SFC制御自動化のための階層型AIアーキテクチャによるサービス機能チェイン経路決定手法

平山 孝弘　地引 昌弘　原井 洋明
情報通信研究機構(NICT)

*** 本研究の一部は、総務省委託研究「革新的AIネットワーク統合基盤技術の研究開発（JPMI00316）」による支援を受けて実施している。**

Takahiro Hirayama [hirayama@nict.go.jp]
©National Institute of Information and Communication Technology

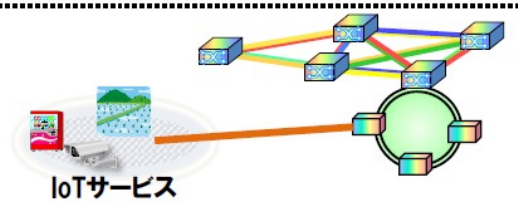
発表概要

- **背景-ネットワーク仮想化(NFV)/サービス機能チェイン(SFC)基盤の制御の自動化**
- サービス機能チェイン(SFC)基盤制御への機械学習活用
  - SFC資源割当自動制御の概要
  - SFCマイグレーション(移行)スケジューリングのための機械学習モデル
    - Encoder-Decoder Recurrent Neural Network (ED-RNN)
  - 階層型AIアーキテクチャによるサービス機能チェイン経路決定手法

5G時代のスライシング

＜IoTサービス＞

あらゆるモノがインターネットにつながることで、新たな付加価値をもたらすサービス。



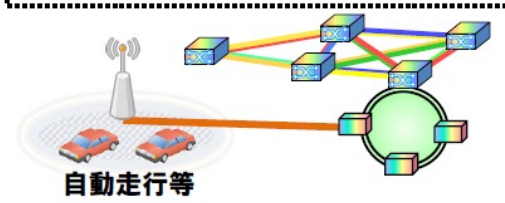
ユースケースの例

- 農業、インフラ管理（現在のモバイル網の通信エリア外での通信も含む。） 等

要求条件： 柔軟化・高弾力化、高効率データ流通、安全・信頼性

＜超リアルタイムサービス＞

自動走行等の高いリアルタイム性を必要とするサービス。



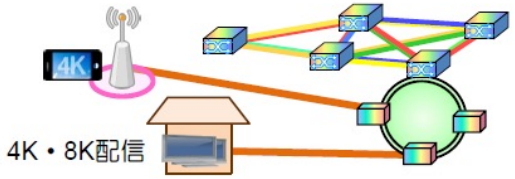
ユースケースの例

- 市街地での自動走行（都市部/郊外地域） 等

要求条件： 超低遅延、安全・信頼性、大容量(アクセス)、省電力化、柔軟化・高弾力化、高効率データ流通

＜高精細映像配信サービス＞

4K・8K等の高精細な映像を、ネットワークを介して高品質で配信するサービス。



ユースケースの例

- 一般宅内でのコンテンツ同時視聴・VoD配信
- 屋外・移動中の視聴者へのコンテンツ配信 等

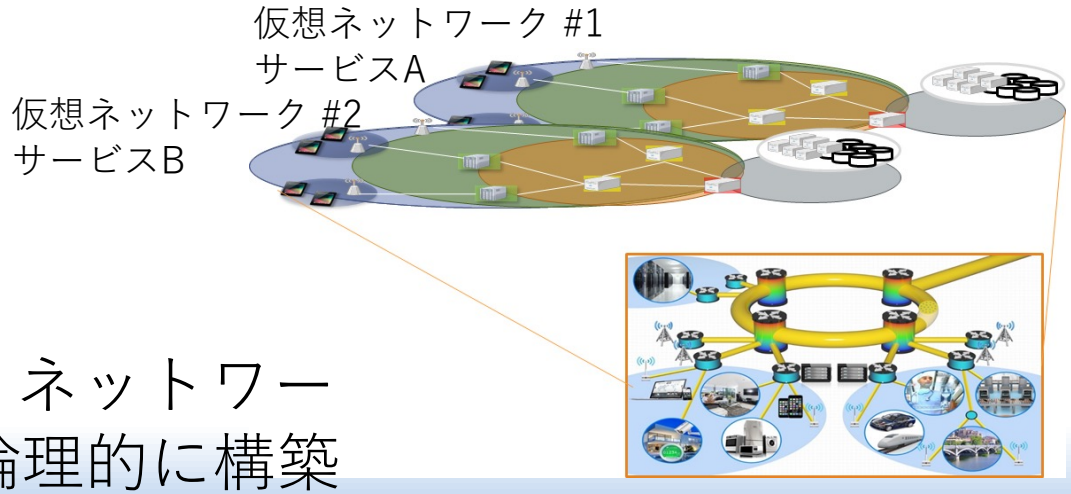
要求条件： 大容量（コア・アクセス）、省電力化、柔軟化・高弾力化、高効率データ流通、安全・信頼性

※ 総務省 将来のネットワークインフラに関する研究会資料（2017.5.26）より抜粋

多種多様な QoS 要求

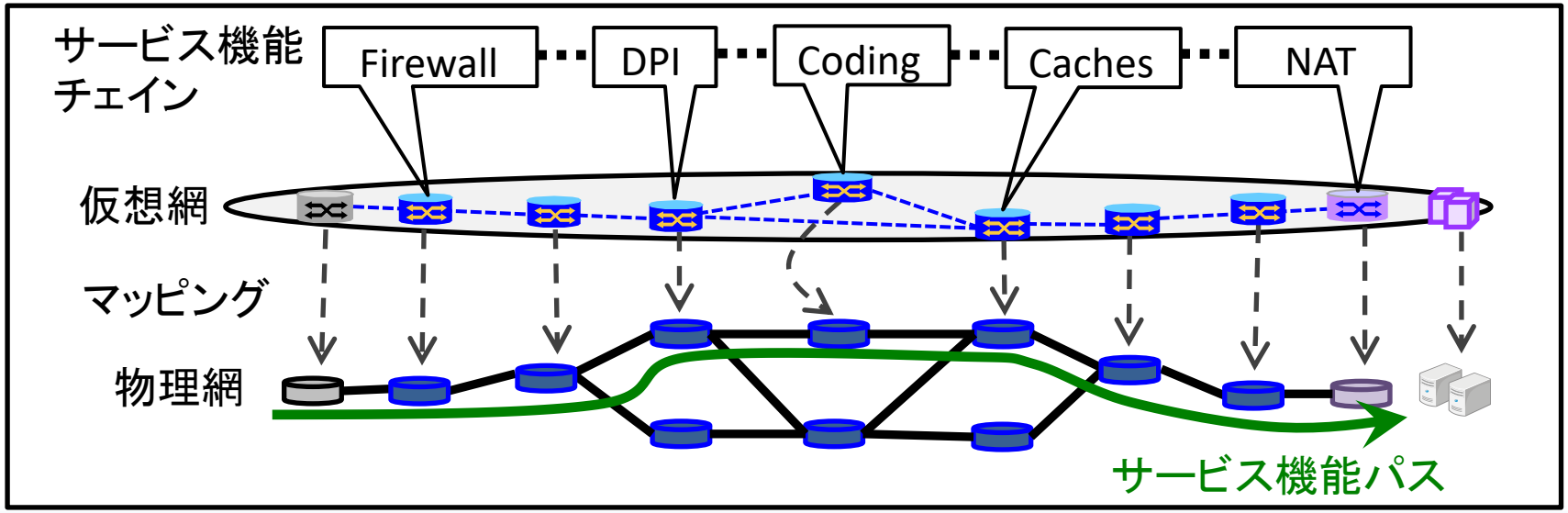
ネットワーク仮想化技術

それぞれのQoS要求に合うネットワークを、同じ物理基盤上で論理的に構築



背景　～ NFV and SFC ～

- Network Function Virtualization (NFV)：
  - 複数のネットワーク機能(NFs)を汎用サーバからなる基盤上でソフトで実装
- Service Function Chaining (SFC)：
  - 必要な複数のNFs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内処理を施すための仮想的なサービス機能チェインを構築
    →NFsを柔軟に分散配置でき、エンドユーザ毎に適時適切なサービスを迅速に提供可能
    - 参考： IETF RFC 7665 / RFC 8300, ETSI GS NFV-EVE 005 V1.1.1

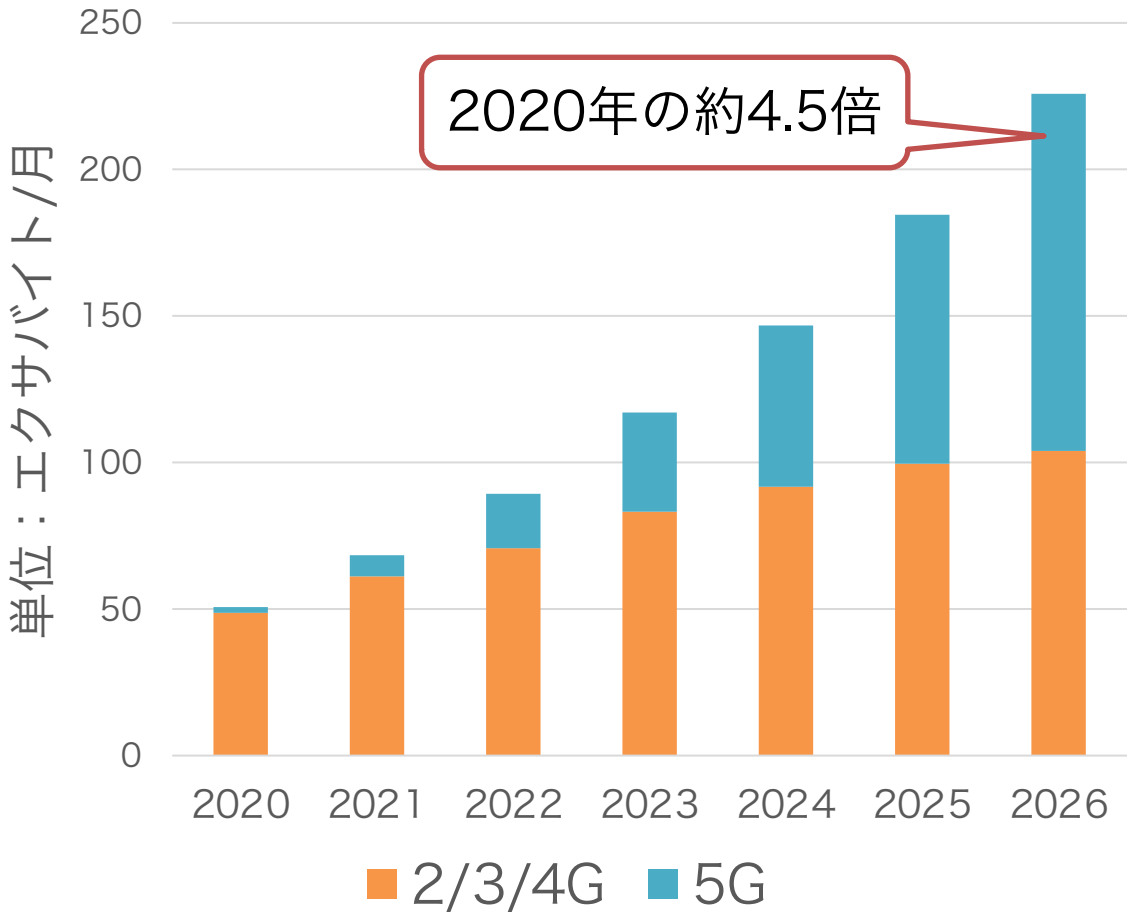


* 資源：コンピュータ資源（CPU、メモリ、ストレージ容量、ソフトウェア基盤）

背景　～トラヒック量と接続端末数の増加～

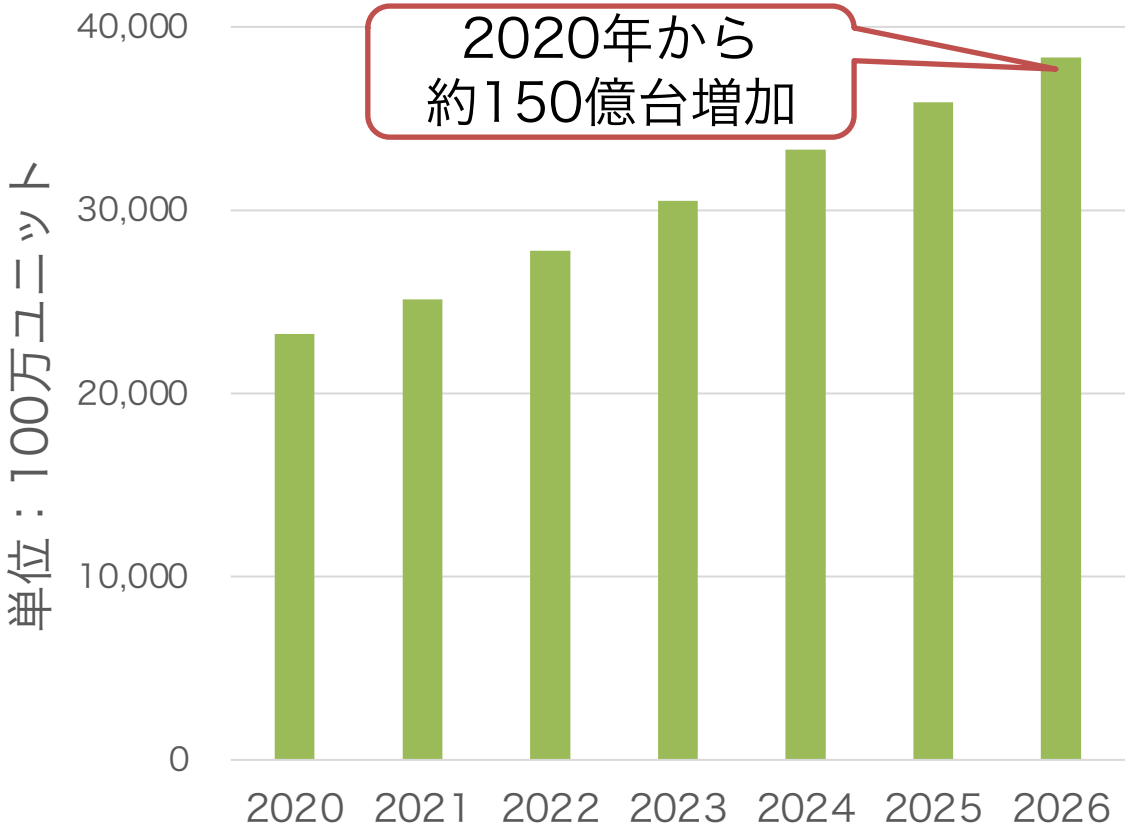
モバイルデータトラフィック量の予測

(出典: Ericsson Mobility Visualizer, 2020 Nov.)

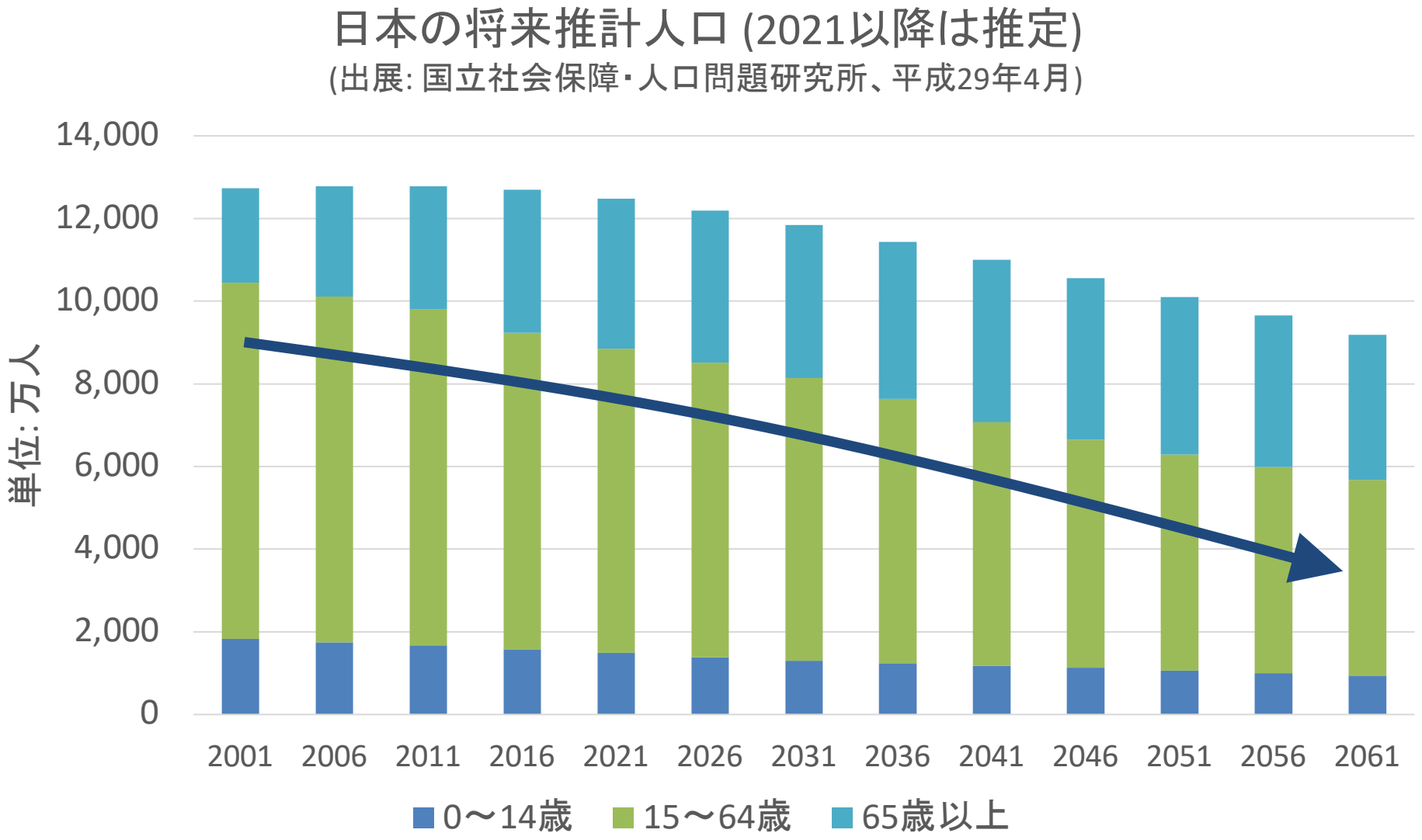


接続端末(モバイル,IoT)数の予測

(出典: Ericsson Mobility Visualizer, 2020 Nov.)



背景 ～求められる自動化～



進む人口減少

AIによる運用補助、
ひいては自動化が必要

サービスネットワーク基盤制御 全体的な課題

- 資源利用状況変動や障害発生時にも、 5G時代における多種多様なサービス品質要求を満たすことが必要
  - 既存手法（例：手動操作や最適化手法等）では、処理に時間がかかり、環境変動に追従できない
  - 各サービスを提供するための計算資源が不足し、サービス品質が劣化

- **サービス品質の維持**と有限な計算**資源の有効利用**を図りつつ、**環境変動に追従する高速な計算資源自動制御機構**を備えた、サービス機能チェイニング基盤（データ・サービス・制御・管理 各々プレーン）が必要

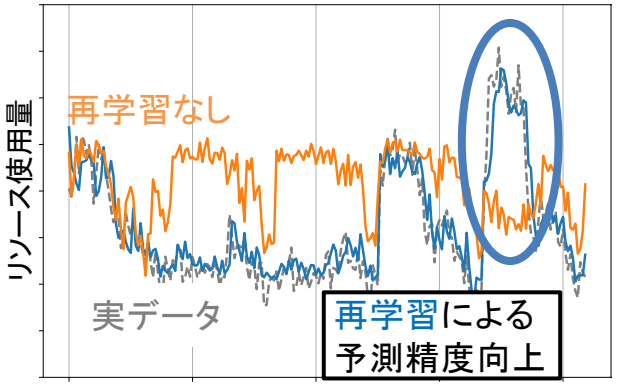
機械学習活用を含め、**AI**サービスネットワーク
基盤制御技術の研究開発の取組みを紹介

発表概要

- 背景-ネットワーク仮想化(NFV)/サービス機能チェイン(SFC)基盤の制御の自動化
- **サービス機能チェイン(SFC)基盤制御への機械学習活用**
  - **SFC資源割当自動制御の概要**
  - SFCマイグレーション(移行)スケジューリングのための機械学習モデル
    - Encoder-Decoder Recurrent Neural Network (ED-RNN)
  - 階層型AIアーキテクチャによるサービス機能チェイン経路決定手法

SFC基盤自動制御技術の検討

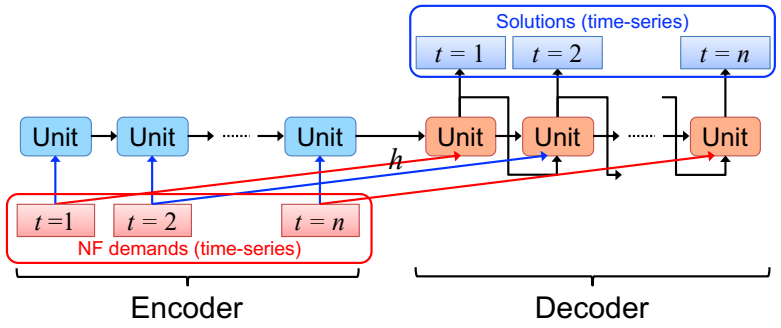
予測ベースの調停



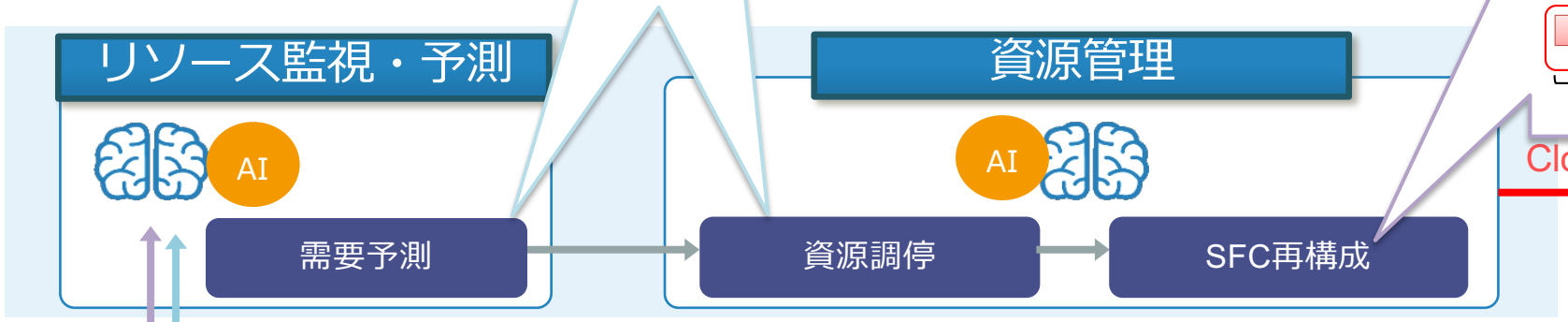
SF移行計画の立案

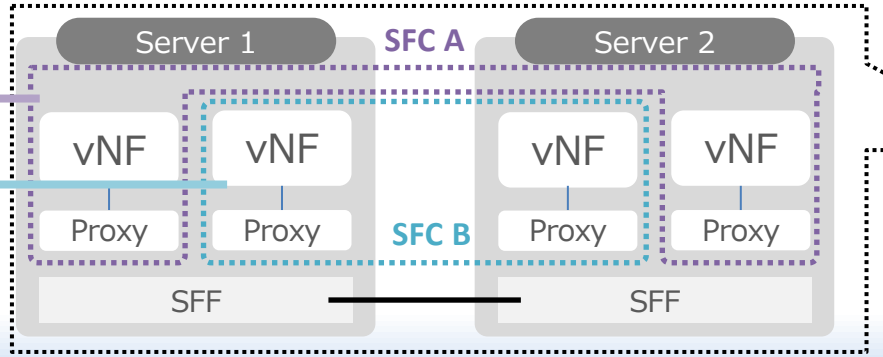
Encoder-Decoder RNNによる
移行時期・移行先決定

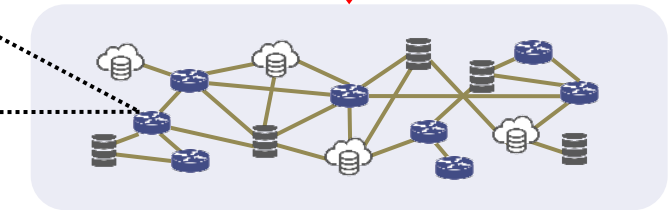
- 需要の時系列データを分析
- いつ・どこへSFを移行させるかを判断



Closed-loop control







ネットワーク基盤

発表概要

- 背景-ネットワーク仮想化(NFV)/サービス機能チェイン(SFC)基盤の制御の自動化
- **サービス機能チェイン(SFC)基盤制御への機械学習活用**
  - SFC資源割当自動制御の概要
  - **SFCマイグレーション(移行)スケジューリングのための機械学習モデル**
    - **Encoder-Decoder Recurrent Neural Network (ED-RNN)**
  - 階層型AIアーキテクチャによるサービス機能チェイン経路決定手法

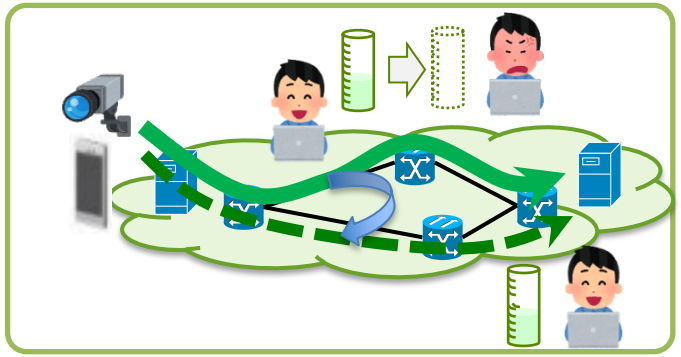
想定される課題と解決策

サービスを実現する機能チェインの再構成回数を最小化し、処理を高速化

サービス機能チェインのイメージ
（例：映像配信サービス）

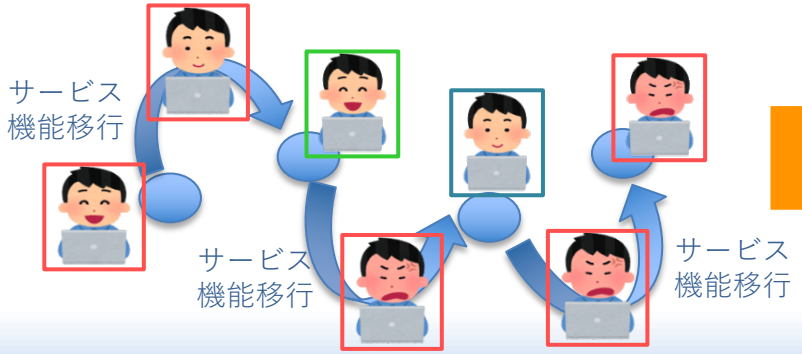
チェイン再構成のイメージ

資源融通と実行場所変更



従来手法

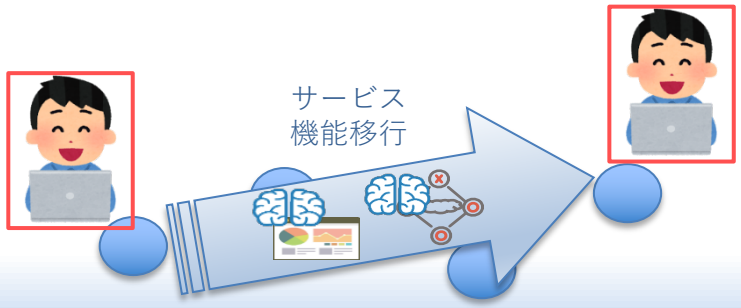
人が経験で解けない問題に対し、
- 逐次の最適解 ⇒ 再構成回数が多い
- 未来を見据えた最適解 ⇒ 1時間から1日以上必要



本研究開発

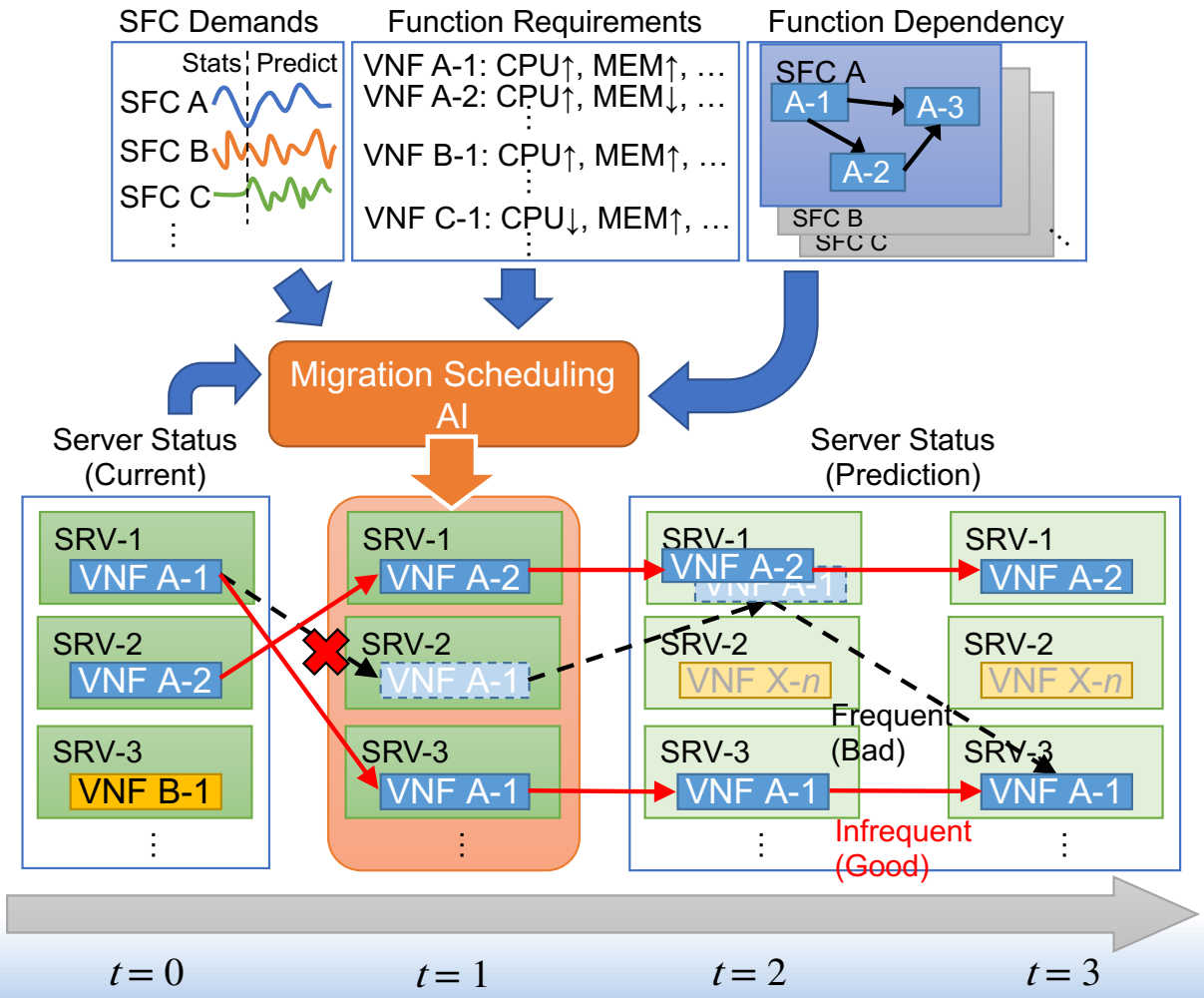
EDRNNでは、
- 時系列分析に基づく再構成回数最小
- 秒単位で、サービス機能移行先を決定

を同時に実現



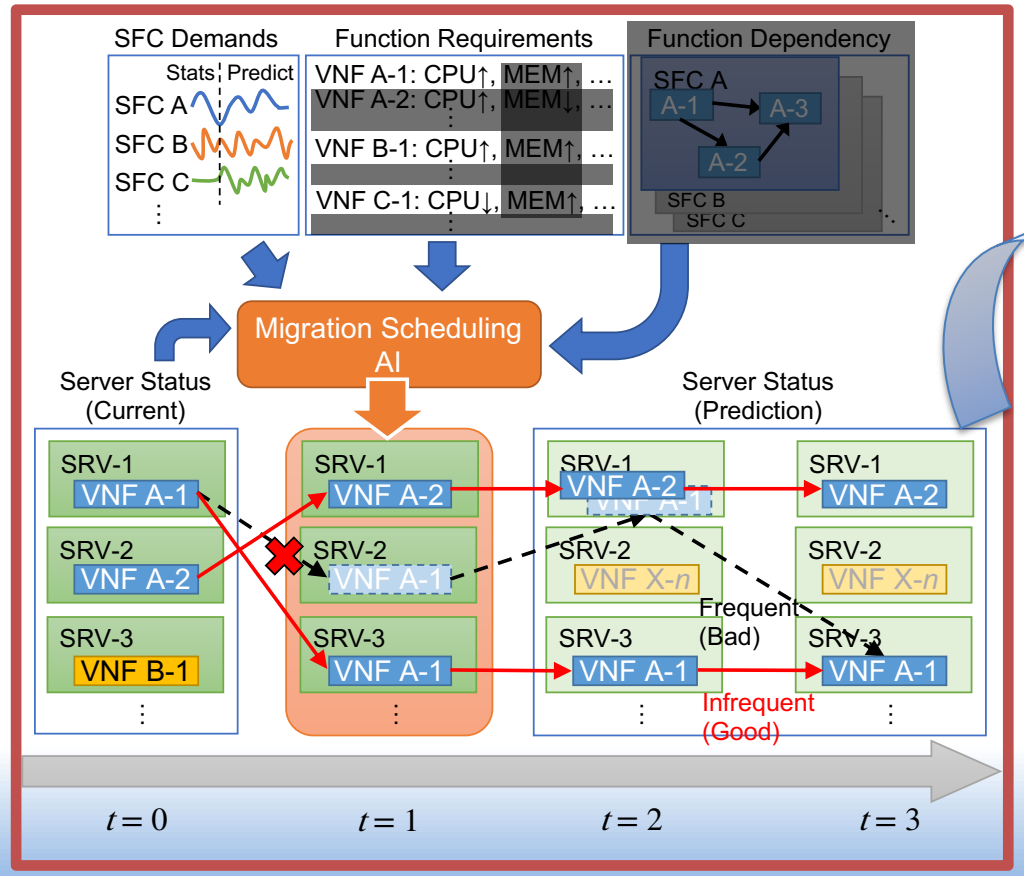
AIによるSF移行スケジューリング (目標)

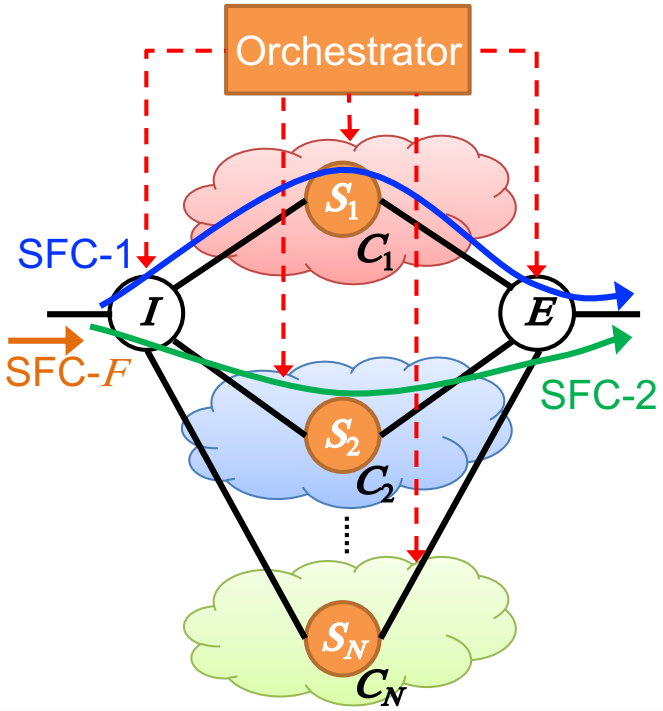
- サーバの資源量、SFチェインの需要変動、VNFのリソース要求量などを参照して、VNFの移行先&移行時期をすばやく(数秒程度で)決定
  - 数ステップ先を見据えて、同じVNFの移行が頻繁に起きないように配慮



SFチェイン移行スケジューリング問題の簡略化

- スケジューリング問題を下記のように簡略化して、AIのモデルの選定、チューニング
  - 各SFチェインを構成するVNFの中の一つに着目
    - リソース要求はCPUのみ、VNF間の依存関係も考慮せず
  - 時刻ごとに各SFチェインの要求するリソース量が変動する中で、複数あるパスのどのパスにどのSFチェインを通すか?

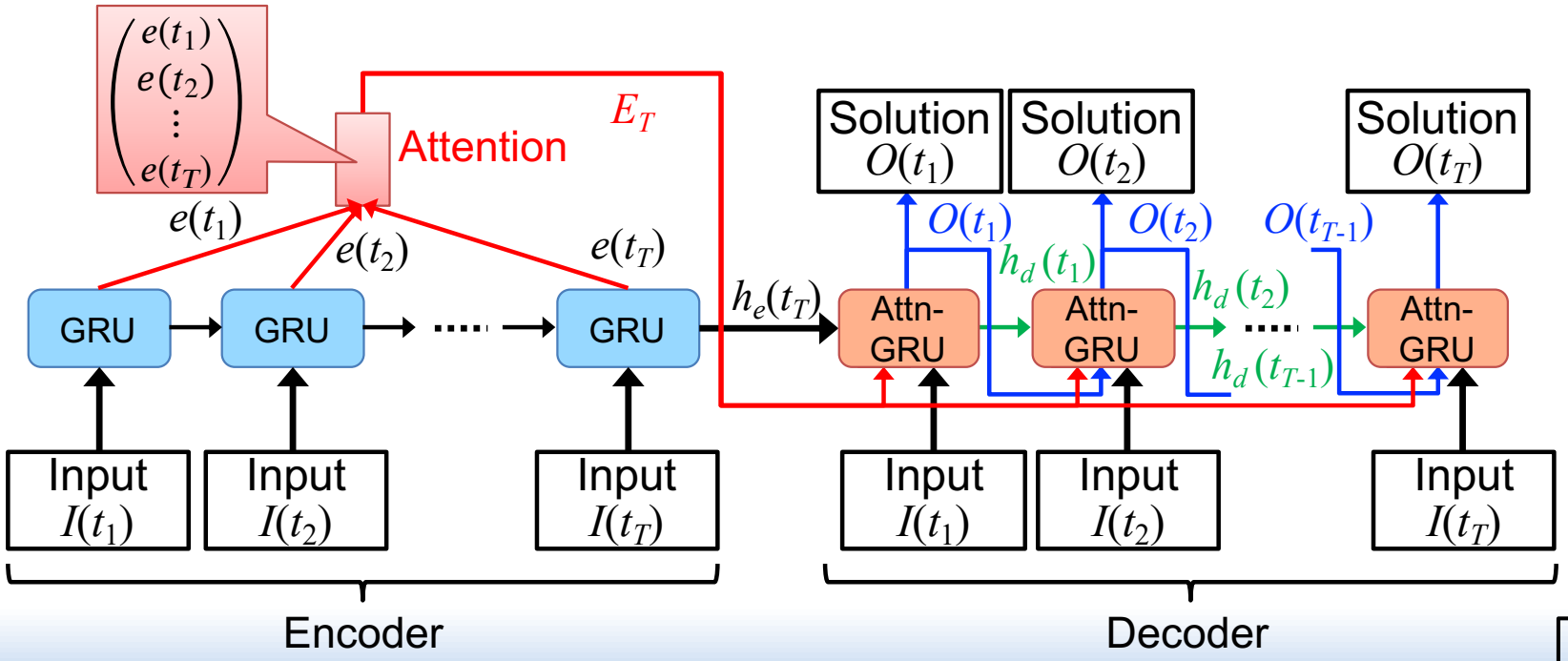




F本のSFチェインが通るパスを時刻ごとに振り分け

エンコーダ・デコーダRNN(ED-RNN)の構成と仕組み

- 時刻t_1から$t_T(T=10)$までの入力をEncoder, Decoderで2度読み込む
  - RNNを構成するセルにはGRU(Gated Recurrent Unit)を使用
- Encoderの最終セル(t_T)の隠れ層出力($h_e(t_T)$)に加え、各時刻の出力(Attention)を記憶し、Decoderへ渡す
  - Attentionを導入し、時刻t_1からt_Tの中で重要な時刻がいつかを決定する
- Decoderは前時刻の出力を参照し次状態を決定するので、前時刻との差分(移行回数)を低くすることが可能になる



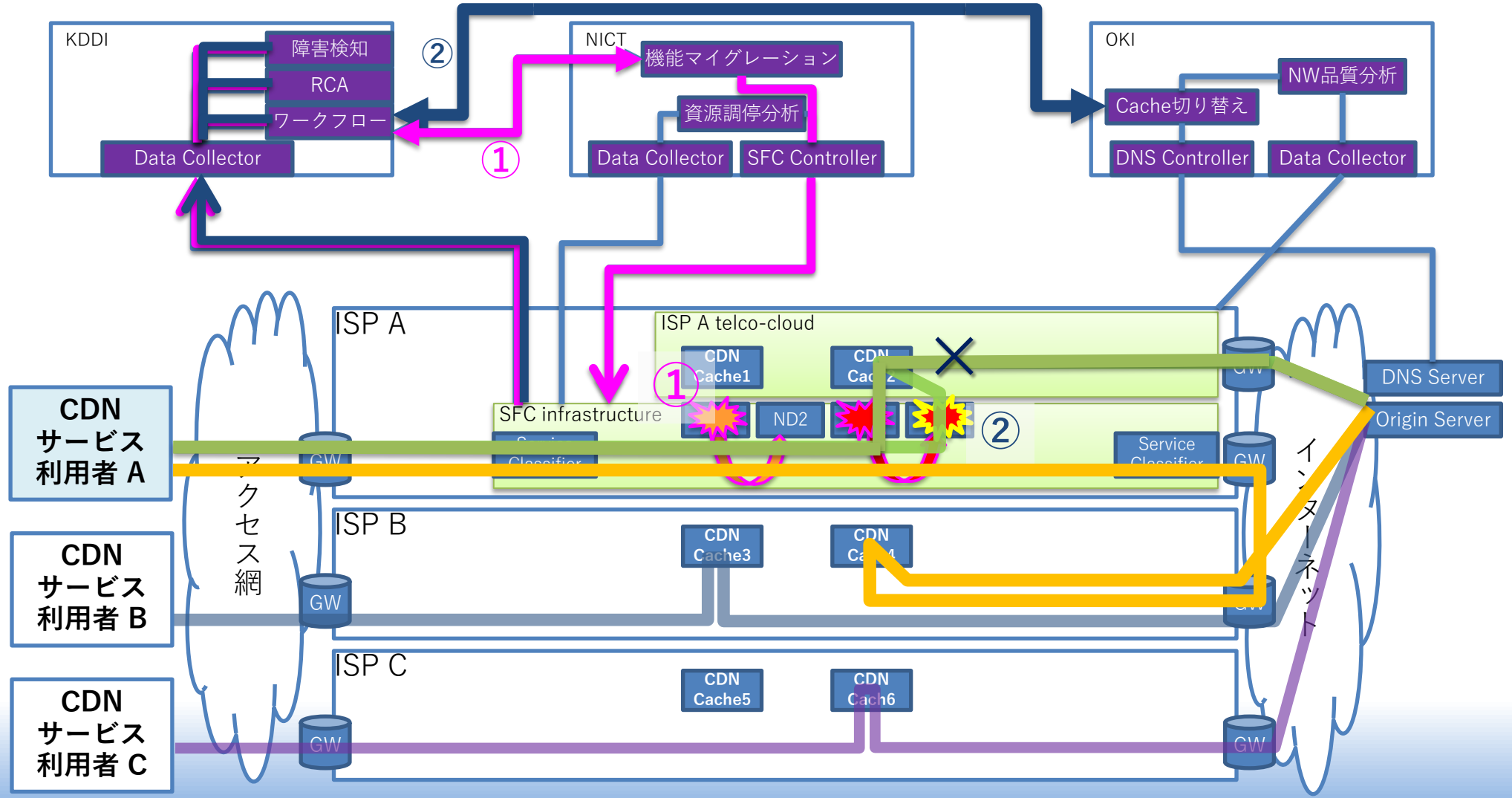
Attn-GRUの構造は割愛
(補足1に記載)

外部サービスとの連携実験

◆ AIを用いたSFC基盤の資源制御により、キャッシュ制御をサポート

① SF 網内にてEDRNNにより移行先を導出 ⇒ 外部への影響なし

② SF 網内だけで解決不能な場合は品質分析・ワークフローAIと連携 ⇒ 外部への影響を最小化



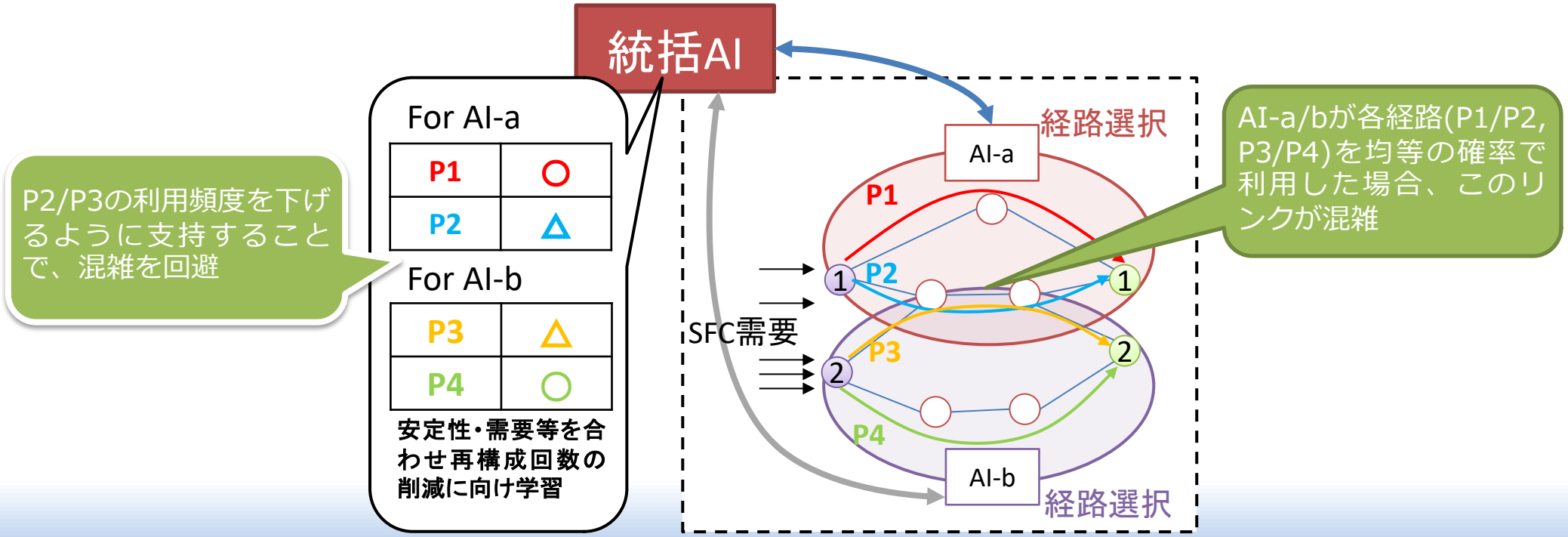
発表概要

- 背景-ネットワーク仮想化(NFV)/サービス機能チェイン(SFC)基盤の制御の自動化
- **サービス機能チェイン(SFC)基盤制御への機械学習活用**
  - SFC資源割当自動制御の概要
  - SFCマイグレーション(移行)スケジューリングのための機械学習モデル
    - Encoder-Decoder Recurrent Neural Network (ED-RNN)
  - **階層型AIアーキテクチャによるサービス機能チェイン経路決定手法**

階層型AIアーキテクチャによるサービス機能チェイン経路決定

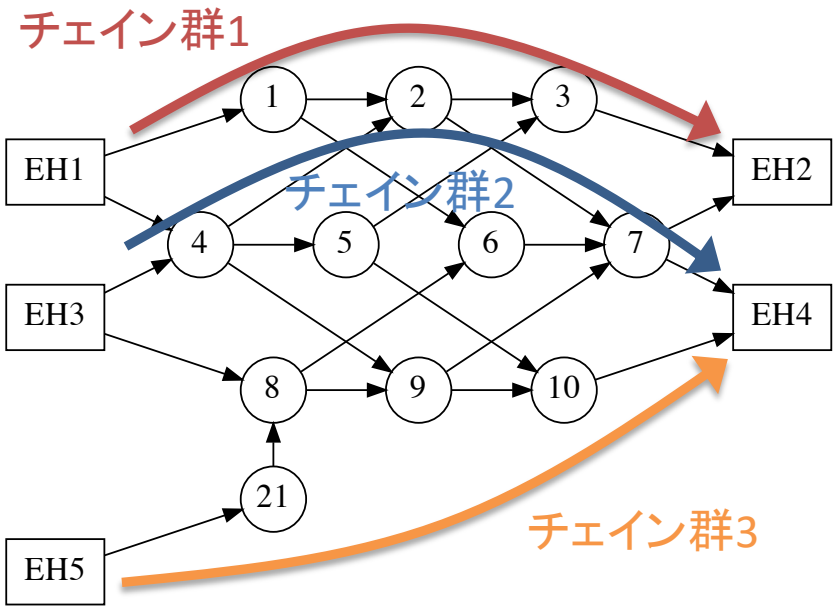
- 先述のVNF移行スケジューリングは、同じIngress/Egressノードペア間を流れるSFCを、どの経路に割り当てるかを判断するもの
- Ingress/Egressペア毎にスケジューリングAIを配備する場合、それぞれが利己的にSFCの経路割当を行うと、特定のノードやリンクに負荷が集中し性能が著しく低下してしまう怖れ

全体を統括するAI(統括AI)が各スケジューリングAIに経路の推奨度を提示することで、特定のノード/リンクへの負荷集中を避ける



統括AIの方針

- トポロジーの分析、機械学習によるトラヒック需要予測を組み合わせて経路の推奨度を定義
- 左下のトポロジーにおいて、EH1→EH2、EH3→EH4、EH5→EH4へ向かうSFCを考える
- 各チェイン群の最短経路の数は、6本、7本、10本 (計23本)



チェイン群1の経路候補

1. [EH3, 4, 2, 3, EH2]
2. [EH3, 4, 5, 3, EH2]
3. [EH3, 4, 2, 7, EH2]
4. [EH3, 4, 9, 7, EH2]
5. [EH3, 8, 9, 7, EH2]
6. [EH3, 8, 6, 7, EH2]

チェイン群2の経路候補

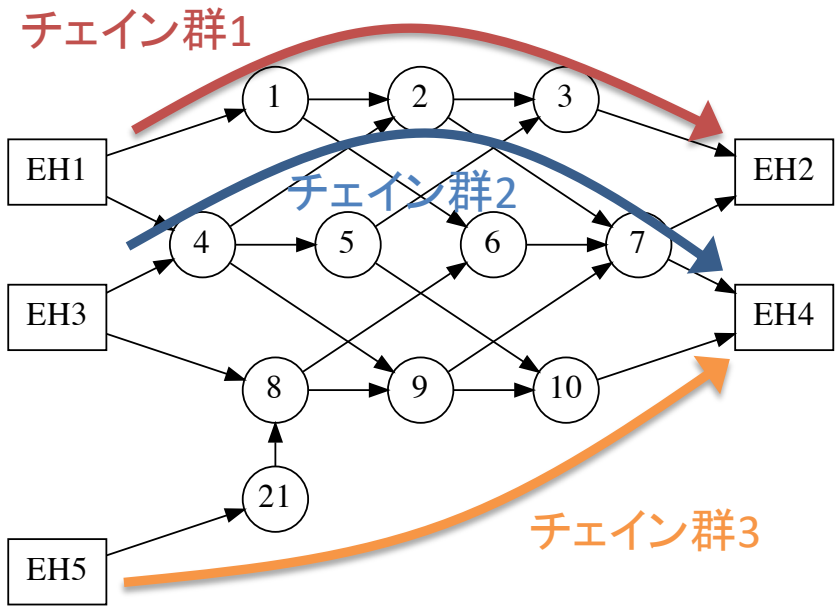
1. [EH3, 4, 2, 7, EH4]
2. [EH3, 4, 9, 7, EH4]
3. [EH3, 8, 9, 7, EH4]
4. [EH3, 8, 6, 7, EH4]
5. [EH3, 4, 5, 10, EH4]
6. [EH3, 4, 9, 10, EH4]
7. [EH3, 8, 9, 10, EH4]

チェイン群3の経路候補

1. [EH5, 21, 8, 6, EH4]
2. [EH5, 21, 8, 7, EH4]
3. [EH5, 21, 6, 7, EH4]
4. [EH5, 8, 6, 7, EH4]
5. [EH5, 21, 8, 9, EH4]
6. [EH5, 21, 9, 7, EH4]
7. [EH5, 8, 9, 7, EH4]
8. [EH5, 21, 8, 10, EH4]
9. [EH5, 21, 9, 10, EH4]
10. [EH5, 8, 9, 10, EH4]

統括AI　～トポロジー分析～

- トポロジーの分析、機械学習によるトラヒック需要予測を組み合わせて経路の推奨度を定義
- 左下のトポロジーにおいて、EH1→EH2、EH3→EH4、EH5→EH4へ向かうSFCを考える
- 各チェイン群の最短経路の数は、6本、7本、10本 (計23本)



チェイン群1の経路候補

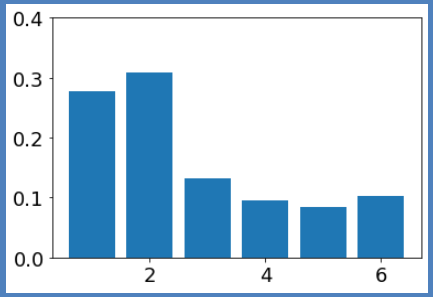
1. [EH3, 4, 2, 3, EH2]
2. [EH3, 4, 5, 3, EH2]
3. [EH3, 4, 2, 7, EH2]
4. [EH3, 4, 9, 7, EH2]
5. [EH3, 8, 9, 7, EH2]
6. [EH3, 8, 6, 7, EH2]

チェイン群2の経路候補

1. [EH3, 4, 2, 7, EH4]
2. [EH3, 4, 9, 7, EH4]
3. [EH3, 8, 9, 7, EH4]
4. [EH3, 8, 6, 7, EH4]
5. [EH3, 4, 5, 10, EH4]
6. [EH3, 4, 9, 10, EH4]
7. [EH3, 8, 9, 10, EH4]

経路1のノード(4,2,3)は他の経路にそれぞれ7,2,1回出現(平均**3.3**回)

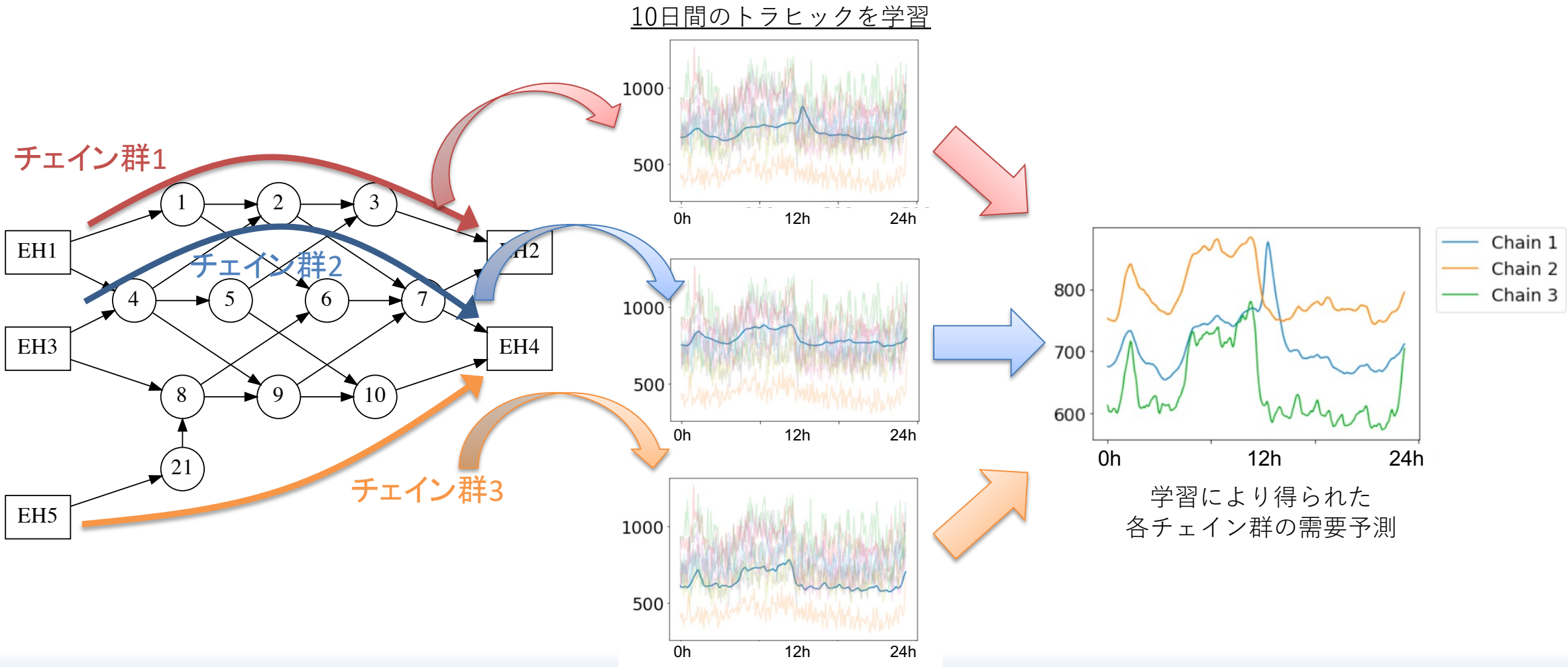
経路1-6の平均出現回数はそれぞれ、3.3, 3, 7, 9.66, 11, 9回



逆数を正規化した値を各経路の基準推奨度と定義

統括AI　～機械学習に基づくトラヒック需要予測～

- チェイン群(Ingress/Egressペア)毎に10日間のトラヒック流量を学習(ガウス過程回帰)させ、
→各チェインの時間帯ごとの需要の予測に用いる

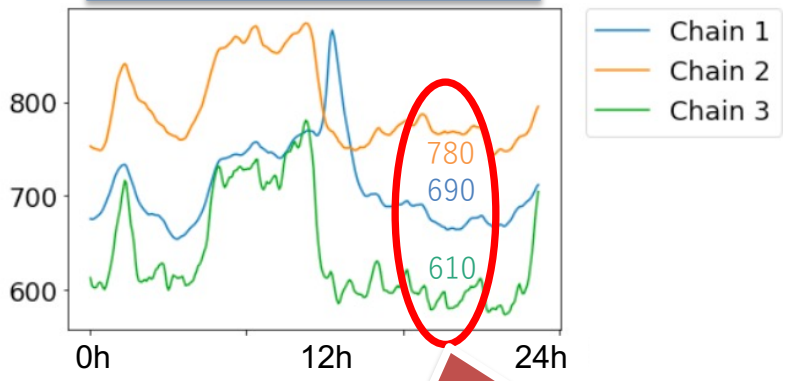


トポロジー分析と機械学習を組み合わせた推奨度の算出

- 統括AIは基準推奨度とチェイン群の需要予測を各経路選択AIに配布

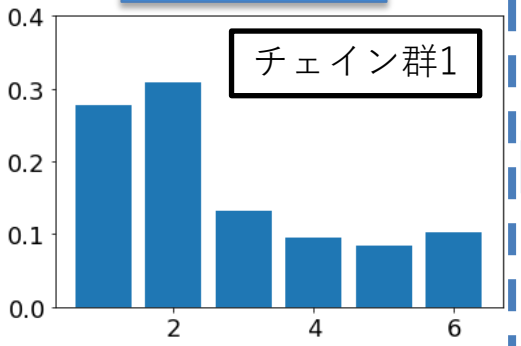
統括AIから経路選択AIへ送るデータ

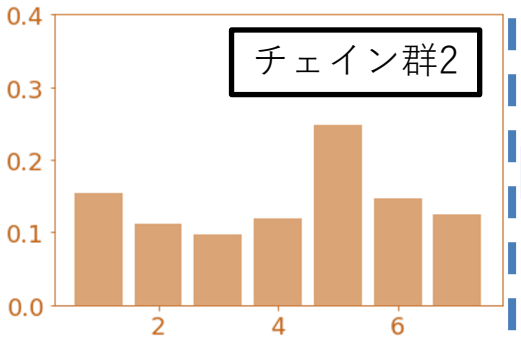
各チェイン群の需要予測



この時間帯に着目すると、
チェイン群1/2/3の需要は、
690/780/610
(中:高:低)

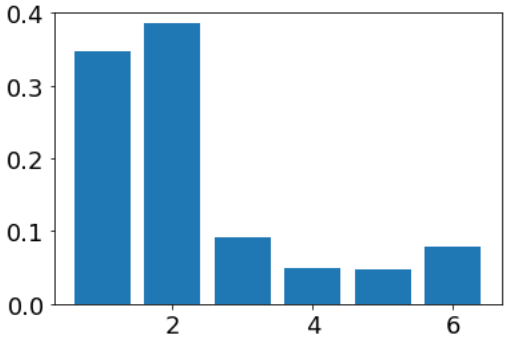
基準の推奨度

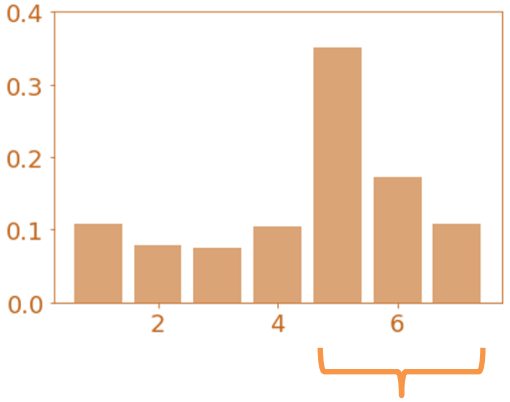




チェイン群3
(割愛)

時刻ごとの各経路の推奨度を経路選択AIが判断





チェイン群1の経路候補

1. [EH3, 4, 2, 3, EH2]
2. [EH3, 4, 5, 3, EH2]
3. [EH3, 4, 2, 7, EH2]
4. [EH3, 4, 9, 7, EH2]
5. [EH3, 8, 9, 7, EH2]
6. [EH3, 8, 6, 7, EH2]

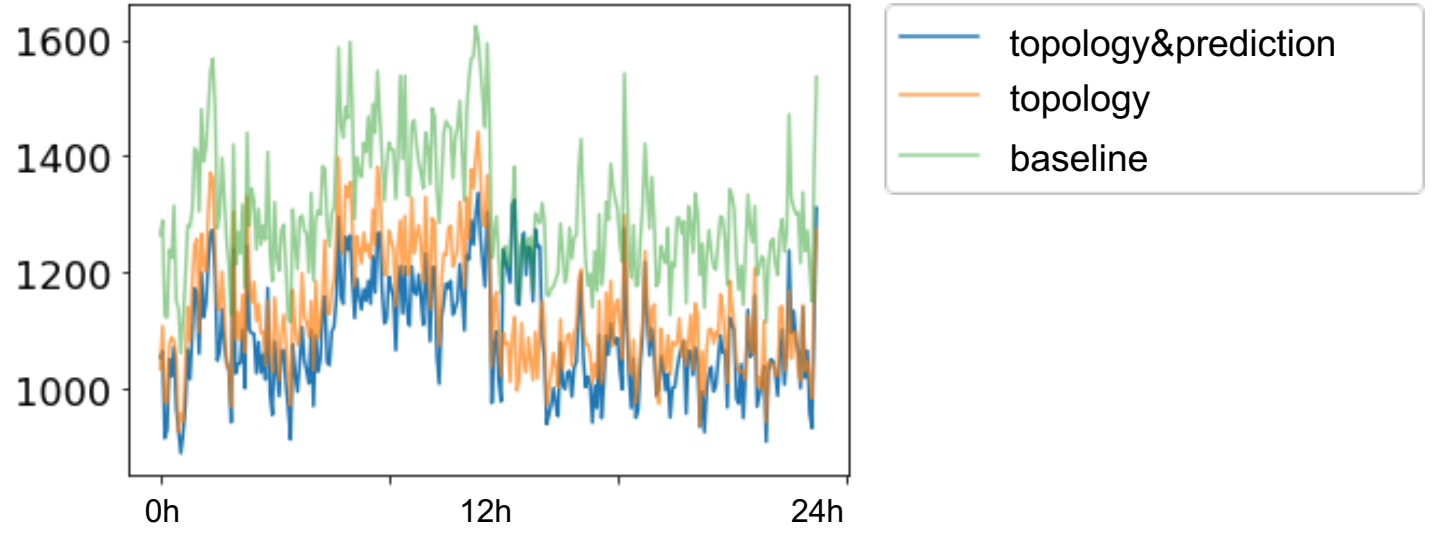
チェイン群2の経路候補

1. [EH3, 4, 2, 7, EH4]
2. [EH3, 4, 9, 7, EH4]
3. [EH3, 8, 9, 7, EH4]
4. [EH3, 8, 6, 7, EH4]
5. [EH3, 4, 5, 10, EH4]
6. [EH3, 4, 9, 10, EH4]
7. [EH3, 8, 9, 10, EH4]

チェイン群2の経路を選択するAIは、
混雑度の低いチェイン群3との重複が多い
(チェイン群1との重複が少ない)経路を優先

テストデータを用いた性能評価

- 学習に使ったトラヒックとは異なる日時のデータを投入し、性能を評価
  - 評価指標: ノード負荷(ノードを経由したトラヒック量)の最悪値
  - 下記3パターンを評価
    - トポロジー分析&機械学習による需要予測を併用 (topology&prediction)
    - トポロジー分析のみ利用 (topology)
    - どちらも使わずに均等の確率で経路を選択 (baseline)
- ほとんどの時間帯で提案手法が最もノード負荷を軽減
  - topology&predictionは、baselineに比べて平均20%負荷を軽減
  - topology&predictionは、topologyに比べて平均5%負荷を軽減



まとめと今後の課題

- エンコーダ・デコーダ(ED)RNNを用いたSFC移行スケジューリング
  - 時系列データをエンコーダ、デコーダで2度読み込み解析
  - 最適化問題の解を、教師データとして学習させる
  - リソースの不足を回避しつつ、VNFの移行回数も抑える

- 階層型AIアーキテクチャによるサービス機能チェイン経路決定手法
  - トポロジーの分析、機械学習によるトラヒック需要予測を組み合わせて経路の推奨度
  - 時刻毎の推奨度を再計算し、混雑しにくいと予測される経路を優先的に選択
  - 等確率に経路を選択(推奨度なし)、トポロジー分析(基準推奨度のみ)に比べて、ノードにかかる負荷(最悪値)をそれぞれ20%, 5%軽減
    - 推奨度なし→トポロジー分析：約15%軽減、トポロジー分析→機械学習との組み合わせ：更に約5%軽減

- 今後の課題
  - より大規模/複雑なトポロジーを対象にしたスケーラビリティ評価
  - 実機に実装し性能評価