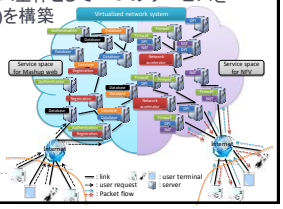
Construction method of service space in virtualized network system based on chemical-inspired tuple space model
(化学反応式を用いたタプル空間モデルに基づく
仮想化ネットワークシステムにおけるサービス空間構築手法)

松岡研究室
桜井 駿



## 研究の背景 (1/2)

- 仮想化ネットワークシステム
  - ネットワーク上において物理サーバが分散配置
  - 一台の物理サーバ内で複数の仮想サーバがネットワークを介してサービス機能を提供
  - サーバへのサービス機能の配置・再配置、及びサービス機能の分散実行によって、システム全体として一つのサービスを提供する空間(サービス空間)を構築






## 研究の背景 (2/2)

- サービス空間の効果的な構築に求められること
  - サービス機能のサーバへの配置場所の決定
    - 需要を考慮したサービス機能の配置
    - システム障害や環境変動に伴うサービス機能の再配置
  - 各サービス機能へのサーバ資源の割り当て
    - 需要に応じた割り当て
    - 複数サービス機能間の共有
  - サービス機能の分散実行
    - 複数サーバでの負荷分散

各サーバの自律分散的な動作で実現したい

システム障害、環境変動への迅速な対応
及びスケーラビリティの保持



## 研究の目的と手段

- 研究の目的

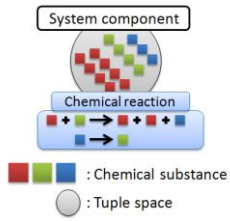
仮想化ネットワークシステムにおける
サービス空間構築手法を提案

  - 仮想化ネットワークシステムのサービス空間を各サーバの自律的な動作によって構築する
- 研究の手段
  - 化学反応式を用いたタプル空間モデルを応用



## 化学反応式を用いたタプル空間モデル(1/2)

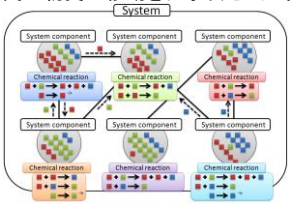
- システムの構成要素を一つのタプル空間としてモデル化
- 構成要素内の情報を化学物質として表現
  - 構成要素の状態や情報量をその濃度で表現
- 構成要素の動作を各タプル空間に設定する化学反応式で記述






## 化学反応式を用いたタプル空間モデル(2/2)

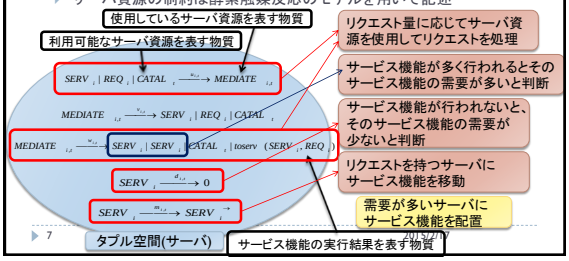
- タプル空間を接続し、ネットワークを構築
- 構成要素間の物質の移動を化学反応式で表現




システム全体の制御を各構成要素
の自律分散的な動作で実現できる

## 提案手法：サービス機能の実行及び配置

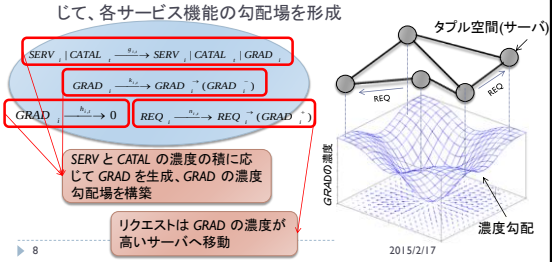
- サーバを一つのタプル空間としてモデル化
- サーバでのサービス機能の実行を化学反応式で記述
  - 反応の速度は反応物の濃度に応じて決定
  - サーバ資源の制約は酵素触媒反応のモデルを用いて記述






## 提案手法：勾配場によるリクエストの移動

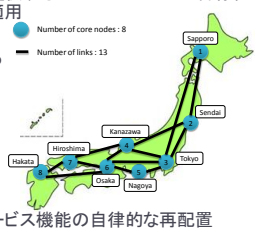
- リクエストが、より適切なサーバへ移動する挙動を化学反応式として記述
  - サーバの利用可能な資源とサービス機能に対する需要に応じて、各サービス機能の勾配場を形成






## 性能評価：評価環境

- SINETのバックボーンネットワークトポロジーを用いた評価
  - 各ノードにサービス機能を提供するためのサーバが一台存在すると仮定し、提案手法を適用
  - 各サーバは単位時間あたり 50 のリクエストを処理できる資源を持つと仮定
    - *CATAL* (触媒)の初期濃度 1,000に相当
- 確認する動作
  - サービス機能間の自動的なサーバ資源割り当て
  - 需要が多いサーバへのサービス機能の自律的な再配置

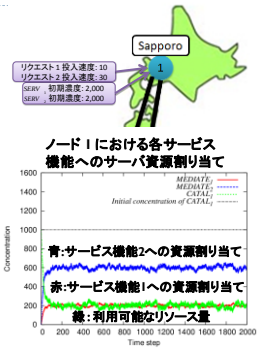





## シナリオ1：サーバ資源の割り当て

- 提供するサービス機能: 2 種類
- ノード 1 の 各 *SERV* の初期濃度: 2,000
- リクエスト投入速度
  - リクエスト 1: 単位時間当たり 10
  - リクエスト 2: 単位時間当たり 30
- シミュレーション結果
  - リクエスト投入速度に応じて各サービス機能の *MEDIATE* の濃度が一定値に収束
  - *CATAL* の濃度が *MEDIATE* の濃度の増加に応じて減少

サーバ資源をリクエスト量に応じて各サービス機能に割り当て、サービス機能を実行している

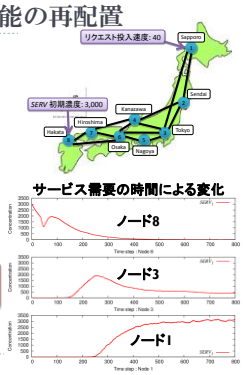





## シナリオ2：サービス機能の再配置

- ノード 8 の *SERV* の初期濃度: 3,000
- ノード 1 へのリクエスト投入速度
  - リクエスト: 単位時間当たり 40
- シミュレーション結果
  - *SERV* がノード8からリクエストが投入されているノード1に移動している

サービス機能を実行するに適したノードにそのサービス機能が移動






## まとめと今後の課題

- 仮想化ネットワークシステムにおいて、各サーバの自律分散的な動作によってサービス空間を構築する手法を提案
- シミュレーション評価により、リクエスト量に応じたサーバ資源割り当て、サービス機能の分散実行、適切なサーバへのサービス機能の再配置が実現できることを示した
- 提案手法のNFVに適用する拡張モデルを説明し、提案手法が現実的なアプリケーションに対応できることを示した
- 今後の課題
  - NFVに適用した拡張方式の性能評価
  - システム障害や環境変動に対する性能評価

## 付録



## 提案手法の拡張：NFVに対する適用

- Network Function Virtualization (NFV)
  - ネットワーク機能が仮想化され、サーバに配置
  - フローは決められた順序で各機能を適用
- ネットワーク機能の配置場所やフローがどのサーバで機能を適用するかを決定するために提案手法を拡張

フローを適用する機能の勾配場に応じて移動

機能を適用後、フローは次に適用する機能のリクエストを持つフローに変化

自律的なサービス空間の構築を実現

$$SERV_f \mid FLOW_c \xrightarrow{\tau_{c,f}} SERV_f \mid SERV_f \mid FLOW_{c,f,f'(c,f)} \mid toserv\ (SERV_f, FLOW_c)$$

$$SERV_f \xrightarrow{\tau_{c,f}} 0 \quad SERV_f \xrightarrow{\tau_{c,f}} SERV_f \quad FLOW_c \xrightarrow{\tau_{c,f}} FLOW_c\ (GRAD_{f,f'(c,f)})$$

$$SERV_f \xrightarrow{\tau_{c,f}} SERV_f \mid GRAD_f \quad GRAD_f \xrightarrow{\tau_{c,f}} 0 \quad GRAD_f \xrightarrow{\tau_{c,f}} GRAD_f\ (GRAD_f)$$



## シナリオ3：負荷分散

- ノード 6 の *SERV* の初期濃度: 1,000
- ノード 3 へのリクエスト投入速度
  - リクエスト: 単位時間当たり 80
- *SERV* の拡散範囲をノード4、5、6に制限
  - ノード 3 ではサービスを実行できない
- シミュレーション結果
  - サービス機能の実行が三つのノードで分散的に行われている

負荷分散の実現


十分に時間が経過した際のサービス機能実行結果の生成速度