

近接作用の概念に基づく ネットワーク制御手法の クラスタリングへの応用

KDDI研究所 久保 健、長谷川 輝之、長谷川 亨
UCI 須田 達也

背景

- 多数のコンピューティングデバイス（ノード）を使うような新しい利用方法（自律分散システム）を開発したい。
 - 多数のノード（PC、センサ、携帯電話等）がばらまかれ、それぞれが勝手に動く環境を想定
- 近接作用の概念に基づくネットワーク制御手法を提案した。
 - ネットワーク上に特別なノードの存在を仮定しない。
 - 各ノードはユニークな識別子を持たない。
 - 各ノードはグローバルな情報を持たない。
 - ネットワーク全体のトポロジなど
 - 隣接ノードと交換した制御情報に基づき自ノードの制御情報を更新する動作（＝近接作用）だけでネットワーク全体の振る舞いを制御する。
 - 近接作用を偏微分方程式で表現する。

クラスタリングアルゴリズムへの応用

- ネットワーク全体を自律分散的に均一なサイズにクラスタリングする。
 - クラスタのサイズ＝クラスタ内のノード数
 - マルチホップのクラスタが前提
 - クラスタヘッドから2ホップ以上離れたノードもクラスタのメンバになり得る。
 - ネットワークを2次元空間とみなせることが前提
 - 2次元 グリッドトポロジー、または
 - 2次元 perturbed (乱れのある) グリッドトポロジー
- メリット
 - 多数のノードで構成されるネットワークを階層化するための基本的なアルゴリズムになり得る。
 - クラスタリングによって、大規模なネットワーク内で、サービス・情報処理の局所化が可能になる。
 - 例：クラスタサイズを制御することによって、データベースサービスでのレプリカの配置数を制御する。
 - レプリカの数が多い(＝クラスタサイズが大きい)ほど、可用性や応答速度が高まるが、ノードはメモリを大量に消費する。
 - 形成されるクラスタのサイズを理論的に予測できる。
 - クラスタサイズが偏微分方程式の係数で決まる。

3

アウトライン

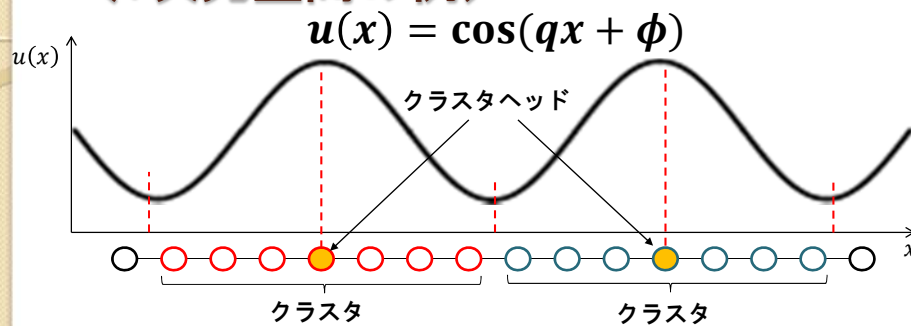
- 1次元空間の例を用いた方式説明
- 2次元空間への拡張
- シミュレーション結果

4

アウトライン

- 1次元空間の例を用いた方式説明
- 2次元空間への拡張
- シミュレーション結果

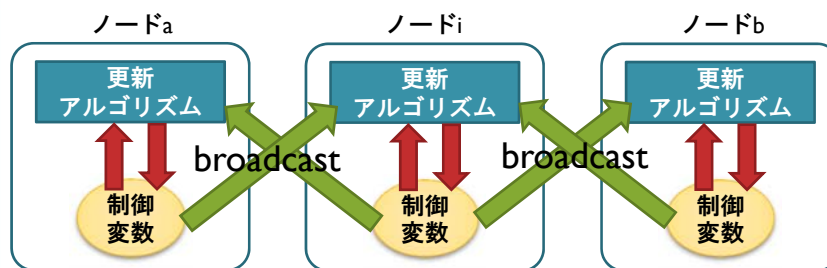
クラスタリングのアイデア (1次元空間の例)



- 各ノードは1次元ユークリッド空間上の格子点に存在する。
 - ただし、各ノードは自分のx座標は知らない。
- 各ノードが制御変数 $u(x)$ を持つ。
 - 各ノードがx座標を知ることなしに、 $u(x)$ がノード全体で正弦波になれば、均一なサイズのクラスタリングが可能。
 - パラメータ q がクラスタのサイズをコントロールする。

ノードモデル

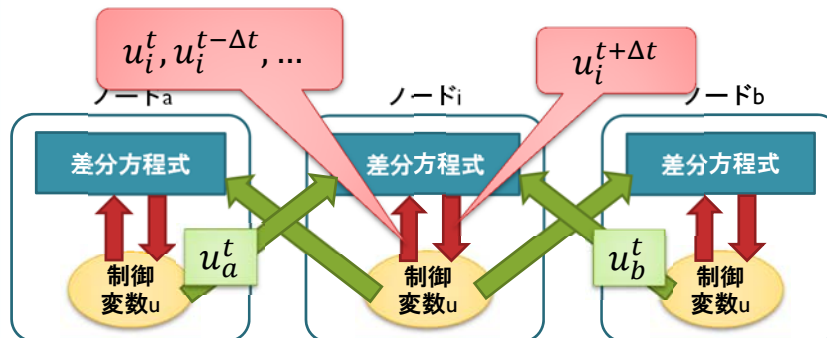
- ノードは制御変数の値を時々刻々更新する。
 - ノードiが更新に用いる情報
 - ノードiの現在および過去の制御変数値
 - ノードiの隣接ノードから受信した制御変数値
- ノードは隣接ノードとのみ通信する。
 - Broadcast



7

差分方程式による アルゴリズムの記述

- ノードは制御変数 u_i^t の値を Δt ごとに更新する。
 - ノードiが更新に用いる情報
 - ノードiの現在および過去の制御変数値： $u_i^t, u_i^{t-\Delta t}, \dots$
 - ノードiの隣接ノードから受信した制御変数値： u_a^t, u_b^t



8

KDDI KDDI P&A LABS

偏微分方程式を用いた動作アルゴリズムの記述

- 差分方程式（離散空間の式）
 - $f(\Delta t, \underline{u_i^{t+\Delta t}}, u_i^t, \overline{u_i^{t-\Delta t}}, u_i^{t-2\Delta t}, \dots, u_a^t, u_b^t, \dots) = 0$

時間間隔

自ノードの制御変数

隣接ノードの制御変数

近似

- 偏微分方程式（連続空間の式）
 - $g\left(u, \underline{\frac{\partial u}{\partial t}}, \frac{\partial^2 u}{\partial t^2}, \dots, \overline{\frac{\partial^2 u}{\partial x^2}}, \frac{\partial^2 u}{\partial y^2}, \dots\right) = 0$

自ノードの制御変数

隣接ノードの制御変数

9

KDDI KDDI P&A LABS

偏微分方程式を用いるメリット

- 数学的な取り扱いが容易
 - 関数 $u(x, t)$ は偏微分方程式の解となる。
- つまり、正弦波を解に持つような偏微分方程式が導出できればよい。
 - その偏微分方程式を離散化することで、均一なクラスタリングを実現するアルゴリズムが得られる。

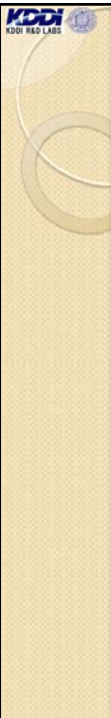
10



偏微分方程式の 初期値と定常状態の解について

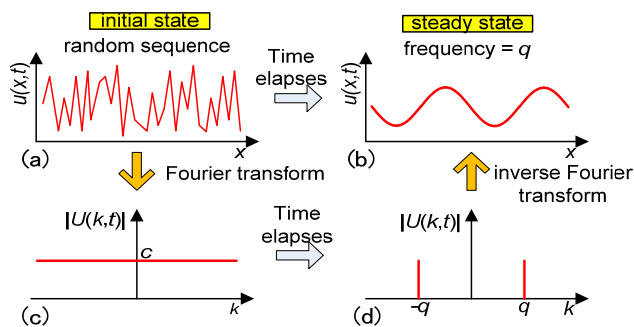
- 初期値
 - 各ノードに乱数を設定する。
 - ・ $u(x, 0)$ の分布はホワイトノイズになる。
 - ・ あらゆる周波数成分を含む
- 定常状態
 - $\lim_{t \rightarrow \infty} u(x, t)$ が周波数 q の正弦波となる。

このような条件を満たす偏微分方程式を導出する



周波数空間での考察

- 初期値と定常状態の周波数成分に着目する。
 - 必要な周波数成分だけを残すような偏微分方程式を求めればよい。
 - 周波数空間に変換すると、偏微分方程式を常微分方程式に変換できる。
 - ・ 簡単に微分方程式が導出できる。
 - ・ 導出過程は割愛

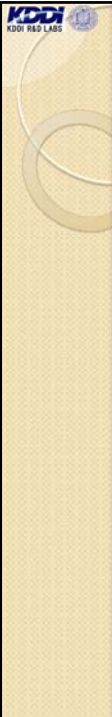


アウトライン

- 1次元空間の例を用いた方式説明
- 2次元空間への拡張
- シミュレーション結果

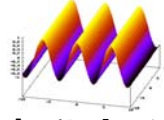
正弦波の2次元ユークリッド空間への拡張

- 2次元空間
 - ノードが2次元ユークリッド空間の格子点に置かれているものとする。
 - 各ノードが時刻 t に持つ制御変数を $u(x, y, t)$ とする。
- 2次元空間での正弦波とは？
 - $u(x, y, t) = \cos(qx)$ を考える。
 - x 軸方向に変動する波板のような形
 - さまざまな方向に変動する、さまざまな位相の正弦波を足し合わせる。
 - 各ノードは方向 (x 方向、 y 方向) がわからない。
 - さまざまな方向に変動する正弦波 (の重ね合わせ) を自律分散的に作ることになる。
 - 各ノードは原点を知らない (全ノードが合意した原点が存在しない)。
 - 自律分散的に作られた (さまざまな方向に変動する) 正弦波は、それぞれ原点が異なる。つまり異なる位相を持つ。

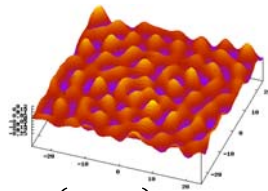
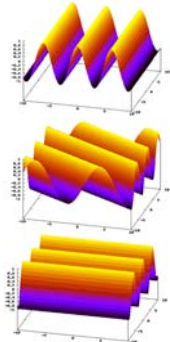


正弦波の2次元ユークリッド空間への拡張（つづき）

- $u(x, y, t) = \cos(qx)$



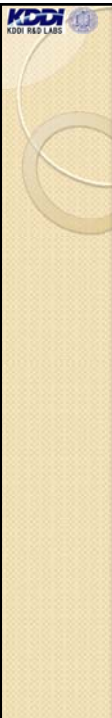
- さまざまな方向に変動する、さまざまな位相の正弦波を足し合わせる。



複数の山が形成される
(山=クラスタ)

φごとに決まるランダムな位相

$$u(x, y, t) = \int_0^\pi C \cos(qx \cos \phi + qy \sin \phi + \theta_\phi) d\phi$$



2次元空間で正弦波の重ね合わせを実現する偏微分方程式

- 偏微分方程式

$$\frac{\partial^2 u}{\partial t^2} + 2A(q^2 - \Delta) \frac{\partial u}{\partial t} + A^2 \Delta^2 u + 2A^2 q^2 \Delta u + A^2 q^4 u = 0$$

- ただし、 Δ はLaplacian ($= \frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2}$)を表す
- 差分方程式は以下の通り

$$\begin{cases} u_i^{t+\Delta t} = u_i^t + \Delta t v_i^t \\ v_i^{t+\Delta t} = v_i^t - \Delta t \left\{ A^2 q^4 u_i^t + 2Aq^2 v_i^t + 2A^2 q^2 w_i^{t+\Delta t/2} \right. \\ \quad \left. - 2A \sum_{n \in \text{neighbors}} \frac{v_i^t - v_n^t}{(\Delta d)^2} \right. \\ \quad \left. + A^2 \sum_{n \in \text{neighbors}} \frac{w_i^{t+\Delta t/2} - w_n^{t+\Delta t/2}}{(\Delta d)^2} \right\} \\ w_i^{t+\Delta t/2} = \sum_{n \in \text{neighbors}} \frac{u_i^t - u_n^t}{(\Delta d)^2} \end{cases}$$

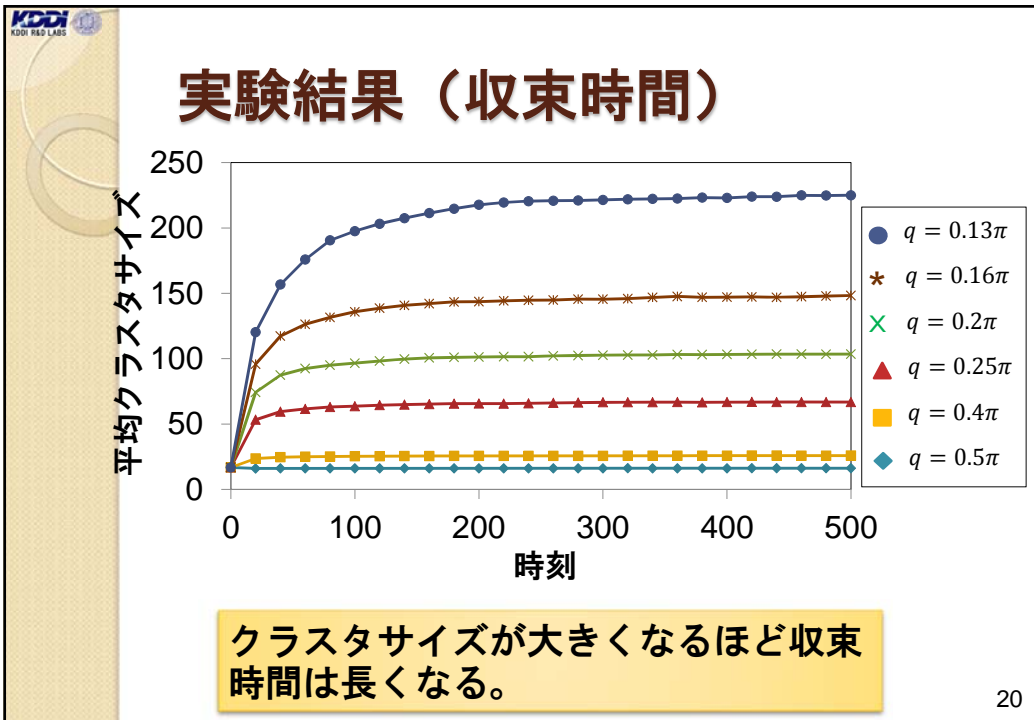
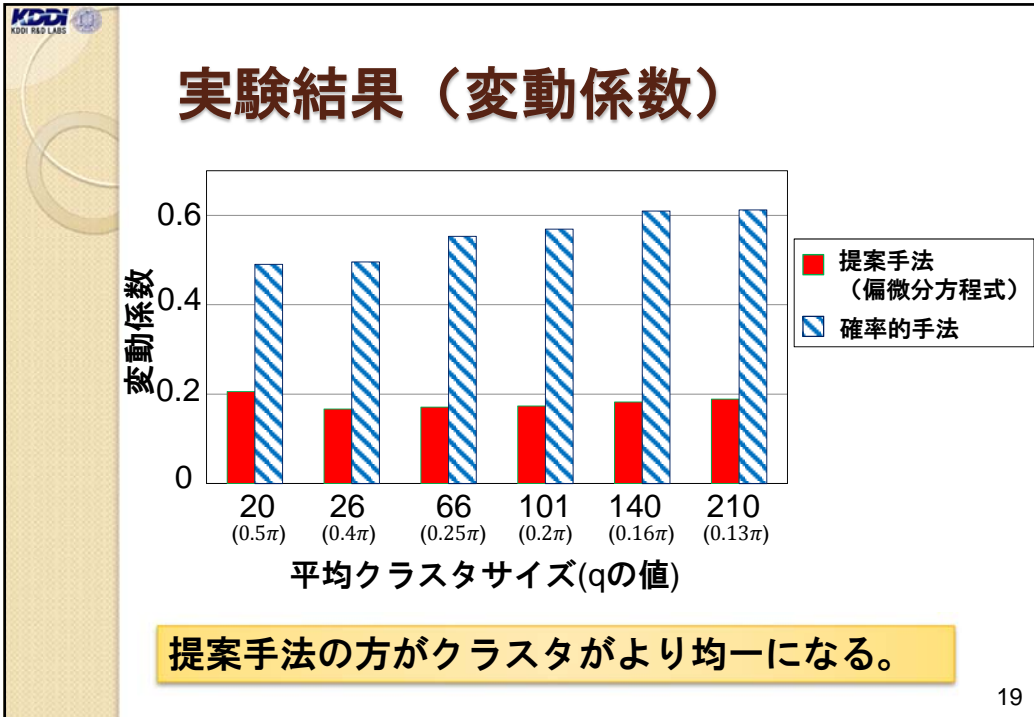
v_i^t および w_i^t は補助変数
(u_i^t に加えてこれらも隣接ノードと交換する)

アウトライン

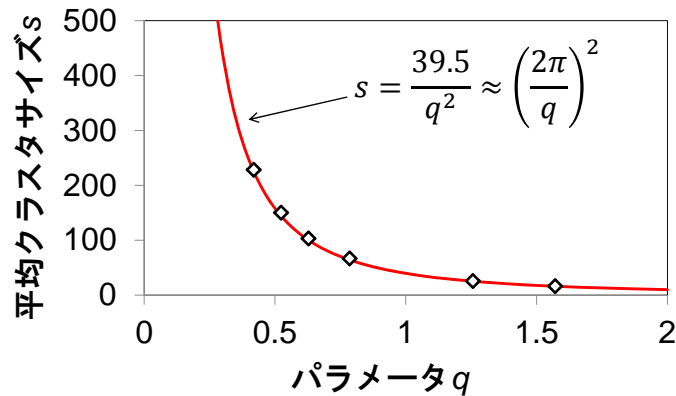
- 1次元空間の例を用いた方式説明
- 2次元空間への拡張
- シミュレーション結果

シミュレーション評価

- 実験条件
 - ノードを50x50のグリッド状に配置
 - 隣接ノード間距離 $\Delta d = 1$ 、更新時間間隔 $\Delta t = 0.1$
 - $q = 0.5\pi, 0.4\pi, 0.25\pi, 0.2\pi, 0.16\pi, 0.13\pi$
 - ホップ数：それぞれ4,5,8,10,12,15ホップに対応する
- 比較対象：確率的手法
 - 各ノードが確率 p でクラスタヘッドになり、その他のノードは一番ホップ数の近いクラスタヘッドのクラスタに参加する。
- 評価項目
 - 変動係数
 - 標準偏差÷平均クラスタサイズ
 - 0に近いほど均一である。
 - 各 q について、100回シミュレーションを行い平均および標準偏差を求め、変動係数を算出する。
 - 収束時間
 - 平均クラスタサイズの時間変動がある値を中心に $\pm 1\%$ 以下になった時点で、収束したものとみなす。



実験結果（パラメータ q と平均クラスタサイズ s の関係）



微分方程式のパラメータ q によってクラスタサイズを制御できる。

まとめ

- 近接作用の概念に基づくネットワーク制御手法の応用について提案した。
 - 均一なクラスタを形成する自律分散アルゴリズムを偏微分方程式で記述する。
 - 2次元ユークリッド空間にグリッド状に配置されたネットワークについて検討した。
 - 偏微分方程式（差分方程式）のパラメータ q によって平均クラスタサイズを制御できる。
 - 確率的手法に比べ変動係数が半分以下に抑えられる。
 - より均一なクラスタリングが可能

謝辞

- 本研究は、情報通信研究機構の「新世代ネットワーク技術戦略の実現に向けた萌芽的研究」の「副題：システムの大域的振舞いを生み出すための局所相互作用に基づく自律分散ネットワーク設計法」の一環として進めています。
 - 共同研究者の首都大学東京の会田先生、広島市立大の高野先生、UCBのWalrand先生には、有益なご議論およびご助言をいただきました。