

# コアペリフェリー構造を有する NFV ソフトウェアシステムの有効性評価

---

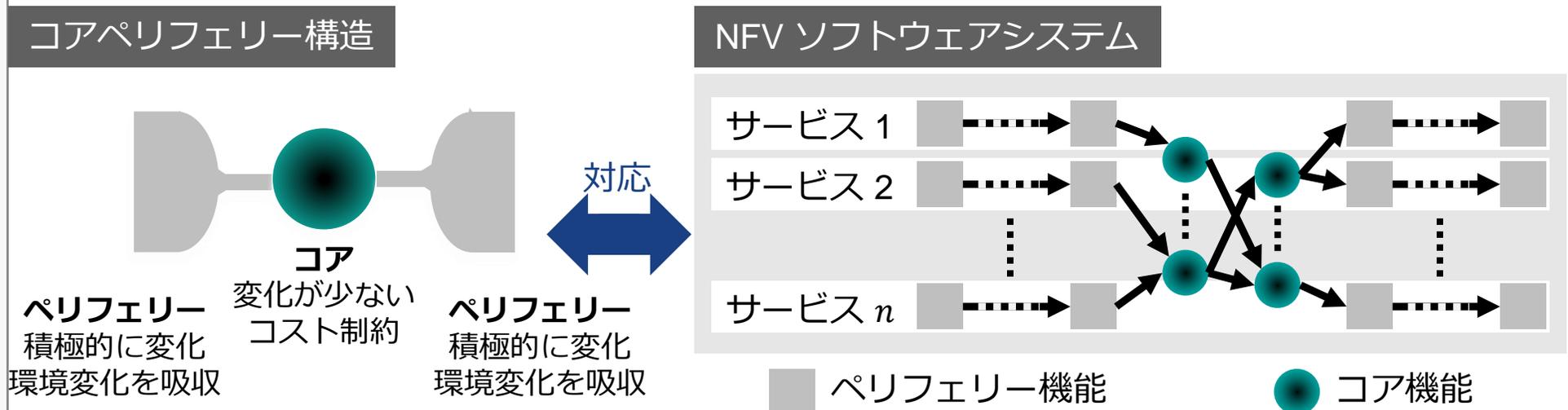
大阪大学 大学院情報科学研究科  
村田研究室 津久井佑樹

# 研究背景

- ・ **ネットワークへのユーザ要求の多様化<sup>[1]</sup>**
  - ・ スマートフォンなどのデバイスの世界的普及
  - ・ ビッグデータなどネットワークの新たな利用形態の登場  
→ ネットワークにはこれらへの対応が求められる
- ・ **NFV (Network Function Virtualization) への期待**
  - ・ ソフトウェア化されたサービスコンポーネントを仮想環境上で実行
    - ・ 複数のサービスコンポーネントが単一の汎用サーバ上で動作し、省電力・省スペース
  - ・ ユーザが要求するネットワーク化サービスをより柔軟に収容
    - ・ サービスコンポーネントをネットワークを介して連結することで収容可能
  - ・ 加えて、実装コスト抑制のため NFV ソフトウェアの適切な設計も重要
    - ・ 実装コストが少なければ、多様なユーザ要求への対応がより容易に

# コアペリフェリー構造による NFV ソフトウェアの解釈

- ・ **生物システムなどで確認されるコアペリフェリー構造<sup>[12]</sup>**
  - ・ システム要素を”コア”と”ペリフェリー”に分類
- ・ **NFV ソフトウェアはコアペリフェリー構造と解釈可能**
  - ・ サービスコンポーネントを”コア機能”と”ペリフェリー機能”に分類
    - ・ コア機能: 複数のネットワーク化サービス収容に用いられるサービスコンポーネント
    - ・ ペリフェリー機能: 1 種類のネットワーク化サービス収容にのみ用いられるサービスコンポーネント



# 研究目的

- ・ コア機能数の大小に応じて実装するペリフェリー機能数が変化

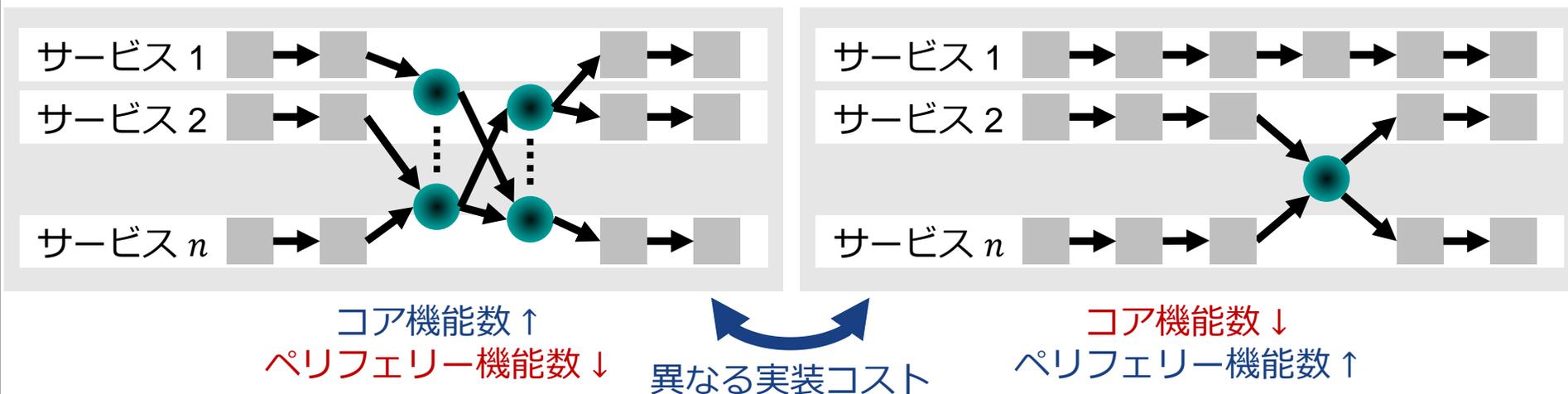
→ NFV ソフトウェアの実装コストを抑制するコア機能数の決定が必要

- ・ **研究目的**

全体の実装コストを抑制する NFV ソフトウェアの設計指針の獲得

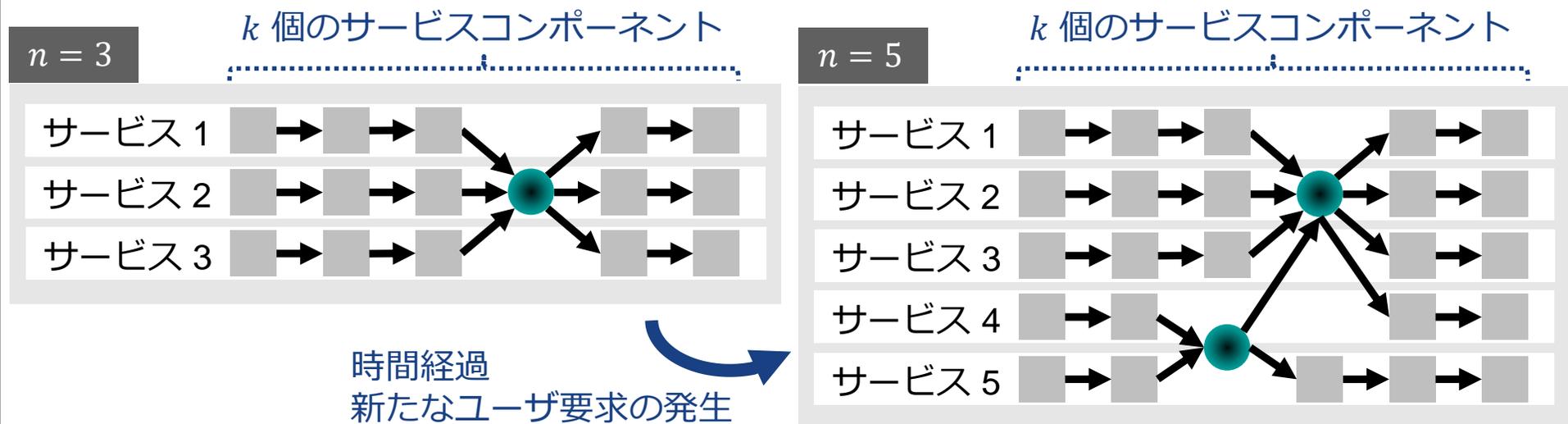
- ・ **アプローチ**

- ・ コアペリフェリー構造に基づいて実装コストをモデル化
- ・ 異なる数のコア機能を有した設計指針の実装コストを比較・議論



# ネットワーク化サービス収容のモデル化

- ・ 時間の経過に伴い新たなユーザ要求が生じ  
収容するネットワーク化サービスの種類数  $n$  が増加
  - ・  $n$  の増加に伴ってコア機能やペリフェリー機能を追加
  - ・ ネットワーク化サービスあたりの収容に用いる平均サービスコンポーネント数  $k$ 
    - ・  $n$  が増加しても  $k$  は変化しないと仮定



# NFV ソフトウェアの実装コストの定義式

## 全体の実装コスト

$$c_{all}(n) = \sum_{i=0}^{f_c(n)} c_c(i) + \sum_{j=0}^n k_p(j) c_p(j)$$

## $i$ 種類目のコア機能の実装コスト

$$c_c(i) = \alpha i$$

## $j$ 種類目のネットワーク化サービスの ペリフェリー機能あたりの実装コスト

$$c_p(j) = e^{-\beta f_c(j)}$$

## $j$ 種類目のネットワーク化サービス あたりのペリフェリー機能数

$$k_p(j) = k - k\gamma f_c(j)$$

## サービスコンポーネントの総数

$$f_{all}(n) = f_c(n) + f_p(n)$$

## ペリフェリー機能数

$$f_p(n) = \sum_{j=0}^n k_p(j)$$

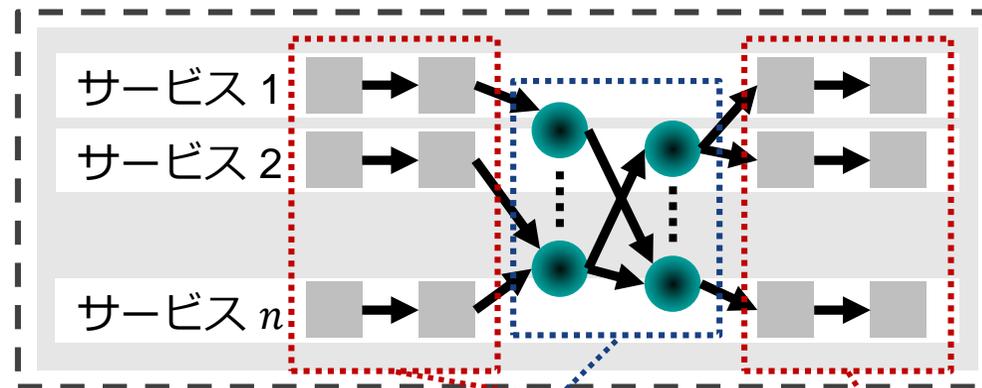
## 変数・パラメータ

$n$	ネットワーク化サービスの種類数
$k$	ネットワーク化サービスあたりの平均サービスコンポーネント数
$f_{all}(n)$	サービスコンポーネントの総数
$f_c(n)$	コア機能数
$f_p(n)$	ペリフェリー機能数
$k_p(j)$	$j$ 種類目のネットワーク化サービスあたりのペリフェリー機能数
$c_{all}(n)$	全体の実装コスト
$c_c(i)$	$i$ 種類目のコア機能の実装コスト
$c_p(j)$	$j$ 種類目のネットワーク化サービスのペリフェリー機能あたりの実装コスト
$\alpha$	実装済みのコア機能数に比例した $c_c(i)$ の増加度を決定するパラメータ
$\beta$	コア機能数の増加に伴った $c_p(j)$ の減少度を決定するパラメータ
$\gamma$	$f_c(n)$ 個のコア機能が $j$ 種類目のネットワーク化サービスの収容に用いられる頻度を表すパラメータ

# NFV ソフトウェア全体の実装コストのモデル化

## 各サービスコンポーネントの実装コストの和として定義

- $$c_{all}(n) = \sum_{i=0}^{f_c(n)} c_c(i) + \sum_{j=0}^n k_p(j) c_p(j)$$
  - $f_c(n)$  個のコア機能それぞれに実装コスト  $c_c(i)$  が必要
  - $n$  種類のネットワーク化サービスそれぞれの収容に  $k_p(j)$  個のペリフェリー機能を利用
  - それぞれのペリフェリー機能に実装コスト  $c_p(j)$  が必要



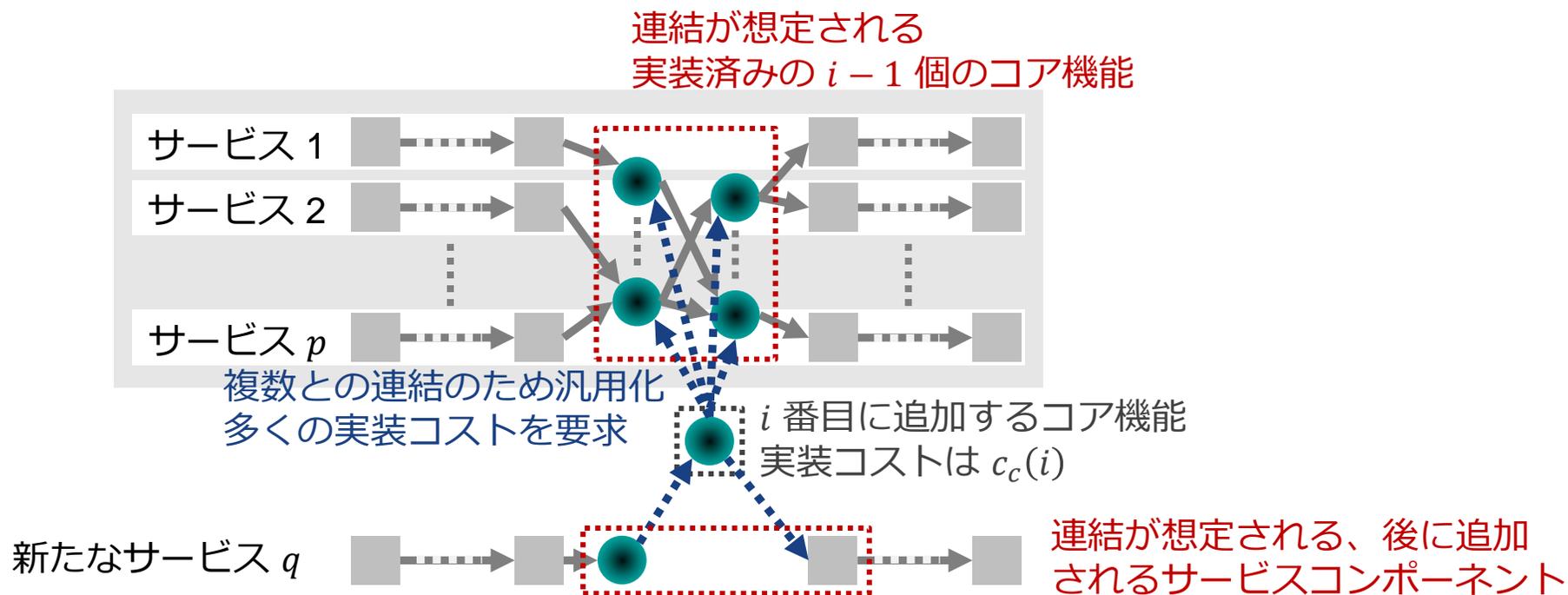
NFV ソフトウェア  
 全体の实装コスト  $c_{all}(n)$

$$= \text{コア機能の实装コストの和} + \text{ペリフェリー機能の实装コストの和}$$

$$= \sum_{i=0}^{f_c(n)} c_c(i) + \sum_{j=0}^n k_p(j) c_p(j)$$

# コア機能の実装コストのモデル化

- ・ 実装済みのコア機能数に比例して追加のコア機能の実装コストが増加
  - ・ コア機能は複数のサービスコンポーネントとの連結を可能にするために汎用化が必要
  - ・ より多くの連結を可能にするための汎用化に、より多くの実装コストを要求
  - ・  $c_c(i) = \alpha i$ 
    - ・  $\alpha$ : 実装済みのコア機能数に比例した  $c_c(i)$  の増加度を決定するパラメータ



# ペリフェリー機能の実装コストのモデル化

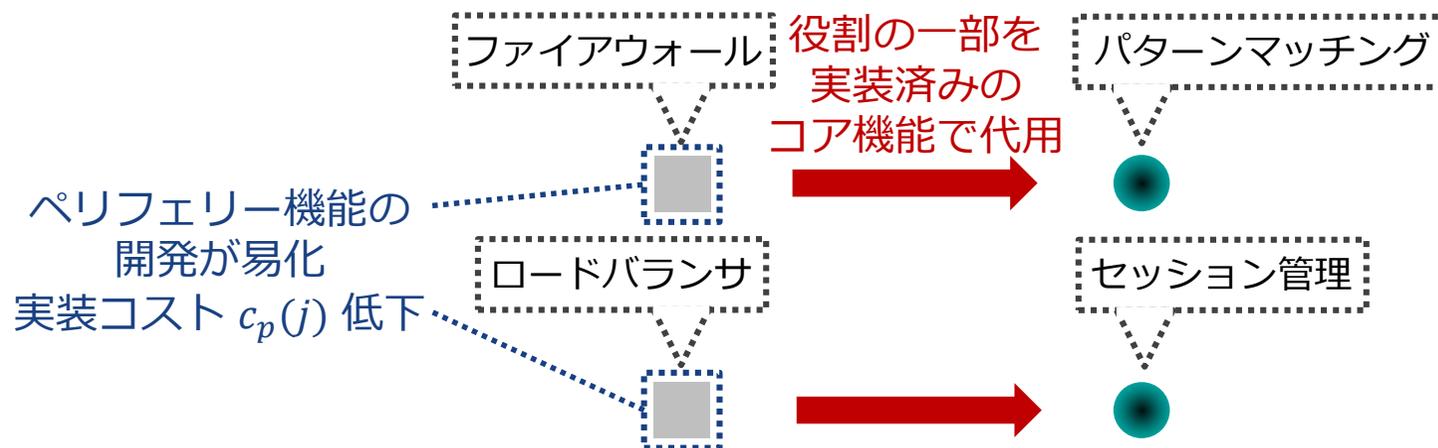
## コア機能数が増加するとペリフェリー機能の実装コストが減少

- ・ 役割の一部を実装済みのコア機能で代用することで開発が容易に
- ・ コア機能数が増加するほど代用できる機会が増加
- ・ ただしコア機能数増化に伴う  
ペリフェリー機能の実装コスト減少は一定の値に収束

- ・ 実際に代用される機能は一定のものに限定されるため

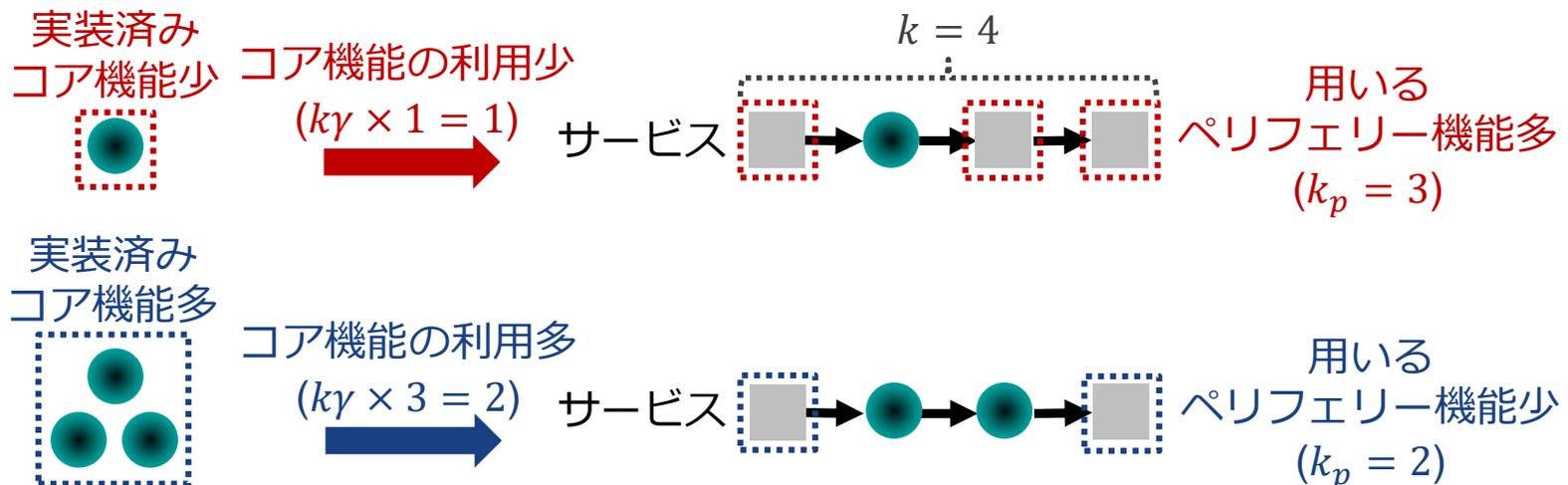
- ・  $c_p(j) = e^{-\beta f_c(j)}$

- ・  $\beta$ : コア機能数の増加に伴う  $c_p(j)$  の減少度を決定するパラメータ



# サービス収容に用いるペリフェリー機能数のモデル化

- コア機能数が増加するとネットワーク化サービス収容に用いるペリフェリー機能数が減少
  - コア機能数が増加するとネットワーク化サービスの収容にコア機能を利用する機会が増加
  - $k_p(j) = k - k\gamma f_c(j)$ 
    - ネットワーク化サービスあたりの収容に用いるコア機能を  $k\gamma f_c(j)$
    - $\gamma$ :  $f_c(n)$  個のコア機能が、あるネットワーク化サービスの収容に用いられる頻度を表すパラメータ



# 数値結果の導出シナリオ

- ・  $0 \leq n \leq 1000$  の範囲で  $n$  が増加をするシナリオを想定
- ・ コア機能数の異なる設計指針の実装コストを比較
  - ・ 各設計指針で収容するネットワーク化サービスの種類は同一と想定
    - ・ 変数およびパラメータも共通
  - ・ 時間経過に伴い、新たに多様なユーザ要求が生じても実装コストを抑えてネットワーク化サービスを収容可能かを重視
    - ・  $n = 1000$  時点の NFV ソフトウェア全体の実装コスト  $c_{all}(1000)$  を比較

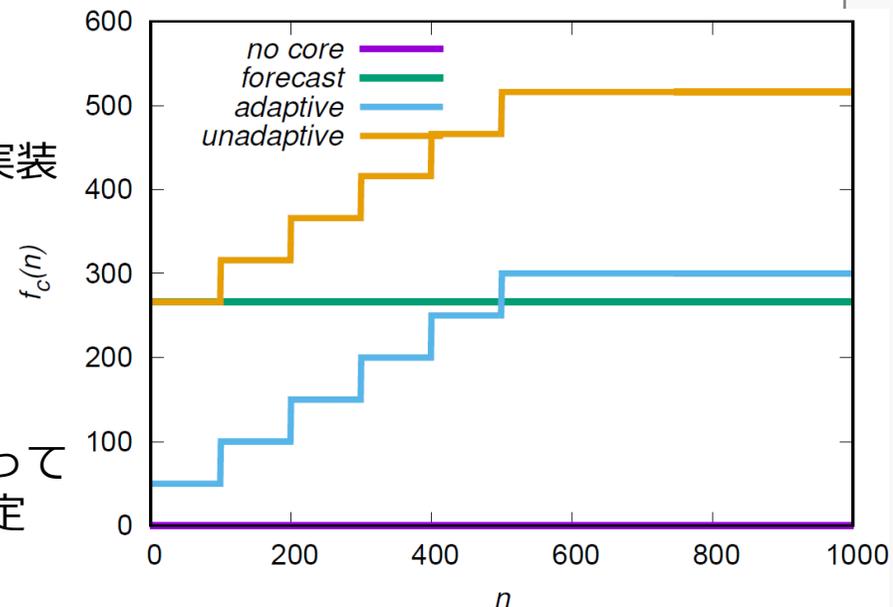
## 変数・パラメータ設定

パラメータ	値
$k$	10
$\alpha$	0.05
$\beta$	0.001
$\gamma$	0.001

# 比較に用いる設計指針

## 図のようにコア機能数を設定した四つの設計指針

- ・ *no core*
  - ・ コア機能を用いず  
ペリフェリー機能のみを用いてネットワーク化サービスを收容
- ・ *forecast*
  - ・ 将来のサービス收容を予測し、それに基づきコア機能を事前に実装
  - ・ 予測は正確で、事前に実装したコア機能が十分に再利用され  
実装コストを最小化可能であると仮定
- ・ *unadaptive*
  - ・ *forecast* と同様に予測に基づきコア機能を実装
  - ・ 予測は不正確と仮定し  
 $n$  の増加に伴い生じる想定外のサービスの收容のためコア機能を追加
- ・ *adaptive*
  - ・ 新たなネットワーク化サービスの收容に伴って  
コア機能の設計を見直し追加することを想定

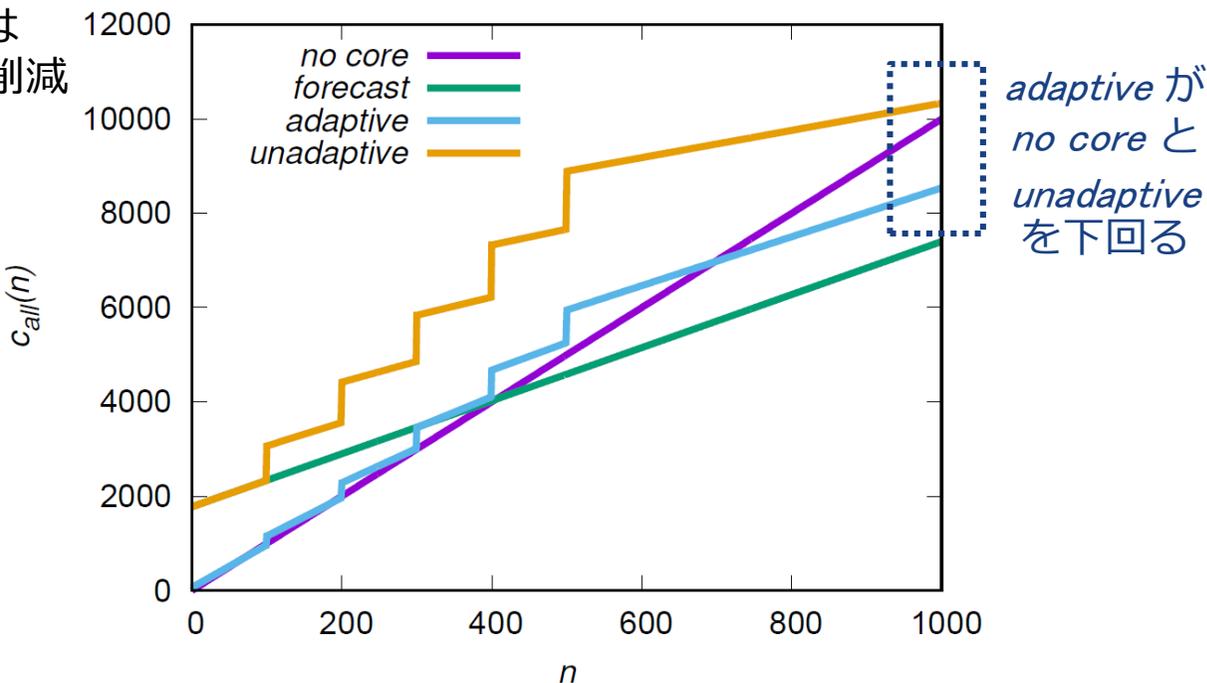


各設計指針のコア機能数の設定

## 数値結果: コア機能数の異なる設計指針の実装コストの比較

*adaptive* のようにコア機能を少量とし、新たなユーザ要求発生に伴い追加する設計指針がより現実的に全体の実装コストを抑制することを確認

- ・  $c_{all}(1000)$  は  $adaptive < no\ core < unadaptive$ 
  - ・ *forecast* は予測が正確であるという仮定があるため、他の設計指針がより現実的
- ・ コア機能とペリフェリー機能両方を有することで実装コストを抑制
  - ・ *adaptive* の  $c_{all}(1000)$  は *no core* と比べ約 15% 削減
- ・ コア機能数が多いと実装コストが増加



各設計指針の実装コスト

# まとめと今後の課題

## ・ まとめ

- ・ NFV ソフトウェアの実装コストに着目
- ・ コアペリフェリー構造による NFV ソフトウェアの解釈
  - ・ 複数のネットワーク化サービスの収容に用いられるコア機能
  - ・ 一種類のネットワーク化サービスの収容に用いられるペリフェリー機能
  - ・ コア機能数の大小によりペリフェリー機能数が変化
- ・ コア機能を少量とし、新たなユーザ要求発生に伴い追加する設計指針がより現実的に全体の実装コストを抑制することを確認

## ・ 今後の課題

- ・ コア機能およびペリフェリー機能の物理インフラへの配置方法の考察
  - ・ 物理インフラのリソース制約のため、サービスコンポーネント数を考慮
  - ・ 複数個所に複製したサービスコンポーネントを配置するシナリオが想定されるためコア機能およびペリフェリー機能の複製の容易さを考慮