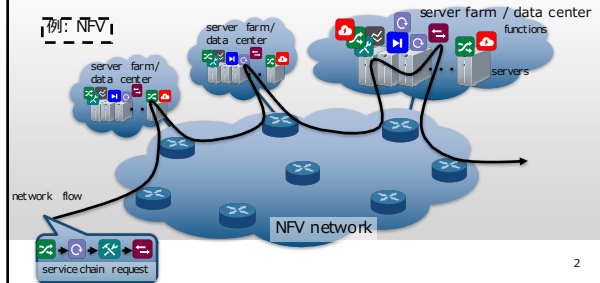


化学反応式モデルに基づく仮想サービス配置手法の安定性評価

長谷川 剛 (ごう)、 村田 正幸
 1大阪大学サイバーメディアセンター
 2大阪大学大学院情報科学研究科

仮想サービス配置問題

- 対象: クラウド環境を利用したネットワークサービス
 - 例: NFV、マッシュアップWebサービス、...
- サーバにサービス (機能、コンポーネント) を配置して構築される



本研究の目的

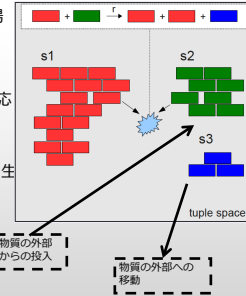
- 提供するサービスの負荷、ユーザからの需要などに応じて適切に配置する必要がある
 - サービスのサーバへの配置、サーバ・ネットワーク資源の割当
 - システム障害や需要変動に応じた動的な配置及び資源割当の変更
- 自律的な制御が求められる



- 化学反応式を用いた空間協調モデルに基づくネットワークサービス空間構築手法を提案
- パラメータチューニングに対する安定性を評価

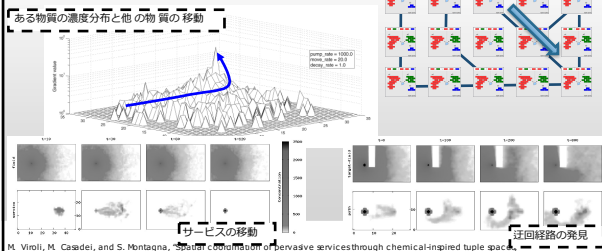
化学反応式を用いた空間的協調手法 (1)

- Tuple space (ノード): 化学反応が起こる場
- 複数の化学反応式が定義される
 - $s1 + s2 \xrightarrow{r} s1 + s1 + s3$
 - 反応速度は、反応物質 (s1, s2) の濃度と反応速度係数 (r) の積に比例して決定される
- 化学反応式の定義で様々な挙動を実現
 - サービスやリクエスト等を表す物質の提供、生成、死滅など



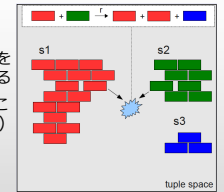
化学反応式を用いた空間的協調手法 (2)

- 複数ノードがネットワークを形成
- 反応の結果生成された物質がネットワークを介して拡散する
- 拡散された物質の濃度分布に応じて他の物質が移動する



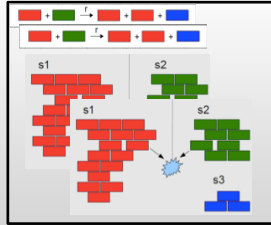
ノードにおけるサービスの実行、成長、衰退

- (USE) $SERV + REQ \xrightarrow{u} SERV + SERV + toserve(SERV, REQ)$
 - SERV: サービス (コンポーネントや機能) を表す
 - 濃度はそのサービスの需要の大きさを表す
 - REQ: リクエストを表し、外部から投入される
 - サービスがリクエストと適合すると、サービスを実行 (toserveを生成) し、リクエストを削除する
 - サービスが実行された際に、SERVを増加することでサービスを成長させる (positive feedback)
- (DECAY) $SERV \xrightarrow{d} 0$
 - 実行されないサービスは徐々に衰退する (negative feedback)



複数サービスの共存

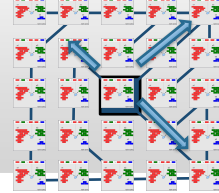
- (USE1) $SERV1 + REQ1 \xrightarrow{r_1} SERV1 + SERV1 + toserve(SERV1, REQ1)$
- (USE2) $SERV2 + REQ2 \xrightarrow{r_2} SERV2 + SERV2 + toserve(SERV2, REQ2)$
- (DECAY1) $SERV1 \xrightarrow{d_1} 0$
- (DECAY2) $SERV2 \xrightarrow{d_2} 0$
 - 2つのサービスが1つのノードで行われる
 - SERVの濃度に応じてサーバ資源を割り当てることで、需要に応じた資源共有を行う



7

サービスの拡散

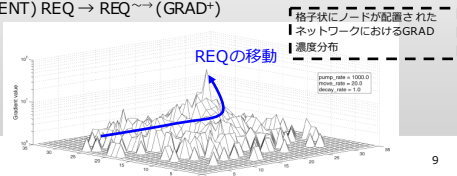
- (DIFFUSE) $SERV \xrightarrow{m} SERV \rightsquigarrow$
 - サービスは繋がっている周囲のノードへ(均等に)拡散する
 - 拡散先でREQと反応すれば、そこでのサービスが成長する
 - 拡散先にREQが無ければ、そこでのサービスは衰退する
- サービスの需要が大きいと、濃度が高くなり、周囲への拡散速度が大きくなる
 - 需要が高いサービスの拡大



8

リクエストの移動

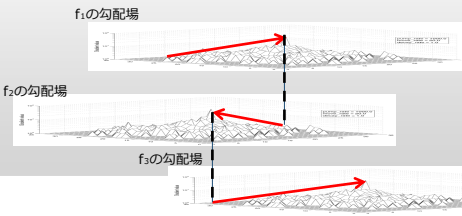
- あるノードのREQがSERVと反応しない場合、サービスが行われている (= SERV濃度の高い) ノードへ移動させる
- SERV濃度に応じた勾配場の形成
 - (GRAD_PUMP) $SERV \xrightarrow{p} SERV + GRAD$
 - (GRAD_DECAY) $GRAD \xrightarrow{d_g} 0$
 - (GRAD_MOVE) $GRAD \xrightarrow{m_g} GRAD \rightsquigarrow$ (GRAD)
- 勾配に基づくREQの移動
 - (REQ_ASCENT) $REQ \xrightarrow{m_r} REQ \rightsquigarrow$ (GRAD)



9

NFVのサービスチェイニングの実現

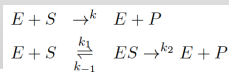
- (USE) $SERV + REQ \xrightarrow{r} SERV + SERV + REQ + toserve(SERV, REQ)$
 - 物質REQを持つサービスチェーン要求: $\{f_1, f_2, f_3\}$
 - フローがサービスを受けると、受けたサービスをChainから削除し、REQに変化する - $\{f_1, f_2, f_3, \dots\} \rightarrow \{f_2, f_3\}$
- 勾配場はサービス毎に形成
- フローは次に受けるサービスの勾配場に応じて移動し、サービスを受ける



10

処理能力の上限の導入 (1)

- (USE) $SERV + REQ \xrightarrow{r} SERV + SERV + toserve(SERV, REQ)$
 - 反応式左辺の物質濃度の積だけで反応速度を決めるモデルでは、SERVとREQが増えるとそれに従って反応速度が増加する
 - 反応速度に上限を与えることができない
- ミカエリス・メンテン式
 - 酵素触媒反応の反応速度を説明するモデル
 - 基質S、酵素E、生成物P
 - 酵素が基質と結合して生成物を生成する
 - 中間体ES
 - 触媒反応を説明するために導入した(仮想的な)物質



- 触媒 (E) の濃度によって反応速度の上限が決まる

11

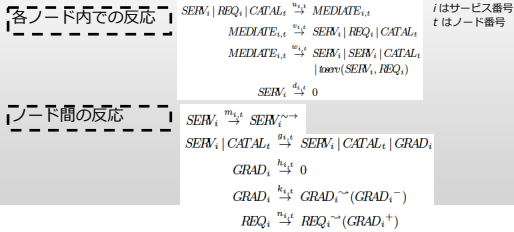
処理能力の上限の導入 (2)

- (USE) 反応式を下記のように拡張
 - $SERV + REQ + CATAL \xrightarrow{r} MEDIATE$
 - $MEDIATE \xrightarrow{v} SERV + REQ + CATAL$
 - $MEDIATE \xrightarrow{m} SERV + SERV + toserve(SERV, REQ)$
- CATAL: 触媒を表す物質 (前頁 E に相当)
 - CATAL濃度は、余剰のサーバ資源を表す
- MEDIATE: 触媒反応を説明するために導入した中間物質 (前頁 ES に相当)
 - MEDIATE濃度は、各サービスに割り当てられたサーバ資源を表す

12

パラメータチューニングとシステム安定性

- 提案システムは多くのパラメータを持つ
 - 各化学反応式の反応速度係数 $u, v, w, d, m, p, d_g, m_g, m_r$
 - 反応速度は反応物質の濃度と反応速度係数の積で決定される
- パラメータの設定方法と、パラメータ設定がシステム安定性に与える影響の評価が必要



パラメータ設定方法の例

- 例: サーバがWebサービスを行う場合
 - $SERV_i | REQ_i | CATAL_t \xrightarrow{u_i,t} MEDIANE_{i,t}$ (8)
 - $MEDIANE_{i,t} \xrightarrow{v_i,t} SERV_i | REQ_i | CATAL_t$ (9)
 - $MEDIANE_{i,t} \xrightarrow{w_i,t} SERV_i | SERV_i | CATAL_t$ (10)
 - $| toev(SERV_i, REQ_i)$ (10)
 - $SERV_i \xrightarrow{d_i,t} 0$ (11)
- (8)(9)式: サービスへのCPU等のリソース割り当てに相当
- (10)(11)式: サービスのCPU等より十分速く反応するように設定する
- (10)式: サービスの実行に相当
 - サーバが1つのリクエストを処理するのにかかる時間に応じて決定
 - CPU能力、I/O速度、リクエストのサイズ(ドキュメントサイズ等)等に応じて決定
- 各サーバの資源量、ネットワーク速度などに応じて各ノードで独立に決定する

パラメータ設定がシステム安定性に与える影響 (1)

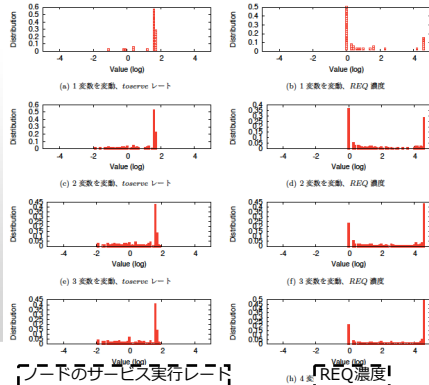
- 各ノードが独立にパラメータ設定を行うと、システム全体として安定して動作することが担保できない
 - システム全体ではリクエストを処理できるだけの資源が存在するにもかかわらず、パラメータ設定が悪いためにリクエストを処理し切れない、などが発生することがある
- 文献[17]: 遺伝子ネットワークやシグナル伝達ネットワークなどの、化学反応式のネットワークによって表現されるシステムの安定性を議論
 - Distributed robustness: システムが持つ全てのパラメータのあらゆる変動に対する、システム出力の変動の大きさ
 - r-robustness: システムが持つパラメータのうち r 個のあらゆる変動に対する、システム出力の変動の大きさ
 - Weak r robustness: システムが持つパラメータのうち r 個のランダムな変動に対する、システム出力の変動の大きさ

[17] A. N. Gorban and O. Radulescu, "Dynamical robustness of biological networks with hierarchical distribution of time scales," *IET Systems Biology*, vol. 1, pp. 238–246, July 2007.

パラメータ設定がシステム安定性に与える影響 (2)

- 非線形システムにおけるロバスト性の評価
 - システムが持つパラメータのうちいくつかをランダムに変動させた時の、システム出力の変化を観察し、それを基にシステムが(weak) r-robustness であるか否かを評価する
- 化学反応式の反応速度係数 (1ノードあたり9つ×ノード数だけ存在) をランダムに設定し、システム出力を観察することで、システム安定性を評価する
- システムの出力
 - 各ノードにおけるサービス実行レート (toserveの生成レート)
 - システム全体のサービス実行レートの和
 - 各ノードにおける最終的なREQ濃度 (サービスを受けるまでの遅延時間に相当)
- 評価方法
 - ランダムに設定するパラメータの数がシステム出力の変動の大きさに与える影響を評価
 - 複数ノードシステムのトポロジはフルメッシュ

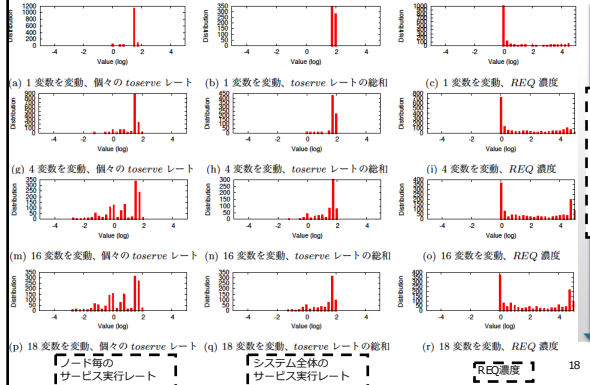
評価結果: 1ノードシステムの場合



変動パラメータ数

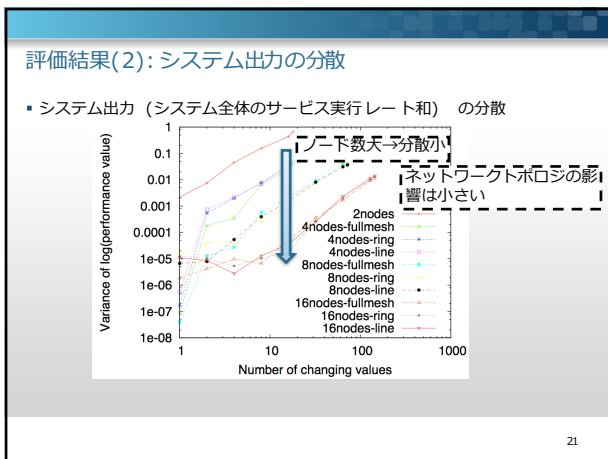
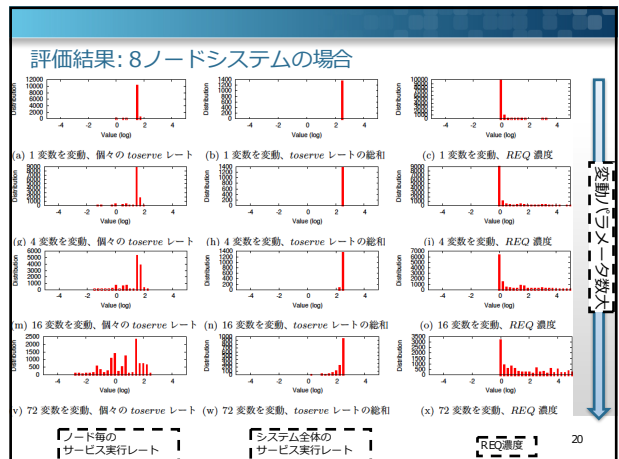
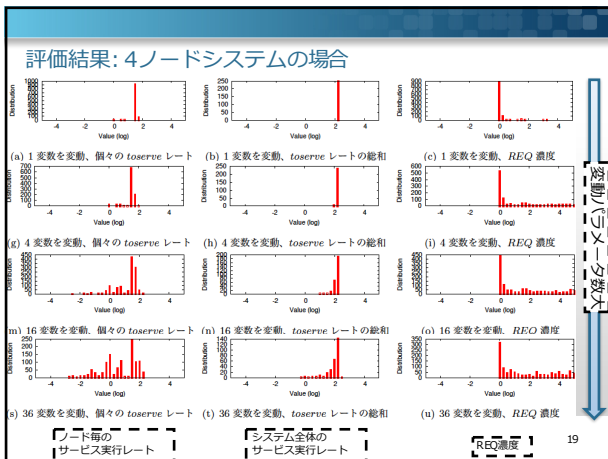
ノードのサービス実行レート (a) (b) (c) (d) (e) (f) REQ濃度

評価結果: 2ノードシステムの場合



変動パラメータ数

ノード毎のサービス実行レート (a) (b) (c) (d) (e) (f) (g) (h) (i) 18変数変動, toserveレートの総和 (j) 18変数変動, toserveレートの総和 (k) 18変数変動, REQ濃度



考察

- 個々のノードのサービス実行レート
 - 変動するパラメータ数やノード数が大きくなると出力のばらつきが大きくなる
- システム全体のサービス実行レート、REQ 濃度
 - 変動するパラメータ数の影響を受けにくい
 - ノード数が多くなると、逆に小さくなる
 - 個々のノードの出力のばらつきを、他のノードが吸収する
- ネットワークポロジの影響は小さい

↓

- システムを構成するノード数が多くなることによって、パラメータ設定に対するシステム全体のロバスト性は向上する

22

まとめと今後の課題

- 化学反応式を用いた空間協調モデルに基づくネットワークサービス空間構築手法
 - 各ノードの挙動を化学反応式として定義して自律分散実行させることで、ネットワークサービス空間の適応的な構築を行う
 - パラメータ設定に対するシステムのロバスト性を評価し、ネットワーク規模が大きくなるとロバスト性が高まることを確認した
- 今後の課題
 - 冗長系の配置手法
 - 具体的なサービスへの適用検討
 - NFV、モバイルコアネットワーク、...

23

24

