大規模ネットワークの大域的秩序形成のための自律分散制御および仮想計算機配置問題への応用

○ 作元 雄輔(首都大東京)
会田 雅樹(首都大東京)

研究の背景と目的

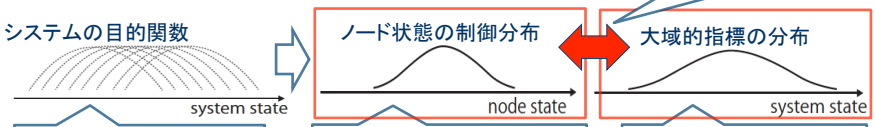
- ネットワークは**大規模・複雑なシステム**の代表例
  - 膨大な数のノードが接続されている
  - システムを取り巻く環境は動的に変化し，複雑な挙動を示す
- 大規模・複雑なシステムに対応可能な制御技術が求められる
  - 個々のノードは自律分散的に動作
  - システム全体を望みの方向に制御できる (システム**大域的秩序**を形成できる)

研究の目的: **システム大域的秩序の形成のための自律分散制御技術**の構築

自律分散制御によって大域的秩序を形成するための検討

- 考慮すべきこと
  - ノードはシステムの部分的な情報(局所情報)しか取得できない
  - システムを取り巻く環境は時間とともに変化
- 求められること
  - ノードは**確率的要素**を取り入れた**幅のある制御**を行う必要がある
  - システム全体を望みの方向に制御できる
    - 大域的指標の分布を制御できるように

どのようにしたら
結びつけたら良いか??



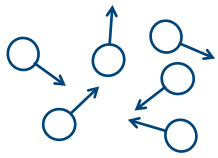
関数が定まらないため
確定的に制御できない

確率的要素を取り入れた
幅のある制御

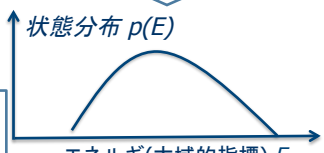
望み通りに制御できる

統計力学の知見を用いた自律分散的秩序形成の可能性

- **統計力学**: システムの微視的なダイナミクスと大域的な性質を結びつける力学
  - システムの微視的なダイナミクス
    - 例: 分子の運動
    - 全分子の状態を知ることはできないため確率的な記述になる
  - システムの大域的性質
    - 例: システムのエネルギがボルツマン分布に従う



微視的なダイナミクス
(個々の分子の確率的振舞い)

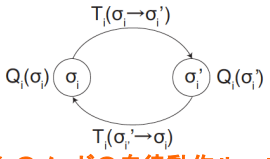


システムの大域的性質

個々のノードの**確率的な動作**と
通信システムの大域的性質を結びつけるために
統計力学の知見が利用できるのでは?

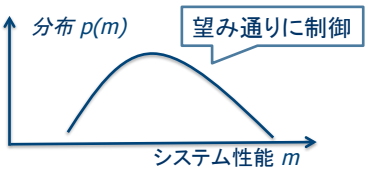
自律分散的秩序形成技術の概要

- 統計力学の知見に学び，個々のノードが自律分散的に振る舞うにもかかわらず，システム内に**大域的秩序を形成できる制御技術**を構築
  - 大域的秩序：システム全体を望みの方向に制御できること
  - 統計力学の知見である**メトロポリス法**をノードの自律動作設計に応用
    - メトロポリス法：システムの大域的物理量が従う分布を制御できる



個々のノードの自律動作ルール
(ノード状態の確率的遷移図)

メトロポリス法によって設計



システム性能が従う分布

システム性能が従う分布の制御によってシステム全体を制御する!

自律分散的秩序形成技術の設計：システムモデルとノードの基本動作

- システムモデル
  - システムの状態 σ は各ノードの状態 σ_i の組で表される

$$\sigma = (\sigma_1, \sigma_2, \ldots, \sigma_{|V|})$$

  - システムの性能 $m(\sigma)$ は各ノードの性能 $m_i(\sigma)$ の**線形和**で表される

$$m(\sigma) = \sum_{i \in V} m_i(\sigma)$$

    - ノード性能 $m_i(\sigma)$ は小さければ小さいほど良いとする
- ノードの基本動作
  - 各ノードは自律動作ルールに基づいて自身の状態を変化させる
    - 状態遷移を行った場合の性能差(以下)が得られると仮定

$$\Delta m_i(\sigma \to \tilde{\sigma}_{\sigma_i \to \sigma_i'}) = \underbrace{m_i(\tilde{\sigma}_{\sigma_i \to \sigma_i'})}_{\text{状態遷移後の性能}} - \underbrace{m_i(\sigma)}_{\text{現状態の性能}}$$

自律分散的秩序形成技術の設計：大域的秩序形成のための状態遷移確率の設計

- メトロポリス法を状態遷移確率 $T_i(\sigma_i \to \sigma_i')$ の設計に応用
  - メトロポリス法：システムの大域的物理量が従う分布を制御する方法
  - 具体的には，以下の詳細釣り合いの条件から $T_i(\sigma_i \to \sigma_i')$ を導出
    ＝ $Q_i(\sigma_i)$ が定常分布となるように $T_i(\sigma_i \to \sigma_i')$ を逆算

詳細釣り合いの条件

$$Q_i(\sigma_i)T_i(\sigma_i \to \sigma_i') = Q_i(\sigma_i')T_i(\sigma_i' \to \sigma_i)$$

$Q_i(\sigma_i)$：ノードの状態の分布(入力)

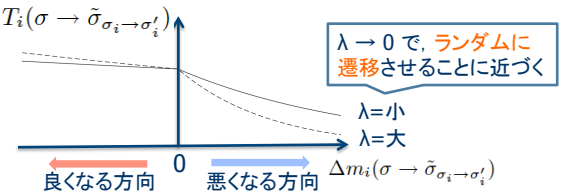
システム全体が従う分布から決定

自律分散的秩序形成技術の設計：メトロポリス法に基づく自律動作ルール

- ノードは状態遷移確率 T_i に従って自身の状態を確率的に遷移させる

$$T_i(\sigma \to \tilde{\sigma}_{\sigma_i \to \sigma_i'}) = \begin{cases} \beta \exp\left\{-\alpha\lambda\Delta m_i(\sigma \to \tilde{\sigma}_{\sigma_i \to \sigma_i'})\right\} & if\ \ \Delta m_i(\sigma \to \tilde{\sigma}_{\sigma_i \to \sigma_i'}) \leq 0 \quad \text{(性能が良くなる場合)} \\ \beta \exp\left\{-(1-\alpha)\lambda\Delta m_i(\sigma \to \tilde{\sigma}_{\sigma_i \to \sigma_i'})\right\} & otherwise \quad \text{(性能が悪くなる場合)} \end{cases}$$

ただし，$\beta = 1/|V|$　$0 < \alpha < 0.5$



λ → 0 で，ランダムに遷移させることに近づく

※ $Q_i(\sigma_i) = \exp\{-\lambda m_i(\sigma)\}$ を使用

自律分散的秩序形成技術：状態遷移確率に対応した大域的性質

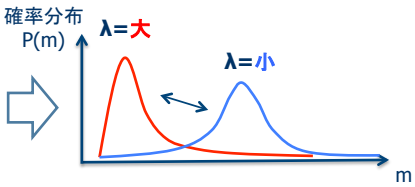
- システムの性能 m は以下の分布 $P(m)$ (ボルツマン分布)に従う
  - ボルツマン分布：統計力学でよく性質が理解されている分布
  - ボルツマン分布であればシステム全体を望みの方向に制御することが可能となる

ボルツマン分布

$$P(m) = \frac{1}{Z} g(m) \boxed{\exp\{-\lambda m\}}$$

$$Z = \sum_{x \in \Omega} g(x) \exp\{-\lambda x\}$$

※ g(m)：ランダムにシステム状態を決めた時のシステムの性能分布



λが大きくなるとランダム性が弱くなり，良い m が選択されやすくなる

DCN(Data Center Network)における仮想計算機配置問題への応用

- VM (Virtual Machine) 配置問題
  - VM を適切な PM (Physical Machine) に配置する問題
    - DCN のリソースを有効活用することが目的
- システムの均一化と局所化を考慮した自律分散制御を設計
  - 「システム均一化」が行なわれるように配置
    - 物理計算機の負荷が均一になるように物理計算機を配置
  - 「システム局所化」が行なわれるように配置
    - 通信頻度の高い仮想計算機同士を近くに配置
  - 均一化と局所化を適切なバランスの下で制御を実行
    - 均一化と局所化はトレードオフの関係にあるため

DCN における
仮想計算機配置問題の定式化

- システムのノード ⇔ PM
- ノード状態 ⇔ PM が保持する VM 集合
- システム性能 ⇔

ランダム性が高くなると良くなる指標である

全PM間の総トラヒック遅延積
(for **システム局所化**)

$$s(\sigma) = \sum_{i=1}^{N_{PM}} s_i(\sigma)$$

ノードごとの総トラヒック遅延積

$$s_i(\sigma) = \sum_{j=1}^{N_{PM}} C_{ij} D_{ij}$$

他の PM とのトラヒック遅延積の和

全 PM 負荷に対する2次モーメント
(for **システム均一化**)

$$v(\sigma) = \frac{1}{N_{PM}} \sum_{i=1}^{N_{PM}} v_i(\sigma)$$

ノードごとの2次モーメント

$$v_i(\sigma) = \left\{ \sum_{k \in \sigma_i} (\rho_k^{(VM)}) - e(\sigma) \right\}^2$$

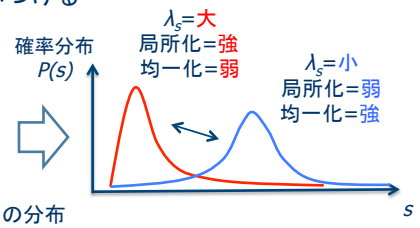
負荷の平均値 $e(\sigma)$ からのばらつき

システム局所化と均一化を考慮した
仮想計算機配置制御

- システム局所化と均一化を自律分散的秩序形成技術によって制御
  - 局所化制御：総トラヒック遅延積 *s* をボルツマン分布 *P(s)* に従わせる
    - 各ノードは *s* の変化を見て状態遷移を行う
  - 均一化制御：ランダム性と対応づける

s に対するボルツマン分布

$$P(s) = \frac{g(s)\, e^{-\lambda_s s}}{\sum_{s' \in \Omega} g(s')\, e^{-\lambda_s s'}}$$

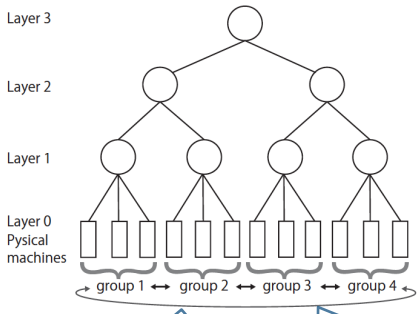


g(s) : ランダムにシステム状態を決めた時の s の分布
λ_s : 局所化の強さを表すパラメータ
(ランダム性の弱さを表すため均一化の弱さとも言える)

λ_s の調整によって均一化と局所化のバランスを制御

DCN モデルによる実験の概要：ネットワークトポロジとパラメータ

ネットワークトポロジ



パラメータ

ノード数 (PM 数)	12
VM 数	120
α	0.1
リンクの通信遅延	0.05
PM 内の通信遅延	0.0001
高通信頻度の VM 数	9
低通信頻度の VM 数	110
高通信頻度トラヒックレート	1.0
低通信頻度トラヒックレート	0.1
VM の負荷	1
シミュレーション時間	1,000,000

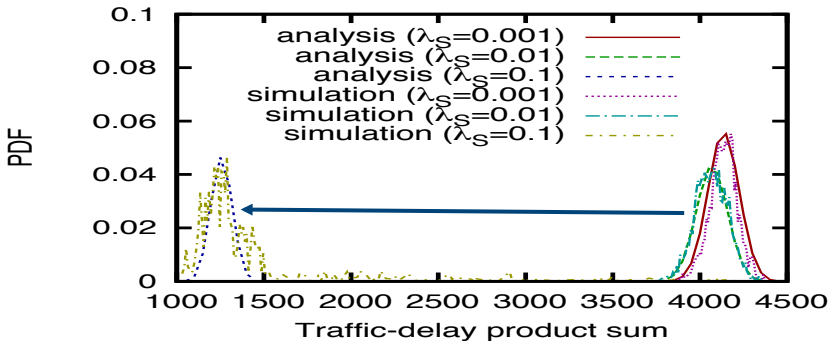
初期時間において
ランダムに VM を配置

PM を 4つのグループに分け，
隣接のグループ間でのみ
VM を移動できるとする

simulation: シミュレーションの時系列データから求めた分布
analysis: ボルツマン分布

実験結果：システム局所化に対するシステム大域的秩序の形成の確認

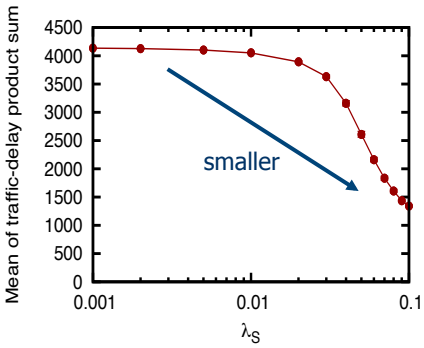
局所化の強さ λ_S を変化させた時の総トラヒック遅延積 s の分布



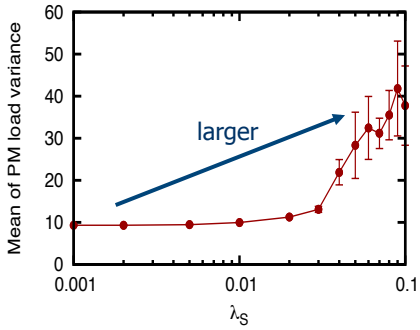
λ_Sを変化させることにより，分布全体を望み通りに制御できている

実験結果：局所化と均一化のバランスの確認

λs vs 総トラヒック遅延積 s の平均



λs vs 負荷の2次モーメント v の平均



λ_S を変化させることにより，均一化と局所化のバランスを調整できている

まとめ

- 大規模ネットワークの大域的秩序形成のための自律分散制御
  - 各ノードの自律的な動作によってシステム全体を望みの方向に制御
  - メトロポリス法を用いてノードの自律動作ルールを設計
- DCN の仮想計算機配置問題への応用
  - システム局所化と均一化を考慮した自律分散制御を設計
  - 実験を通して大域的秩序の形成を確認すると共に，
    局所化と均一化のバランスを調整できることが確認できた

今後の課題

- 様々な状況における有効性の評価
  - 異なるネットワークトポロジを用いた場合の評価
  - トラヒック変動や負荷の変動を考慮した状況下での評価
- 熱・統計力学に基づいた自律分散的秩序形成技術の分析
  - 分布がボルツマン分布に従っていることを利用
  - 例：熱力学を用いて様々な大域的状態量の関係を調べる
- 様々な実ネットワーク制御への応用の検討
  - ネットワーク仮想化分野における応用の検討