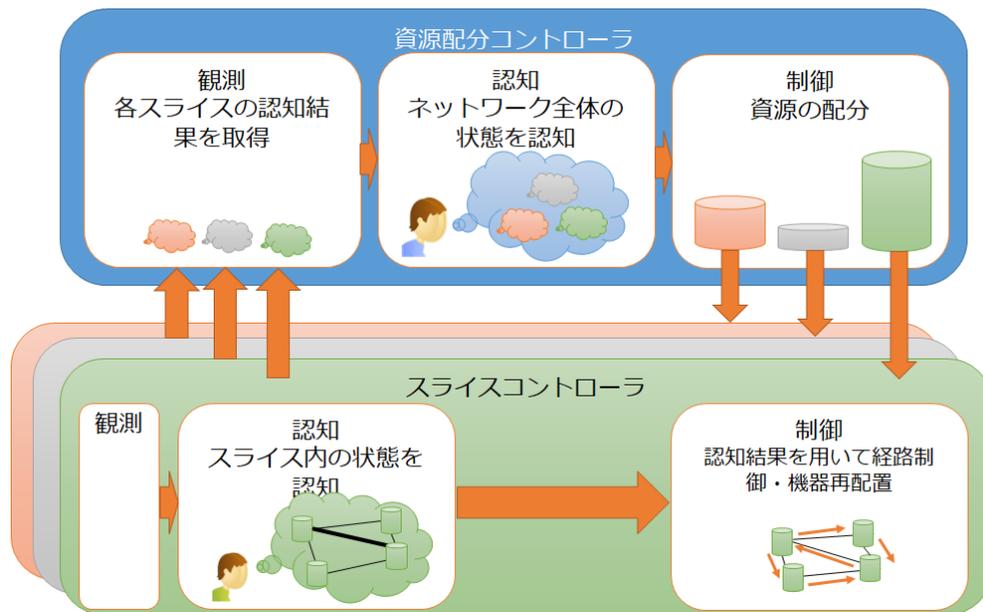
### 3.2.3 制御

本報告では、過去に発生した需要や想定した状況をもとに、発生しうる需要  $\hat{D}_C$  とスライスへの資源割当状況  $R$  の各組み合わせに対して、適切な経路、機能配置をあらかじめ最適化問題や発見的手法により、計算しておく。そして、各時刻においては、得られた認知結果にもとづいて、対応する制御を行う。具体的には、各時刻において、確信度  $P(z_t = \phi_i)$  が最大となる  $i$  を求め、 $i$  番目の選択肢に対応するクラスター  $C_i$  の状況において、近い将来発生しうる需要  $\hat{D}_{C_i}$  が定め、現在の資源割当  $R$  と  $\hat{D}_{C_i}$  の組み合わせに対応する制御を実行する。

### 3.3 資源配分コントローラ

本提案手法における、資源配分コントローラの構図を図3に示す。

図3: 資源配分コントローラ



#### 3.3.1 観測情報と認知対象の状態

資源配分コントローラでは、各スライスから認知状態を収集し、ネットワーク全体の情報を把握する。本報告では、各スライスが資源配分コントローラに送る情報は、当該スライス内での認知を行う際の選択肢数が  $S_i$  の場合、 $o_{t,j} \left( \frac{P(z_t = \phi_1)}{\max_j P(z_t = \phi_j)}, \frac{P(z_t = \phi_2)}{\max_j P(z_t = \phi_j)}, \dots, \frac{P(z_t = \phi_{S_i})}{\max_j P(z_t = \phi_j)} \right)$  とする。これにより、もっとも確信度が高い選択肢に対応する要素が1となり、確信度が低

い選択肢に対応する要素ほど0に近くなる値を持つベクトルが生成される。資源配分コントローラでは、本ベクトルを各スライスコントローラから収集し、結合したベクトル  $O_t = (o_{t,1}, o_{t,2}, \dots, o_{t,N})$  を入力として、各時刻における観測値として用いる。

対応する認知対象の状態は、適切な資源配分に対応させて生成する。本報告では、過去の観測結果や想定される場面のシミュレーションにより得られた、各スライスで発生する需要に対して、最適化問題や発見的手法により、各スライスへの資源配分を決定する。そして、その場面において、各スライスにおいて正しく状況が認知できた場合の観測値  $O_t$  と対応付けて、各スライスへの資源配分を蓄積する。そして、資源配分コントローラでは、新たな観測値が得られた際に、いずれの資源配分に対応する状態であるのかを認知する。

### 3.3.2 認知

資源配分コントローラでは、新たな観測値が得られた際に、いずれの資源配分に対応する状態であるのかを認知する。認知は、観測情報  $O_t$  を抽象化し、抽象化した情報を入力とし、生成モデルをもとに認知状態の更新を行う。本報告では、認知の際の抽象化、認知状態の更新の手順は、スライスコントローラと同一の手順を用いる。

### 3.3.3 制御

資源配分コントローラでは、各選択肢に対応する各スライスへの割当資源量が蓄積されている。そのため、各時刻において、確信度  $P(z_t = \phi_i)$  が最大となる  $i$  を求め、 $i$  番目の資源割当量の候補にあわせて、各スライスへの資源割当を行う。

## 4 動作確認

本報告では、提案手法により、全ネットワークスライスの状況を認知し、各状況に合わせてネットワークスライスへの資源配分を行うことができることを、シミュレーションにより確認する。

### 4.1 評価シナリオ

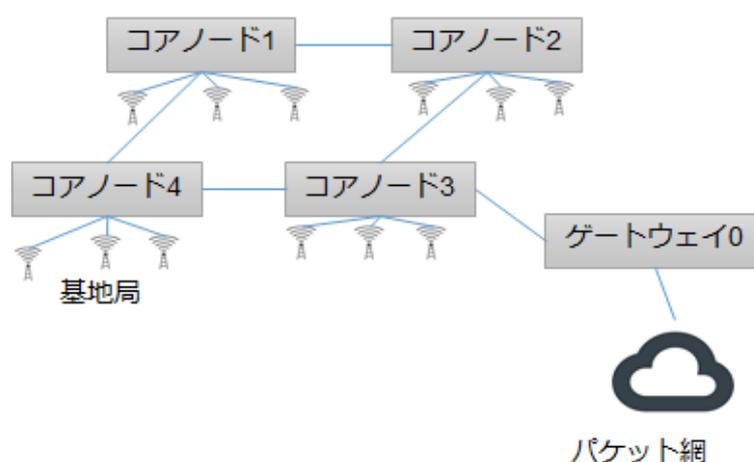
#### 4.1.1 ネットワーク

本報告では、図4に示す物理ネットワークにおいて、リンク帯域のみをスライスに分割可能な資源とし、各スライスでは、割り当てられたリンク帯域をもとに、発生する通信を収容できるようにトラヒックの経路を制御するといった環境で、提案手法の動作を確認する。

本ネットワーク構成は、セルラーネットワークを想定した構成となっており、図中の各ノードは、セルラーネットワークにおけるコアノードとなっている。本ネットワークでは、コアノードは、配下の基地局と接続しており、基地局から流入する通信は、コアネットワークを経由してゲートウェイに送られ、ゲートウェイからパケットネットワークに送出される。逆に各基地局への通信は、パケットネットワークからゲートウェイを経由してコアネットワークに流入し、コアネットワークを経由して各基地局に送出される。

本ネットワークで、ネットワークスライスに割り当てらる資源は、各リンクの帯域であり、本評価では、各リンクは、上り・下りそれぞれ 25Gbps の容量があるものとする。

図 4: 想定するネットワークの概略



#### 4.1.2 ネットワークスライス

本評価では、以下の3つのネットワークスライスに資源を割り当てるものとする。

- スライス A

低遅延であることが求められるサービスに対応するスライス。本報告では、ネットワーク接続された自動車に対する安全運転支援に関するサービスを想定する。低遅延であることを保証するためには、ネットワーク内で発生する遅延についても考慮することが必要となるが、本報告では、簡単のため、本スライスのパケットは物理ネットワーク上において優先制御をされることを前提とし、短いホップ数での収容が求められるものとする。

- スライス B

必要な帯域を確保することが求められるスライス。本報告では、モバイルユーザが動画のストリーミング再生を行うことを想定する。本報告では、ホップ数を増大させても、十分な帯域を確保できれば、必要な通信性能を確保できるものとした。

- スライス C

その他のサービス。スライス A、スライス B に割り当てた帯域以外の余剰帯域を用いてベストエフォート型のサービスの提供を行うスライス。

#### 4.1.3 トラヒック量と現実世界の観測情報

本報告では、各スライスにおいて、観測可能な情報は、各エリアにおける当該スライスに関連するユーザの人数とトラヒックの流入・流出量であるとした。本報告では、各エリアにおけるユーザの人数は、人、自動車を含む首都圏の日常的な人の移動軌跡を再現した疑似データセットである、Open PFLOW[17] をもとにして生成した。本データは、各時刻における各ユーザの位置座標と徒歩、自動車、電車、静止等の当該ユーザの移動属性が与えられている。そのため、本報告では、この移動属性をもとに、各エリア内における各スライスに関するユーザの人数を定め、各ユーザが送出するトラヒック量を別途定めることにより、当該エリアから流入するトラヒック量を定めた。ただし、Open PFLOW は、典型的な1日分のデータしか含まない。そこで、本報告では、得られた各時刻のエリア内のユーザ数に対して、乱数を加えることにより、学習時と異なるユーザ数の状況を生成し、観測情報と対応するコントローラの選択肢の計算を行う際と、コントローラによる制御を動作させる際のユーザの人数、トラヒック量が異なるものとなるようにした、

本報告では、Open PFLOW のデータのうち、新宿区のある地点（東経 139.691558 北緯 35.714647）を起点とし、その位置から東へ 4km、南へ 4km の 4km 四方のエリアを対象とした。Open PFLOW では、本エリアに含まれる観測されたユーザ総数は 50,565 人であった。

以下、各スライスに対応するユーザ、トラフィック量の定め方について述べる。

**スライス A** 本評価では、スライス A は自動車の安全運転支援サービスに関するトラフィックが収容されるスライスであることを想定している。そのため、スライス A に関するトラフィックは、エリア内の自動車が生産するものとした。また、各ユーザが生産するトラフィック量は、上り・下りともに一定のレートであるとし、上りは  $\infty$  bps、下りは  $\infty$  bps とした。図 5 にスライス A に関連する各エリアからの流入・流出トラフィックを示す。

**スライス B** 本評価では、スライス B は、モバイルユーザが送出するトラフィックであると想定し、エリア内の自動車以外のユーザが送出するトラフィックのうち、一定の割合のトラフィックが属するものとした。各ユーザが送出するトラフィックは、総務省の情報通信データベース [18] をもとに定める。本データベースには、時間帯別の移動通信トラフィック量と移動通信サービス契約者数が含まれており、本データより、各時刻における契約者一人あたりの移動通信トラフィック量が求める。本報告においては、各時刻において、各ユーザが送出するトラフィック量は、上述のようにして求めた契約者一人あたりの移動通信トラフィック量に乱数を加えた値とする。図 5 にスライス A に関連する各エリアからの流入・流出トラフィックを、図 6 にスライス B に関連する各エリアからの流入・流出トラフィックを示す。

#### 4.1.4 各コントローラにおける制御

**スライスコントローラ** 本報告では、スライスコントローラは、経路制御をするものとした。すなわち、割り当てられた資源と認知された将来発生しうる各エリアからの流入・流出トラフィック量が与えられた際に、対応する経路設定を定める。本評価では、将来発生しうる各エリアからの流入・流出トラフィック量を降順にソートし、大きいものから順に、当該トラフィックを収容する帯域があることを条件とした Constrained Shortest Path First (CSPF) により経路を決定する。これにより、最短ホップで当該トラフィックを収容することができれば、最短ホップで収容し、最短ホップでは十分な帯域を確保することができない場合であっても、十分な帯域を確保できる経路があれば、その経路にトラフィックが収容される。

また、スライスコントローラにおいては、現在の状態の認知に、Bayesian Attractor Model を用いている。Bayesian Attractor Model では、生成モデル中に含まれる 2 つの誤差項  $v$ 、 $w$  の標準偏差をパラメータとして定める必要がある。本評価では 1 に示すパラメータを用い、5 分に一回、状態の判断を行うものとした。

図 5: スライス A の発生トラフィック量

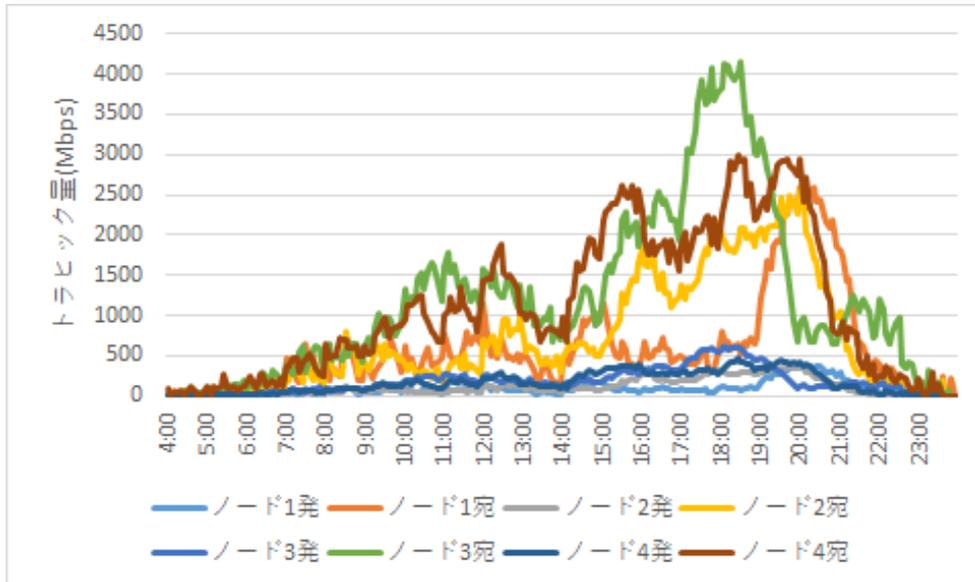


図 6: スライス B の発生トラフィック量

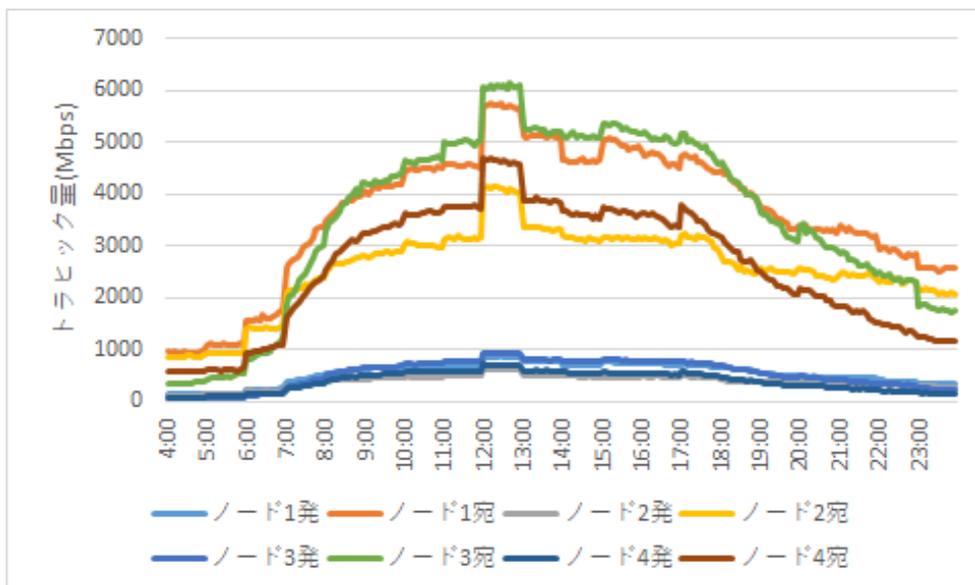


表 1: 評価に用いた Bayesian Attractor Model のパラメータ

パラメータ名	スライスコントローラ	資源配分コントローラ
$v$ の標準偏差	0.30	0.10
$w$ の標準偏差	0.20	0.10

**資源配分コントローラ** 本評価では、資源配分コントローラにおいては、スライス A、スライス B、スライス C に割り当てる資源を決定する。この資源配分は、各状況に対応する将来発生しうるトラヒック量をもとめ、スライス A において、そのトラヒックを最短ホップ収容でき、スライス B で十分な通信帯域を確保することができるように定める。具体的には、スライス A のトラヒックを最短ホップで収容し、その際に、スライス A が必要とする各リンクの帯域をスライス A に割り当てる。また、スライス A のトラヒックを収容した後の残余帯域を用いて、スライス B のトラヒックを CSPF により収容を試みた場合に、各リンクにおいてスライス B が必要とする帯域をスライス B に割り当て、残余帯域をスライス C に割り当てる。

資源配分コントローラにおいても、スライスコントローラと同様に Bayesian Attractor Model による認知を行う。その際のパラメータは表 1 に示すとおりである。

## 4.2 評価結果

図 7、8 にスライス A、スライス B それぞれに対応するコントローラにおいて、各認知対象の選択枝の確信度の時間変動を示す。

これらの図では、横軸は時間を示し、縦軸は各時刻における各選択枝の確信度を示す。また、本評価では、各スライスにおいて、1 時間以内に発生しうるトラヒック量の予測値に対応する選択枝が構成されており、各スライスコントローラの認知対象となる選択枝とその選択枝に対応するトラヒック量の予測結果を表 2、3 に示す。各数字はノードを示し、「発」はそのノードから流出するトラヒック量、「宛」はそのノードへ流入するトラヒック量を示す。

図 7、8 より、各スライスのコントローラにおいて、状況の変化に対応して、認知状態が変化をしていることが分かる。スライス A においては、まず、トラヒック量が少ない状態であると認知され、14 時から 21 時にかけてのトラヒック増加に応じてトラヒック量が多い状態であると認知されている。14 時から 21 時までの間は、各ノードにおける発生トラヒックの増加、減少に対応して、各リンクの必要資源量が変化している。スライス B においては、こちらもまずトラヒック量が少ない状態であると認知され、11 時から 18 時までの間は、ト

図 7: スライス A の選択枝の確信度遷移

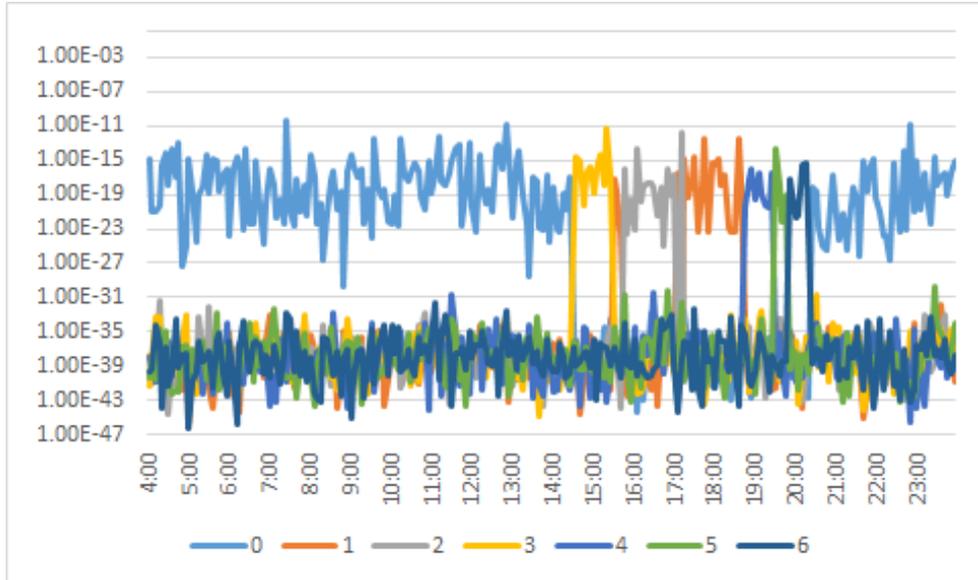


図 8: スライス B の選択枝の確信度遷移

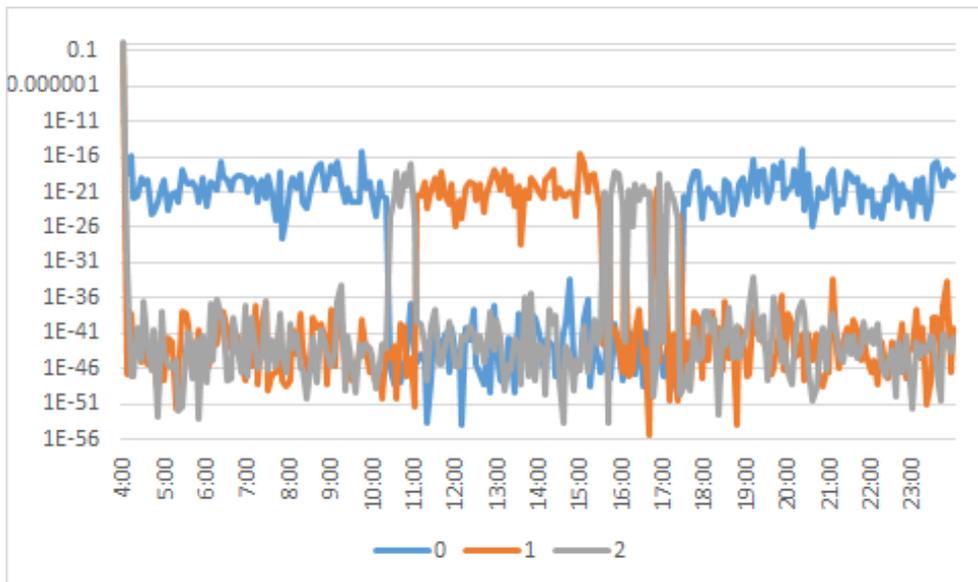


表 2: スライス A 選択枝別予測トラヒック量

選択枝	ノード 1 発	1 宛	2 発	2 宛	3 発	3 宛	4 発	4 宛
0	5000	5000	5000	5000	5000	5000	5000	5000
1	5000	5000	5000	5000	5000	10000	5000	10000
2	5000	5000	5000	5000	5000	10000	5000	5000
3	5000	5000	5000	5000	5000	5000	5000	10000
4	5000	10000	10000	5000	5000	10000	5000	10000
5	5000	10000	10000	5000	5000	5000	5000	10000
6	5000	10000	5000	5000	5000	5000	5000	10000

表 3: スライス B 選択枝別予測トラヒック量

選択枝	ノード 1 発	1 宛	2 発	2 宛	3 発	3 宛	4 発	4 宛
0	5000	5000	5000	5000	5000	5000	5000	5000
1	5000	10000	5000	5000	5000	10000	5000	5000
2	5000	5000	5000	5000	5000	10000	5000	5000

ラヒック量が多い状態であると認知されている。この時間の間、各リンクの必要資源量が変化していることが分かる。

図 9 に資源配分コントローラにおける、選択枝の確信度遷移を示す。図 9 では、横軸は時間を示し、縦軸は各時刻における各選択枝の確信度を示す。また、表 4、5、6 に資源配分コントローラにおける、スライス A、スライス B、スライス C への資源配分量の候補値を示す。

図 9 より、資源配分コントローラにおいても、時間経過とともに、認知状態が変化している。シミュレーション開始当初は、スライス A、スライス B ともに確保が必要な資源量が少ない状態であると判断されている。その後、日中のスライス B のトラヒック量が多い時間帯には、スライス B への資源配分が必要な状態に移行し、夕方の自動車が出発するトラヒック量が多い時間帯には、スライス A への資源配分が必要な状態であると認知されている。

図 10 に各時間帯において、各スライスに割り当てられた資源量を示す。図 10 では、本評価に用いたネットワークの各リンクについて、各時刻にスライス A、スライス B、スライス C に割り当てられた資源量を示している。

図 9: 資源配分コントローラの選択枝の確信度遷移

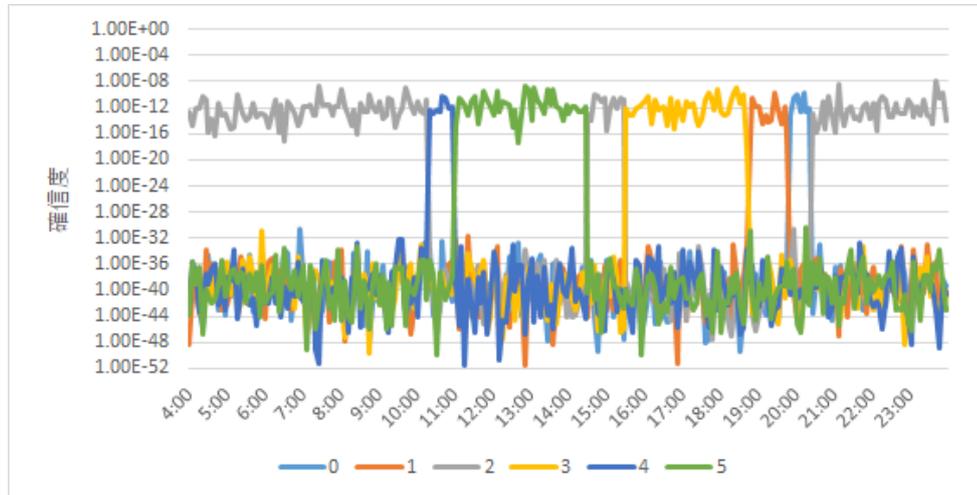


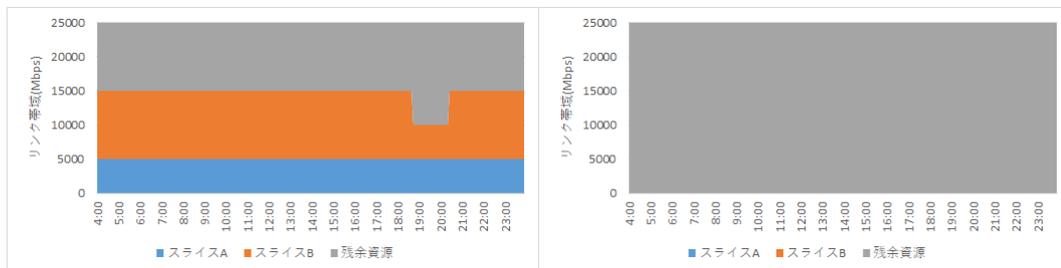
図 10 より、資源配分コントローラにより各時刻の状況に応じて各スライスへの割当資源量が変化していることが分かる。また、上述のように、昼ごろの時間帯はスライス B の通信量が、夕方はスライス A の通信量が増加するが、資源配分コントローラは、そのようなトラフィック増減に合わせて、割当資源を変更していることが分かる。

表 7 にスライス A、スライス B で発生したトラフィックのうち、割り当てられた資源では収容することができなかったトラフィック量を示す。

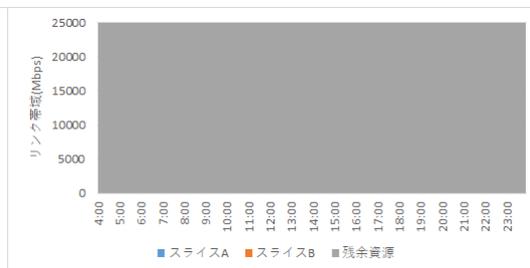
本評価では、資源配分コントローラにより各スライスに資源配分を行った後、スライス内のコントローラによって決められた経路にトラフィックを収容を試みた。しかしながら、適切に資源配分が行われなかったり、適切な経路が設定されない場合には、発生した通信のう

表 4: 選択枝別スライス A 配分資源量の候補値

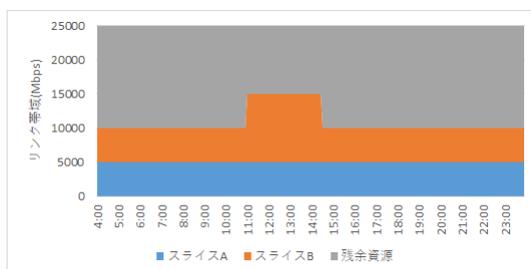
選択枝	1-2	1-4	2-1	2-3	3-2	3-4	4-1	4-3
0	5000	0	5000	10000	10000	5000	0	5000
1	5000	0	5000	10000	10000	5000	0	5000
2	5000	0	5000	10000	5000	5000	0	5000
3	5000	0	5000	10000	10000	5000	0	5000
4	5000	0	5000	5000	5000	5000	0	5000
5	5000	0	5000	5000	5000	5000	0	5000



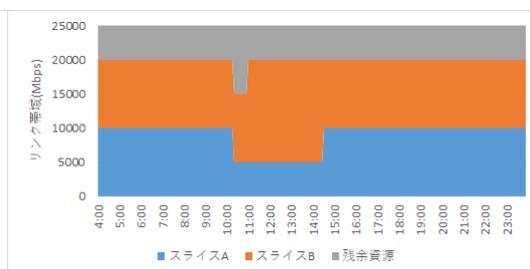
(a) リンク 1-2



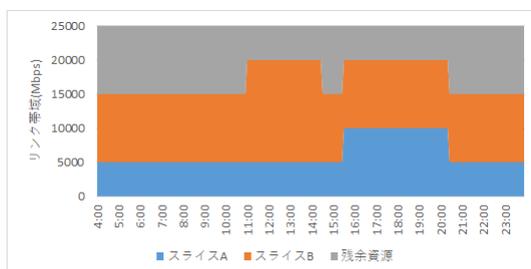
(b) リンク 1-4



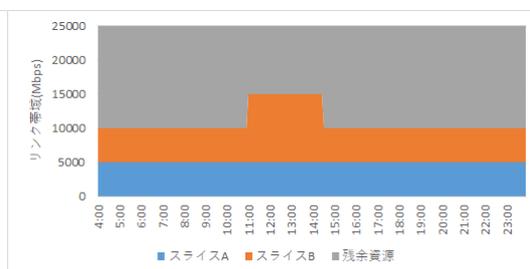
(c) リンク 2-1



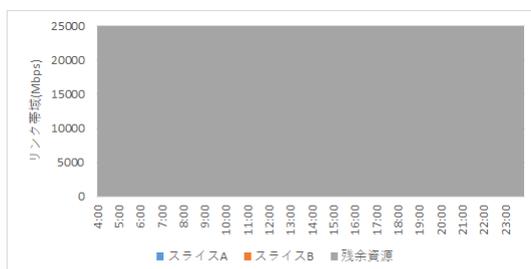
(d) リンク 2-3



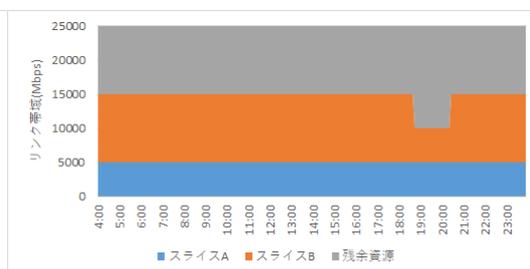
(e) リンク 3-2



(f) リンク 3-4



(g) リンク 4-1



(h) リンク 4-3

図 10: 各リンクの資源量遷移

表 5: 選択枝別スライス B 配分資源量の候補値

選択枝	1-2	1-4	2-1	2-3	3-2	3-4	4-1	4-3
0	5000	0	5000	10000	10000	5000	0	5000
1	5000	0	5000	10000	10000	5000	0	5000
2	10000	0	5000	10000	10000	5000	0	10000
3	10000	0	5000	10000	10000	5000	0	10000
4	10000	0	5000	10000	10000	5000	0	10000
5	10000	0	10000	15000	15000	10000	0	10000

表 6: 選択枝別スライス C 配分資源量の候補値

選択枝	1-2	1-4	2-1	2-3	3-2	3-4	4-1	4-3
0	15000	25000	15000	5000	5000	15000	25000	15000
1	15000	25000	15000	5000	5000	15000	25000	15000
2	10000	25000	15000	5000	10000	15000	25000	10000
3	10000	25000	15000	5000	5000	15000	25000	10000
4	10000	25000	15000	15000	10000	15000	25000	10000
5	10000	25000	10000	10000	5000	10000	25000	10000

ち、帯域が不足して、収容することができない。表では、この影響を評価するため、各時刻、各フローについて、収容される経路上の利用可能な資源量を確認し、不足する資源量を求めた。そして、その値をシミュレーション中の全時間で合計した値を記載している。表より、本評価では、スライス A、スライス B ともに資源不足は発生していないことが確認できる。これは、本手法の認知により、トラフィック量の増加に先駆けて、トラフィック量増加のリスクを把握し、適切な資源割当、経路制御を行うことができているためである。

表 7: 収容できなかったトラヒック量

時間	スライス A	スライス B
4時	0	0
5時	0	0
6時	0	0
7時	0	0
8時	0	0
9時	0	0
10時	0	0
11時	0	0
12時	0	0
13時	0	0
14時	0	0
15時	0	0
16時	0	0
17時	0	0
18時	0	0
19時	0	0
20時	0	0
21時	0	0
22時	0	0
23時	0	0

## 5 まとめ

我々の研究グループでは、人の脳が曖昧・不完全な観測情報をもとに認知を行う仕組みをモデル化した Bayesian Attractor Model にもとづいた状況認知を行うことにより、現実世界のセンシングデータ、トラヒック量の観測情報の両方を用いて、将来発生する資源不足のリスクを把握する手法に関する検討を進めており、これまでに、Bayesian Attractor Model にもとづいた状況認知を、モバイルネットワークにおける特定のエリアから流入するトラヒック量の予測に適用し、トラヒック量増加するリスクを事前に把握できることを示している。

本報告では、Bayesian Attractor Model にもとづいた状態認知を、各スライスにおけるトラヒック量の予測・制御に用いるのみではなく、ネットワークスライス間の資源割当に適用した。各スライスにおけるトラヒック変動のリスクについては、当該スライスに関連する現実世界のセンシングデータとトラヒック量の観測情報をもとに、把握することができる。しかしながら、ネットワークスライス間の資源割り当てを行うためには、各ネットワークにおける資源不足のリスクを把握するのみではなく、全ネットワークスライスの状況を把握することが必要となる。しかし、いずれかのネットワークスライスの状況の把握に有用な情報をすべて収集すると、資源割当の制御を行うコントローラが収集する情報が膨大となり、状態の把握に必要な情報の収集が困難となる。

そのため、本報告では、各スライスにおける状態判断は各スライスに対応する制御を行うコントローラで行い、スライス間の資源割当を行うコントローラでは、各スライスにおける判断結果を収集した上で、ネットワーク全体の情報を判断し、資源割当の制御を行うといった階層的な判断を行う手法を提案した。本報告では、提案する階層的な判断にもとづくネットワークスライス制御を実装し、その動作を確認する。その結果、階層的な認知により、資源割当を行うコントローラにおいて、ネットワーク全体の状況を把握し、各スライスが求める通信性能を維持できるように資源割当を行うことができることを示した。

本報告の評価では、各スライスに割り当てられる資源は、リンクの帯域のみであり、各スライスにおける制御も経路制御のみとした単純なモデルで認知を行った。しかしながら、本報告で提案している脳の認知機能モデルを用いたスライスへの資源割当制御は、計算機資源のスライスへの割当など、他の資源の割当にも適用できる。今後は、計算機資源など、多種の資源をスライスに割り当てる環境下など、多種の環境において、脳の認知機能モデルを用いたスライスへの資源割当制御の評価を行う予定である。

## 謝辞

本研究を進めるにあたってご指導、ご鞭撻いただきました大阪大学大学院情報科学研究科の村田正幸教授に深く感謝申し上げます。また、本研究の方針を含め、様々な疑問に真摯に向き合い、適切なお助言およびご指導をいただきました大阪大学大学院情報科学研究科の天下裕一准教授に心より感謝申し上げます。また、平素から広くご指導いただきました大阪大学大学院情報科学研究科の荒川伸一准教授、大阪大学大学院経済学研究科の小南大智助教に心からお礼申し上げます。また、本報告にて用いました、Bayesian Attractor Model のプログラムを構築していただき、多くの質問に答えてくださいました、大阪大学大学院情報科学研究科の大歳達也特任助教に厚く御礼申し上げます。最後に、日頃より様々な面で支えていただきました佐竹幸大氏、山内雅明氏をはじめ、村田研究室の皆様にご感謝の意を表して謝辞といたします。

## 参考文献

- [1] X. Foukas, G. Patounas, A. Elmokashfi, and M. K. Marina, “Network Slicing in 5G: Survey and Challenges,” *IEEE Communications Magazine*, vol. 55, pp. 94–100, May 2017.
- [2] H. Zhang, N. Liu, X. Chu, K. Long, A. H. Aghvami, and V. C. M. Leung, “Network Slicing Based 5G and Future Mobile Networks: Mobility, Resource Management, and Challenges,” *IEEE Communications Magazine*, vol. 55, pp. 138–145, Aug. 2017.
- [3] X. Zhou, R. Li, T. Chen, and H. Zhang, “Network Slicing as a Service: Enabling Enterprises’ Own Software-Defined Cellular Networks,” *IEEE Communications Magazine*, vol. 54, pp. 146–153, July 2016.
- [4] P. Kumar, Y. Yuan, C. Yu, N. Foster, R. Kleinberg, and R. Soul, “Kulfi: Robust Traffic Engineering Using Semi-Oblivious Routing,” *arXiv preprint arXiv:1603.01203*, Mar. 2016.
- [5] M. Chiesa, G. Kindler, and M. Schapira, “Traffic Engineering With Equal-Cost-MultiPath: An Algorithmic Perspective,” *IEEE/ACM Transactions on Networking*, vol. 25, pp. 779–792, Apr. 2017.
- [6] H. H. Liu, S. Kandula, R. Mahajan, M. Zhang, and D. Gelernter, “Traffic Engineering with Forward Fault Correction,” *ACM SIGCOMM Computer Communication Review*, vol. 44, no. 4, pp. 527–538, 2015.
- [7] I. F. Akyildiz, A. Lee, P. Wang, M. Luo, and W. Chou, “A Roadmap for Traffic Engineering in SDN-OpenFlow Networks,” *Computer Networks*, vol. 71, pp. 1–30, Oct. 2014.
- [8] 佐竹 幸大 , 大下 裕一 , 村田 正幸, “人の脳情報処理プロセスに着想を得た実世界情報協調予測型トラヒックエンジニアリング手法,” *Technical Reports of IEICE (IN2017-93)*, mar 2018.
- [9] B. N. Astuto and et al, “A survey of Software-Defined Networking: Past, Present, and Future of Programmable Networks,” *IEEE Communications Surveys and Tutorials*, vol. 16, no. 3, pp. 1617–1634, 2014.

- [10] H. Hassan and et al, “NFV: state of the art, challenges, and implementation in next generation mobile networks (vEPC),” *IEEE Network*, vol. 28, no. 6, pp. 18–26, 2014.
- [11] S. Rob and et al, “Flowvisor: A Network virtualization layer,” *OpenFlow Switch Consortium*, Oct. 2009.
- [12] B. Sushil, L. Jain, and S. Jain, “Cloud Computing: A Study of Infrastructure as a Service (IAAS),” *International Journal of engineering and information Technology*, vol. 2, pp. 60–63, Jan. 2010.
- [13] A. Mohammad and E.-N. Huh, “Dynamic Resource Provisioning Through Fog Micro Datacenter,” *IEEE International Conference*, 2015.
- [14] C. Liang and F. R. Yu, “Wireless Virtualization For Next Generation Mobile Cellular Networks,” *IEEE Wireless Communications*, vol. 22, pp. 61–69, Feb. 2015.
- [15] S. Bitzer, J. Bruineberg, and S. J. Kiebel, “A Bayesian Attractor Model for Perceptual Decision Making,” *PLOS Computational Biology*, vol. 11, Aug. 2015.
- [16] T. Kashiyama, Y. Pang, and Y. Sekimoto, “Open pflow: Creation and Evaluation of an Open Dataset for Typical People Mass Movement in Urban Areas,” *Transportation Research Part C: Emerging Technologies*, vol. 85, pp. 249–267, Dec. 2017.
- [17] 総務省 総合通信基盤局 電気通信事業部 データ通信課, “我が国のインターネットにおけるトラヒックの集計結果,” Aug. 2018.