

## 動的ネットワークにおける生態系パラダイムに基づく モバイルエージェント数制御

鈴木朋子<sup>†</sup>，泉泰介<sup>†</sup>，大下福仁<sup>†</sup>，増澤利光<sup>†</sup>

近年，分散システムの効果的な実現法として，モバイルエージェントを利用した設計法への期待が高まっている。モバイルエージェントを利用した分散システムでは，ネットワーク中のエージェント数が多いほど処理に要する時間が減少するが，システムの負荷は増加する。従って，ネットワーク中のエージェント数をネットワークの規模に対して適切に設定することが重要である。本研究では，動的に変化するネットワークにおいて，エージェント数をノード数の一定割合に保つ問題について考察し，生態系パラダイムに基づいた手法を提案する。また，提案手法がエージェント数をノード数に対して所定の比に保つことをシミュレーションによって示す。

### A Biologically Inspired Approach to Mobile Agent Population Control in Dynamic Networks

Tomoko Suzuki<sup>†</sup>，Taisuke Izumi<sup>†</sup>，Fukuhito Ooshita<sup>†</sup>，Toshimitsu Masuzawa<sup>†</sup>

Mobile agent is one of the most important paradigms to design distributed systems. In mobile-agent-based systems, larger number of agents generally attains shorter processing time but consumes more resources (e.g., processing power and network bandwidth). Therefore, it is indispensable to keep an appropriate number of agents for its application. In this paper, we consider the mobile agent population control problem in dynamic networks; it requires to adapt the number of agents to a constant fraction of the network size. We propose an algorithm inspired by a biological paradigm and show by simulations that the proposed algorithm efficiently adjusts the number of agents.

#### 1 はじめに

近年，ネットワークで接続された多数の計算機から構成される大規模な分散システムが実現されつつある。分散システムの大規模化，処理の複雑化，さらに，P2P，モバイルアドホックネットワークなどに見られるダイナミズムの高度化に伴い，分散システムの設計はますます困難になってきている。大規模で動的な分散システムの有効な設計パラダイムとして，モバイルエージェントを利用した設計法が有望視されている [5]。モバイルエージェント（以下，エージェントとよぶ）とは，ネットワーク中の計算機（ノード）間を移動し，他のエージェントと通信を行ないながら，協調的に自律動作するソフトウェアである。

現在，エージェントを利用したさまざまな分散システムの構築が考えられている。例えば，ネットワークを巡回して収集したノードや通信リンクの利用状況や動作状況に基づいてネットワークを管理する手法が提案されている [3, 7]。このようなシステムでは，一般に，ネットワーク中のエージェントが多いほど処理の並列性が高くなり，ネットワーク全体の処理に要する時間が減少する。その一方で，過剰な数のエージェントは計算機の処理能力やリンクの通信帯域をより多く占有することになり，システムの

負荷は増加する。従って，ネットワーク中に存在するエージェント数をネットワークの規模に対して適切に設定することが重要である。例えば，前述のネットワーク管理においては，エージェント数をネットワーク全体のノード数の約半数に保つことが望ましいことが示されている [3]。

しかし，エージェント数をネットワークの規模に対して適切に設定するのは，規模が動的に変化する大規模ネットワークでは非常に困難である。近年，大規模で動的な分散システムにおいて，スケーラビリティや動的変化に対する適応性を実現するために，生態系のメカニズムを模倣するというアプローチが提案されている。例えば，Bio-Networking プロジェクト [2] では，エージェントシステムのための生態系メカニズムを利用したプラットフォームの構築を目指している [6]。

そこで本稿では，ネットワーク中のノード数に対し，エージェント数を一定の割合に保つエージェント数制御問題について考察し，生態系メカニズムに基づく手法を提案する。提案手法は，生態学の分野で提案されているロジスティックモデルに基づいている。ロジスティックモデルとは，ある地域に一種類の生物だけが存在するときに，その生物の個体数が，環境によって定まる許容個体数に収束することをモデル化したものである [4]。提案手法では，エージェントを生物と見なし，ネットワークを環境と見なす。そして，ネットワーク中に存在できるエージェント数

<sup>†</sup>大阪大学大学院情報科学研究科コンピュータサイエンス専攻  
Department of Computer Science, Graduate School of  
Engineering Science, Osaka University

を、環境が提供する食物供給量で制御する。本稿では、提案するアルゴリズムがエージェント数をノード数に対する所定の比に保つことが可能であることをシミュレーションによって示す。また、エージェントを利用するアプリケーションでは、エージェント数だけでなく、各エージェントが生成されてから消滅するまでの生存期間が十分に長いことが必要であることが多い。そこで本稿では、提案手法における各エージェントの生存期間についても評価し、それが十分に長いこともシミュレーションによって示す。

## 2 諸定義

動的ネットワーク  $N(t) = (V(t), E(t))$  は、時刻  $t$  におけるノードの集合  $V(t)$ 、リンクの集合  $E(t)$  で定義される。ここで時刻  $t$  は自然数で表される離散時刻とし、 $V(t), E(t)$  は時刻  $t$  とともに変化する。リンクは異なるノードの非順序対であり、ノード  $u, v$  間のリンクを  $e_{uv}$  あるいは  $e_{vu}$  と表す。以降、ネットワーク  $N(t)$  のノード数を  $n(t) (= |V(t)|)$ 、リンク数を  $e(t) (= |E(t)|)$  と表す。

エージェントとは、ネットワーク中の異なるノード間を移動可能な、自律的に動作するソフトウェアのことである。時刻  $t$  にノード  $u$  に存在するエージェントは、 $e_{uv} \in E(t)$  である場合にのみノード  $v$  に移動可能である。また、訪問ノードにおいて、エージェントはノードの状態を参照し、書き換えることが可能である。また、各エージェントは新たなエージェントの生成、および自律的な消滅が可能であるとする。実際のエージェントシステムにおいては、アプリケーションによってエージェントの移動パターンが規定されているが、本稿ではエージェントはランダムウォークによって移動すると仮定する。すなわち、あるノードから各隣接ノードへの移動確率は等しいものと仮定する。エージェントの移動にランダムウォークを用いているエージェントシステムもいくつか存在する [3]。

各ノードと各エージェントは、正確に進む時計を持っているとする。ただし、異なるノード、およびエージェントの時計が同じ時刻を表しているとは限らない。

### 定義 2.1

ある値  $\gamma (0 < \gamma \leq 1)$  が各ノードに与えられているとする。このとき、以下の関係式を満たすようにネットワーク  $N(t)$  中のエージェント数  $A(t)$  を制御することが、エージェント数制御の目的である。

$$A(t) = \gamma n(t)$$

## 3 エージェント数制御アルゴリズム

本節では、エージェント数制御に対するアルゴリズムを示す。まず提案するアルゴリズムの背景となるロジスティックモデルについて述べる。

### 3.1 ロジスティックモデル

単一種の個体群のみからなる環境において、環境に応じたある値で、その個体数が安定することをモデル化したものが単一種個体群モデルである。個体数は、環境が供給する食物量に応じて変化する。個体が生存するためには、一定期間に一定量以上の食物を得ることが必要であり、一定量の食物を得ることができなければ、その個体は死亡する。環境が提供する食物量が個体数に対して少ない場合、いくつかの個体は死亡し、個体数が減少する。一方、環境が提供する食物量が個体数に対して多い場合、各個体は生存に必要な食物量以上の食物を得ることができ、子孫を生み、繁殖することが可能である。単一種個体群モデルでは、環境が提供する食物量に適した個体数で個体群が安定することが知られている。

単一種個体群モデルでの、時刻  $t$  におけるある個体数  $P(t)$  の変化率は、ロジスティック方程式として知られる以下の非線形 1 階微分方程式で表せることが知られている [4]。

$$\frac{\Delta P(t)}{\Delta t} = P(t) \cdot R(t) = P(t)(k \cdot F_a(t) - k \cdot F \cdot P(t))$$

ここで、 $F_a(t)$  は環境が個体群に対し提供できる食物量、 $F$  は 1 個体が生存に必要な食物量である。また、個体当りの変化率を表す成長率は  $R(t) = k(F_a(t) - F \cdot P(t))$  と表され、個体群の生存に必要な量以上に提供される食物量に比例する。ロジスティック方程式において、 $P(t) = 0$ 、または  $P(t) = \frac{F_a(t)}{F}$  である個体群は変化率が 0 であり、個体数に変化がなく、安定していると考えられる。個体数は平衡数  $P(t) = \frac{F_a(t)}{F}$  に収束し、平衡数に近い個体群においては成長率が 0 に近付き、個体数は安定する。

### 3.2 アルゴリズム

前節で紹介したロジスティックモデルに基づき、エージェント数を制御するアルゴリズムを示す。提案するアルゴリズムでは、エージェント群を個体群とし、ネットワーク  $N(t)$  のエージェントに対する環境収容力が  $\gamma n(t)$  になるように各ノードが食物を供給する。具体的には、各ノードは一定時間  $UNIT$  ごとに食物を供給し、各エージェントは一定時間  $UNIT$  ごとに、食物を一定量得ていれば存続、一定量得ていなければ消滅の判断を行なう。また、各エージェントは存続に必要な食物として消費されず余った食物から、新たなエージェントを生成する。これは、ロジスティックモデルにおいて、成長率が  $R(t) = k(F_a - F \cdot P(t))$  と表されており、エージェントに消費されず余った食物量によって新たに生成されるエージェント数が決まると考えられるためである。

各ノードと各エージェントの具体的な処理を図 1 に示す。以下、アルゴリズムの詳細について述べる。

ノードの処理：ノードは、制御すべきエージェント数に応じた適切な量の食物を時間  $UNIT$  ごとに供

給する。そこで、ノード  $v$  が時間  $UNIT$  ごとにエージェントに提供する食物量  $f(t, v)$  について考える。ネットワーク全体としてエージェントに供給する食物量  $\sum_{v \in V(t)} f(t, v)$  は、環境が一定期間に提供する食物量  $F_a(t)$  に相当する。時刻  $t$  にネットワーク中に存在するエージェント数を  $A(t)$ 、各エージェントが時間  $UNIT$  存続するために必要な食物量を  $AGT\_FOOD$  とすると、 $\frac{F_a(t)}{F} = \frac{\sum f(t, v)}{AGT\_FOOD} = \gamma n(t)$  が成り立つ。よって、ネットワーク全体として時間  $UNIT$  ごとに  $\gamma n(t) \cdot AGT\_FOOD$  分の食物を供給すればよいことがわかる。一方、エージェントが訪れる頻度はノードによって異なる。食物を一様に配置した場合、エージェントが頻繁に訪問するノードでは、訪問するエージェント数に対して供給する食物量が不足し、エージェントがほとんど訪問しないノードでは、供給する食物がエージェントに消費されることなく残る。そこで本手法では、エージェントに適切に食物を供給するために、エージェントが訪問する頻度に比例した食物量を各ノードが供給する。ノード  $v$  に 1 エージェントが存在する確率  $p(v)$  は、ネットワークが静的であるとすると、 $p(v) = \frac{deg_v}{2 \cdot e(t)}$  ( $deg_v$  は  $v$  に接続しているリンク数) と表すことができる。よって、ノード  $v$  において時間  $UNIT$  ごとに量  $\frac{deg_v}{2 \cdot e(t)} \cdot \gamma n(t) \cdot AGT\_FOOD$  の食物を供給する。

**エージェントの動作:** エージェントは生存に必要な量  $AGT\_FOOD$  の食物を得ていなければ、訪問ノードにおいて必要な食物を得、量  $AGT\_FOOD$  の食物をすでに得ていれば、それ以上の食物をノードから消費することはない。時間  $UNIT$  内に量  $AGT\_FOOD$  の食物を得ることができれば、そのエージェントは存続し、新たに訪問ノードから食物を得る動作を繰り返す。ただし、時間  $UNIT$  内に量  $AGT\_FOOD$  の食物を得ることができなければ、ネットワークから消滅する。各ノードは新たに食物を供給するときに、消費されずに余った食物を余剰食物として保持しており、エージェントは訪問ノードにおいて、この余剰食物を獲得してゆく。獲得した余剰食物の量が  $AGT\_FOOD$  に達するとエージェントを新たに 1 つ生成する。

上記の手法を用いることで、ノード数  $n(t)$  に対し、エージェント数が  $\gamma n(t)$  に安定することが期待できる。しかし、エージェントが訪問中のノードの離脱によって、ネットワーク中の全てのエージェントが消滅してしまう可能性がある。このような状況に対応するため、各ノードにおいてある一定期間 1 つもエージェントが訪問しなければ、ノードが新たなエージェントを 1 つ生成する。提案手法では、時間  $\frac{K}{\gamma n(t) \cdot p(v)}$  ( $K$ : 定数) の間 1 つもエージェントが訪れなければ、ノードが新たなエージェントを生成する。ただし、時間  $\frac{K}{\gamma n(t) \cdot p(v)}$  は時間  $UNIT$  より長いとする。

```

ノード  $v$  の処理
  food: 保持している食物量
  old_food: food に余った食物量 (余剰食物量)
  time: 保持する時計
  while (true) do
    if (time mod UNIT = 0) then
      old_food := food
      food :=  $\frac{deg_v}{2 \cdot e(t)} \cdot \gamma n(t) \cdot AGT\_FOOD$ 
    if (時間  $\frac{K}{\gamma n(t) \cdot p(v)}$  の間 1 つもエージェントが訪れない) then
      1 個のエージェント生成

エージェント  $a$  の処理
  eat_food: ノードの food 得た食物量
  clone_food: ノードの old_food から得た食物量
  time: 保持する時計
  while (true) do
    if (time mod UNIT = 0) then
      if (eat_food < AGT_FOOD) then 消滅
      else eat_food := 0
    if (ノード  $v$  に滞在) then
      if (eat_food < AGT_FOOD) then
        f := min{AGT_FOOD - eat_food, food}
        eat_food := eat_food + f
        food := food - f
      if (old_food > 0) then
        f' := min{AGT_FOOD - clone_food, old_food}
        clone_food := clone_food + f'
        old_food := old_food - f'
      if (old_food = AGT_FOOD) then
        1 個のクローン生成
        clone_food := 0
    if (v での処理が終了) then 次ノードに移動

```

図 1: ノード  $v$  とエージェント  $a$  の処理

#### 4 実験結果

提案したアルゴリズムをシミュレーションにより評価する。簡単のため実験では、ノードでの処理時間を 1、ノード間の移動時間を 2 とする。シミュレーション開始時の初期状況 (エージェントの位置、時計のずれ、ノード、およびエージェントの食物量) についてはランダムに決定する。一定期間エージェントが訪れない場合にノードがエージェントを生成する間隔  $\frac{K}{\gamma n(t) \cdot p(v)}$  について、実験では定数  $K = \frac{900}{UNIT}$  とした。本稿では、ネットワーク  $N(t)$  がスケールフリーグラフであるときの結果を示す。スケールフリーネットワークは、大半のノードは少数のリンクしか持たず、少数のノードが莫大なリンクを持つ構造 (ある  $k$  に対してリンク数が  $k$  本であるようなノード数がおおよそ  $k^{-r}$  ( $r$  は正の定数) に比例する) をしているネットワークである。スケールフリーネットワークは、実際のネットワーク構造を的確に模したモデルであるといわれている [1]。

時間  $UNIT$  を 200、ノード数を 100 で固定し、 $\gamma$  の値が 0.8, 0.5, 0.2 の場合それぞれについて、初期エージェント数を 100, 0 とした実験結果を図 2 に示す。この結果より、エージェント数が平衡数に短時間で近付き、その後、平衡数の近傍で安定していることがわかる。また、初期エージェント数が 0 の場合の結果からわかるように、エージェントが全て消滅した場合にも対応していることがわかる。

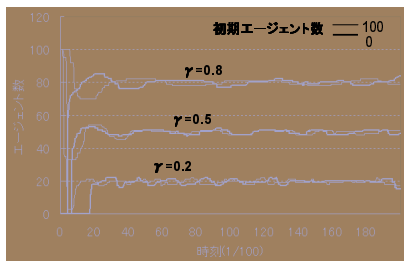


図 2: 静的システムでの実験結果

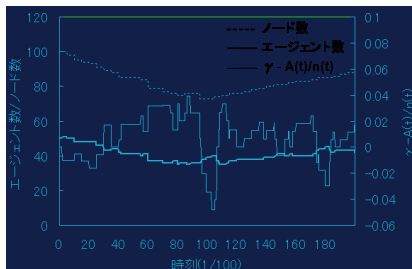


図 3: 動的システムでの実験結果

		UNIT		
		100	200	400
n	100	50	154	331
	200	51	191	591
	400	52	210	652

表 1: エージェントの生存期間 (UNIT)

つぎに、ノード数が時間とともに変わる場合の実験結果を示す。この実験では、初期状況において、ネットワークはノード数 100 のスケールフリーグラフであり、500 時間ごとに時刻 10000 までは確率 0.1 で新たなノードを加え、確率 0.02 で各ノードを離脱させ、時刻 10000 以降は確率 0.95 で新たなノードを加え、確率 0.0005 で各ノードを離脱させる。新たなノードが加わる時、ランダムに選択した  $k$  個のノードと接続され、ノードが離脱するとき、そのノードの接続リンクは消失するとする。時間  $UNIT$  を 200,  $\gamma$  の値を 0.5 としたときの実験結果を図 3 に示す。初期エージェント数は 50 である。この結果から、ノード数の変化に応じてエージェント数が適応していることがわかる。また、平衡数との差異を表す  $\gamma - A(t)/n(t)$  は  $\pm 0.04$  程度の差に収まっていることがわかる。

次に、エージェントの生存期間について述べる。エージェント  $a$  の生存期間  $lt_a$  とは、ある 1 エージェントがネットワーク中に存在している時間であり、消滅時刻  $td_a$  と生成時刻  $tb_a$  の差  $lt_a = td_a - tb_a$  で表される。エージェントの生存期間が長いことは、利用するアプリケーションにとって一般に有用である。今回、生存期間を各エージェントがネットワークに存在した  $UNIT$  数で表す。表 1 に各シミュレーションにおけるエージェントの生存期間の平均を示す。実験では、平衡状態の近傍に達した後のエージェントの生存期間を調べるために、初期エージェント数を平衡数とし実験を行なった。表 1 より、生存期間が時間  $UNIT$  に比例して長くなっていることがわかる。これは、 $UNIT$  が短いと、ネットワーク内に食物があるにも関わらず、時間内に訪問できないために消滅してしまうエージェントが多く存在するためである。よって、適切な  $UNIT$  を与えると、十分に長い生存期間が得られることが期待できる。

本稿では、スケールフリーネットワークでの実験結果を示したが、完全ネットワーク、ランダムネットワーク、ロリポップネットワーク、スターネットワークなどの他のネットワークトポロジでも同様の結果が得られた。

## 5 むすび

本稿では、動的なネットワーク環境において、ネットワーク中のノード数に対し、エージェント数を一定の割合に保つエージェント数制御の手法を提案した。また、提案した手法がネットワーク中のエージェントを適切な数に制御し、保つこと、およびエージェントの生存期間が十分に長いことをシミュレーションにより示した。

提案したアルゴリズムは、ノードにおけるエージェントの存在確率を得るため、時刻  $t$  におけるノード数  $n(t)$ , およびリンク数  $e(t)$  の値を各ノードに与えられていると仮定している。これらはネットワークに関する大域情報であり、特に動的ネットワークにおいては、これらの情報の利用は望ましくない。今後の課題として、局所的な情報からノード数、リンク数の密度を見積ることによる、大域的な情報を用いないアルゴリズムの考案が挙げられる。

謝辞 本研究の一部は、文部科学省 21 世紀 COE プログラム (研究拠点形成費補助金) の研究助成、および、日本学術振興会科学研究費補助金 (基盤研究 (B)(2)15300017) によるものである。

## 参考文献

- [1] R. Albert and A. L. Barabasi. Statistical mechanics of complex networks. In *Rev. Mod. Phys.*, Vol. 74, January 2002.
- [2] The Bio-Networking Architecture. <http://netresearch.ics.uci.edu/bionet/>.
- [3] R. R. Choudhury, K. Paul, and S. Bandyopadhyay. Marp: A multi-agent routing protocol for mobile wireless ad hoc networks. In *Autonomous Agents and Multi-Agent System*, January 2004.
- [4] R. Haberman. *Mathematical Model : Population Dynamics*. PRENTICE HALL, 1977.
- [5] V. A. Pham and A. Karmouch. Mobile software agents: An overview. In *IEEE Communications Magazine*, July 1998.
- [6] J. Suzuki and T. Suda. Design and implementation of a scalable infrastructure for autonomous adaptive agents. In *Proc. of the 15th IASTED International Conference on Parallel and Distributed Computing and Systems*, November 2003.
- [7] C. Du, Timon, Y. Li, Eldon, and C. An-Pin. Mobile agents in distributed network management. In *Communications of the ACM*, Vol. 46, July 2003.