

# グリッドによる マルチエージェントシミュレーションの分散処理

村田忠彦・松本優美・蟻川浩・山口正聡・小橋博道・門岡良昌



文部科学省私立大学社会連携研究推進拠点  
関西大学政策グリッドコンピューティング実験センター

Policy Grid Computing Laboratory,  
Kansai University  
Suita, Osaka 564-8680 Japan  
URL : <http://www.pglab.kansai-u.ac.jp/>  
e-mail : [pglab@jm.kansai-u.ac.jp](mailto:pglab@jm.kansai-u.ac.jp)  
tel. 06-6368-1177  
fax. 06-6330-3304

## **関西大学政策グリッドコンピューティング実験センターからのお願い**

本ディスカッションペーパーシリーズを転載、引用、参照されたい場合には、ご面倒ですが、弊センター（[pglab@jm.kansai-u.ac.jp](mailto:pglab@jm.kansai-u.ac.jp)）宛にご連絡いただきますようお願い申し上げます。

## **Attention from Policy Grid Computing Laboratory, Kansai University**

Please reprint, cite or quote WITH consulting Kansai University Policy Grid Computing Laboratory ([pglab@jm.kansai-u.ac.jp](mailto:pglab@jm.kansai-u.ac.jp)).

# グリッドによる マルチエージェントシミュレーションの分散処理

村田忠彦<sup>1,2</sup>, 松本優美<sup>2</sup>, 蟻川浩<sup>1</sup>, 山口正聡<sup>3</sup>, 小橋博道<sup>1,4</sup>, 門岡良昌<sup>1,4</sup>

## Distributed Multi-Agent Simulation on The Grid Computing

Tadahiko Murata<sup>1,2</sup>, Yumi Matsumoto<sup>2</sup>, Hiroshi Arikawa<sup>1</sup>, Masatoshi Yamaguchi<sup>3</sup>,  
Hiromichi Kobashi<sup>1,4</sup>, Yoshimasa Kadooka<sup>1,4</sup>

### 概要

マルチエージェントシミュレーション (MAS) は、様々な社会現象を仮想的に再現する方法の一つである。しかし、MAS を、単一の計算資源で処理する場合、エージェント数が多くなるにつれて、計算時間が長くなるという問題がある。そこで、本論文ではその問題を解決する方法としてグリッド・コンピューティングに注目し、グリッド上で MAS を行う場合の分散処理方法として Sugarscape モデルの複数の計算資源への分割を提案する。本研究では、環境改善資金調達問題を例題として採用し、分散処理の有無と、人口や汚染物質量、平均財産などの実験データの推移の類似性について、数値実験による検証を行う。その結果、分散処理の有無に関わらず、実験データが高い類似性を示すことがわかった。

### Abstract

Multi-Agent Simulation (MAS) is one of methods for social simulations. When implementing a MAS with a single computational resource, the more the number of agents, the longer the computation time becomes. In order to reduce the computation time for a MAS using Sugarscape model, we propose a distributed MAS on the grid computing. We show results of computer simulation for a social simulation: Financing model for preservation of the global commons. The simulation results show that there is no significant gap between the experimental data by the proposed method and a traditional method using a single computational resource.

キーワード：マルチエージェントシミュレーション，グリッド・コンピューティング，分散処理

Keyword: Multi-Agent Simulation, Grid Computing, Distributed Computing

---

<sup>1</sup> 関西大学政策グリッドコンピューティング実験センター

<sup>2</sup> 関西大学総合情報学部，<sup>3</sup> 関西大学大学院総合情報学研究科，<sup>4</sup> 富士通研究所

<sup>1</sup> Policy Grid Computing Laboratory, Kansai University

<sup>2</sup> Faculty of Informatics, Kansai University，<sup>3</sup> Graduate School of Informatics, Kansai University

<sup>4</sup> Fujitsu Laboratory

## 1. はじめに

マルチエージェントシミュレーション (MAS) は、環境と自律的な行動をするエージェントから構成され、多数のエージェントが相互に作用しながら、環境も変化するという特徴をもっている。MAS では人工的に社会を作り出すことができるため、人流や交通流、経済現象などの社会現象をコンピュータ上で再現し、ある社会現象について容易に分析することができる [1,2]。

しかし、MAS を単一の計算資源で処理する場合、エージェント数が多くなるにつれて、計算時間が長くなるという問題がある。この問題を解決する方法の一つとして、本論文ではグリッド・コンピューティングによる実装を考える。

グリッド・コンピューティングとは、地理的、組織的に広範囲に分散した複数のコンピュータをネットワークで接続、共有することにより、ユーザがそれらを仮想的な巨大コンピュータのように利用し、処理を行うことができるコンピューティング手法である [3,4]。このグリッド・コンピューティングによるシミュレーションを行う場合、どのように処理を分散させるかということを検討しなければならない。そこで、本論文では MAS を行う際の分散処理方法として Sugarscape モデルの分割を提案する。具体的には、Sugarscape モデルを分割することによって、異なる計算機間をエージェントが移動する場合に生じる問題に対する処理を提案し、分散処理を行う場合と分散処理を行わない場合の結果の類似性を検証するための比較実験を行い、提案した分散処理方法の有効性を示す。

## 2. Sugarscape による環境改善資金調達シミュレーション

本論文では、MAS の例として、西崎ら [5] によって考案された環境改善資金調達モデルというシミュレーションモデルを採用する。このシミュレーションモデルは、人工社会モデルの 1 つである Sugarscape モデル [1] を拡張したものである。本章では、Sugarscape モデルと環境改善資金調達モデルについて説明する。

### 2. 1 Sugarscape モデル

Sugarscape モデルは Joshua M. Epstein と Robert Axtell が開発した人工社会モデルで、エージェント、環境、ルールという 3 つの要素から構成されている。Sugarscape モデルでは、トーラス状の 2 次元空間上に資源が配置されており、エージェントは資源を求めて移動し、資源収集、代謝を繰り返す。

Sugarscape には、多数のエージェントが存在し、各エージェントは内部状態と行動ルールをもっている。エージェントの内部状態は、生涯を通じて不変である視力、代謝率、性別、初期財産、寿命、交配可能年齢と、状況によって変動する現在位置、財産、年齢とがある。代謝率とは、1 期間毎にエージェントが消費する資源量のことである。初期状態では、性別は男女数がほぼ等しくなるように定められ、各エージェントには初期財産が与えられる。エージェントは資源収集を行うと同時に代謝も行い、代謝量を差し引いた分の資源が財産として蓄積されていく。

また、Sugarscape は格子状の 2 次元空間で表現され、上下と左右がそれぞれ連続するトラス空間になっている。この 2 次元空間上にはエージェントが生きるために必要な資源が配置されており、エージェントに資源収集されても再生する。2 次元空間上の各地点は、資源の現在量と最大容量と、エージェントの存在有無を要素としてもつ。

エージェントと環境は、それぞれルールをもっている。エージェントとエージェントを結びつけるルールやエージェントと環境を結びつけるルールなどがあり、それらのルールに従ってエージェントは自律的に行動する。

エージェントの行動ルールの例を示す。各エージェントは 1 期間毎に 2 次元空間上の直交する 4 方向を視力に基づいて見渡し、最も資源が多く、他のエージェントがいない地点に移動する。そして、その地点にある資源を全て収集し、代謝しながら財産の増減を行う。この時エージェントの財産が 0 以下になれば、そのエージェントは餓死したものとして Sugarscape から取り除かれるが、その場合以外は寿命まで生き続ける。また、エージェントの移動後、そのエージェントが交配可能年齢であること、生まれた時にもっていた財産より多くの財産をもっていることを交配のための条件とし、その条件を満たしているエージェントは、近隣のエージェントをランダムに選択する。そして、選択されたエージェントも交配条件を満たし、かつ少なくとも一方のエージェントの近隣に子供を配置するための空きがあれば交配を行う。交配時には両親が生まれた時にもっていた資源の半分を子供に与える。

環境におけるルールとして、2 次元空間上の各地点の資源は、最大容量に達していない場合、単位期間あたり一定量ずつ再生される、と設定する。

## 2. 2 環境改善資金調達モデル

この環境改善資金調達モデル[5]では、環境改善のための資金調達システムとして、募金システムと慈善くじシステムを取り上げて、人口や汚染物質質量などの推移のシミュレーションを行っている。以下に、環境改善モデルにおける公害の定義と、公害対策にかかる二つの資金調達システム（募金システムと慈善くじシステム）について説明する。

### 2. 2. 1 汚染物質の発生

環境改善資金調達モデルでは Sugarscape 上に汚染物質を発生させることにより公害を表現する。エージェントの資源収集を生産活動とみなし、資源収集によって汚染物質が生じるものとする。資源が  $m$  単位収集される時、汚染物質が  $\lambda m$  生成されるとする。ある地点の時刻  $t$  における汚染物質の合計量  $p_t$  を式(1)に示す。

$$p_t = p_{t-1} + \lambda m \quad (1)$$

汚染物質は資源の再生率に影響を与える。表1のように蓄積された汚染物質質量に応じて、各地点の資源再生率が変化する。

現実社会では環境汚染が著しい場合、汚染を軽減するような措置をとる。これを考慮し、汚染物質質量に応じて表2のような公害発生率  $\lambda$  を設定する。

各エージェントの環境問題に対する関心度を長さ 20 で 0 と 1 から構成されているタグ文字列を用いて表現する。タグ文字列中に 1 のタグ数が多いエージェントほど環境問題に対する関心度が高いことを意味し、初期状態ではエージェントの各タグ位置にランダムに 0 か 1 を割り当てる。このタグ文字列中の 1 のタグの割合が 80%以上のエージェントを環境問題に対して強い関心があるエージェント (タイプ1)、40%以上 80%未満のエージェントを環境問題に対して通常に関心をもつエージェント (タイプ2)、40%未満のエージェントを環境問題に対してあまり興味のないエージェント (タイプ3) として分類する。なお、交配時には両親のタグ文字列の半分の位置を交差点として交叉手法を用いて子のタグ文字列が与えられる。

表 1: 資源再生率

| Pollutants $p_t$                                      | Recovery rate of resources |
|---|----------------------------|
| $10i \leq p_t \leq 10i + 10$ ( $i = 0, 1, \dots, 8$ ) | $100 - 10i\%$              |
| $90 \leq p_t$   | 10%                        |

表 2: 公害発生率

| Accumulated pollutants $p_t$ | Generation rate $\lambda$ |
|------------------------------|---------------------------|
| Under 50 units               | 0.001                     |
| Over 50 units                | 0.0005                    |

表 3: タグフリップ回数

| Pollutants $p_i$   | Number of tag flipping |
|--------------------|------------------------|
| $p_i < 10$         | 0                      |
| $10 \leq p_i < 30$ | 1                      |
| $30 \leq p_i < 60$ | 1 or 2*                |
| $60 \leq p_i < 80$ | 1, 2 or 3*             |
| $80 \leq p_i$      | 1, 2, 3 or 4*          |

\*タグフリップの回数はランダムに決まる

タイプ1のエージェントは他のエージェントに対して環境改善活動を勧める行動をとる。これを実現するために以下のタグフリップルールをエージェントの行動ルールに取り入れる。タイプ1のエージェントは近隣のエージェントの中からランダムに1人選択し、タグ位置をランダムに選択する。もし選択したタグ位置でタイプ1のエージェントが1で相手のタグが0ならば、相手のタグを1に変更する。また、表3のようにエージェントのいる場所の汚染物質量に依存してタグフリップの回数を変動させる。

### 2. 2. 2 募金システム

募金システムでは、エージェントは環境問題に対する関心度に従って確率的に財産の一部を寄付し、集まった資源は全て環境改善費用に割り当てられる。

エージェントはタイプによって予定募金額が異なり、タイプ1は3口(15資源)、タイプ2は2口(10資源)、タイプ3は1口(5資源)としている。

募金システムを導入した場合、以下のような手順でシミュレーションを行う。

Step 1 各エージェント  $i$  の募金することの効用値  $U_i$  を式(2)に従って計算する。

$$U_i = \alpha(I_i - x_i) + \beta k_i \log X_{i-1} \quad (2)$$

ただし、財産が代謝率の2倍に満たない場合は  $U_i = 0$  とする。  $I_i$  はエージェント  $i$  の財産、  $x_i$  は予定募金額、  $k_i$  はタグ文字列中の1のタグ数、  $X_{i-1}$  は前回の募金で集まった総資源量である。

Step 2 各エージェントは次の参加確率  $P_i$  に基づいて募金するかどうかを決定する。

表 4: タグ 1 減少ルール

| Rate $P_d$           | Type 1     | Type 2     | Type 3     |
|----------------------|------------|------------|------------|
| $P_d \leq 0.1$       | decreasing | decreasing | decreasing |
| $0.1 < P_d \leq 0.2$ | NA         | decreasing | decreasing |
| $0.2 < P_d \leq 0.3$ | NA         | NA         | decreasing |
| $0.3 < P_d$          | NA         | NA         | NA         |

$$P_i = \frac{U_i}{U_{\max}} \quad (3)$$

ここで、 $U_{\max} = \max\{U_i\}$  である。

- Step 3 実際に集まった資源  $X_i$  が環境改善費用となり、それに応じて Sugarscape 上から汚染物質を取り除く。1 資源に対して 0.005 単位の汚染物質を取り除くことができ、汚染物質量の多い地点から順にその地点の汚染物質を全て取り除いていく。この操作を集まった資源がなくなるまで繰り返す。
- Step 4 各エージェントは自身が募金したにも関わらず、募金への参加率  $P_d$  が低い場合はフリーライダー問題に直面し、環境問題に対する関心度を低下させる。つまり、タグ文字列のタグ位置をランダムに 1 つ選び、そこが 1 なら 0 に変更する。表 4 のようにエージェントのタイプによって 1 のタグの減少の仕方が異なる。募金に参加したエージェント数を  $a$ 、総エージェント数を  $n$  とすると、表 4 での募金の参加率は  $P_d = a/n$  となる。

### 2. 2. 3 慈善くじシステム

慈善くじシステムでは、エージェントはくじに当籤することに対する期待と環境問題に対する関心度に従って確率的に慈善くじの購入を行い、集まった資源の一部をくじの当籤金として投資者に与え、残りを環境改善費用に割り当てる。

慈善くじシステムを導入した場合、以下のような手順でシミュレーションを行う。

- Step 1 各エージェント  $i$  の慈善くじを購入することの効用値  $EU_i$  を式(4)に従って計算する。

表 5: 予定くじ購入口数の変動

|                | Type 1 | Type 2 | Type 3 |
|----------------|--------|--------|--------|
| after winning  | ±0     | +1 *   | +1 *   |
| after losing   | ±0     | -1 **  | -1 **  |
| not purchasing | ±0     | ±0     | ±0     |

\* up to 3 lottery tickets, \*\* up to the initial value

$$EU_i = \alpha(I_i - x_i + x_i\gamma) + \beta k_i \log\{(1-\gamma)X_{t-1}\} \quad (4)$$

ただし、財産が代謝率の2倍に満たない場合は0とする。  $x_i$  は予定くじ購入額、  $\gamma$  はくじ1口あたりの当籤金への初期還元率である。

Step 2 各エージェントは式(5)のくじの購入確率に基づいて慈善くじを購入するかどうかを決定する。

$$P_i = \frac{q}{\gamma} \frac{EU_i}{EU_{\max}} \quad (5)$$

この式の  $q$  は還元率であり、1期間前に購入されたくじの総口数  $S_{t-1}$  が多くなれば一定の限度まで還元率が増加する。これを式(6)に示す。

$$\begin{aligned} \text{if } S_{t-1} < 1000 & \quad q = \gamma + \frac{\gamma}{1000} S_{t-1} \\ \text{otherwise} & \quad q = 2\gamma \end{aligned} \quad (6)$$

Step 3 慈善くじを実施する。今期に購入された総口数を  $S_t$ 、1口あたりの当籤確率を  $\delta$  とし、実際に集まった資源  $X_t$  の  $100q\%$  を当籤したエージェントに均等配分する。つまり、1口あたりの当籤額を  $qX_t/\delta S_t$  とする。

Step 4 表5に従い、くじの当たりはずれによって各エージェントの次回の予定くじ購入額  $x_i$  を変動させる。予定くじ購入額の初期値は募金の場合と同じである。

Step 5 実際に集まった資源  $X_t$  の  $100-(1-q)\%$  が環境改善費用となり、それに応じて Sugarscape 上から汚染物質を取り除く。1資源に対して

0.005 単位の汚染物質を取り除くことができ、汚染物質量の多い地点から順にその地点の汚染物質を全て取り除いていく。この操作を集まった資源がなくなるまで繰り返す。

Step 6 各エージェントがフリーライダー問題に直面した場合は表 4 に従って環境問題に対する関心度を低下させる。

### 3. 提案手法

MAS では、エージェント同士が相互作用をするため、通信を頻繁に行わなければならない。しかし、グリッド・コンピューティングを実装する目的は処理時間の短縮であるため、時間がかかる通信はできる限り少なくする必要がある。本章では、通信回数を考慮したマルチエージェントシミュレーションの分散処理方法を提案する。

#### 3. 1 Sugarscape の分割

本論文では、Sugarscape を複数の区域に分割し、各区域の処理を各コンピュータに行わせることにより分散処理を行う。ただし、今回は仮想的に分散処理を行い、その結果が分散処理を行わない場合と類似性を示せるか否かということに焦点を当てる。

#### 3. 2 隣接区域との通信

3. 1 では、Sugarscape を複数の区域に分割すると記述したが、そのままシミュレーションを行うと、エージェントは初期区域だけでしか行動できなくなる。Sugarscape の分割数が増加するに従って、エージェントの動ける範囲が狭くなり、分割しない場合と比べエージェントの分布が大きく異なり、類似性が低くなると考えられる。そこで、各区域でエージェントが移動する前にあらかじめ隣接区域と通信を行い、隣接区域の一部分の環境情報を付加する。ここでの隣接区域の一部分とは、隣接する辺からエージェントの最大視力分までの奥行きを指す。

隣接区域からの環境情報を受信、付加した後、各区域でエージェントの移動を行う。区域を出たエージェントの情報は蓄積しておき、全てのエージェントが移動し終わってから蓄積されたエージェント情報を該当する隣接区域に送信し、移動前の区域でのエージェント情報を削除する。

これにより、各エージェントの移動毎に通信を行うよりはるかに通信回数が少なくなると言える。

以上の隣接区域との通信について図を用いて説明すると次のようになる。図 1 は 1 つの Sugarscape を 9 区域に分割した図である。分割はしているが、本来 Sugarscape は 1 つのトーラス状の 2 次元空間であるので、仮に区域 7 に

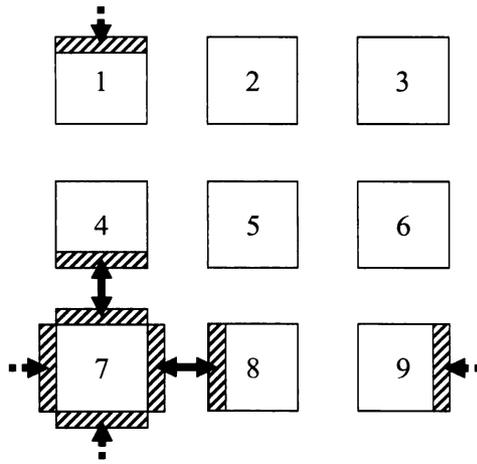


図 1: 隣接区域との通信例

注目した場合，上側の斜線部分には区域 4 の斜線部分の環境情報を，下側の斜線部分には区域 1 の斜線部分の環境情報を，左側の斜線部分には区域 9 の斜線部分の環境情報を，右側の斜線部分には区域 8 の斜線部分の環境情報を受信することになる．全エージェントの移動後は，区域 7 で上側の斜線部分に入ったエージェントの情報を区域 4 に，下側の斜線部分に入ったエージェントの情報を区域 1 に，左側の斜線部分に入ったエージェントの情報を区域 9 に，右側の斜線部分に入ったエージェントの情報を区域 8 に送信する．そして，隣接区域に出たエージェントの情報を区域 7 から削除する．

### 3. 3 エージェント衝突時の処理

隣接区域との通信により，区域を移動してきたエージェント情報を受信し，その情報に基づいて各エージェントを予定移動地点に配置する．しかし，その場合エージェント同士が衝突するという問題が発生することが考えられる．通常，エージェントが移動する時には必ず他のエージェントがいない地点を選択するので，複数のエージェントが同時に同じ場所に存在することはできない．だが，Sugarscape を分割することによって，区域を移動してきたエージェントを配置する前に，移動先の区域のエージェントは移動し終わっているため，予定移動地点にすでに他のエージェントが存在し，衝突してしまう．

その場合，本論文では元々その区域にいたエージェントを優先する．区域を移動してきたエージェントは予定移動地点の近隣をランダムに選択して移動する．もしどの近隣にも他のエージェントが存在する場合は，さらにそ

|           |           |           |           |
|-----------|-----------|-----------|-----------|
| (0, 0) 10 | (1, 0) 30 | (2, 0) 40 | (3, 0) 10 |
| (0, 1) 0  | (1, 1) 20 | (2, 1) 90 | (3, 1) 50 |
| (0, 2) 0  | (1, 2) 20 | (2, 2) 90 | (3, 2) 50 |
| (0, 3) 10 | (1, 3) 60 | (2, 3) 30 | (3, 3) 20 |

● : 区域を移動してきたエージェント      😊 : 元々区域にいたエージェント

(各セルの左上の数字は座標を, 右上の数字は資源量を表す)

図 2: 衝突時のエージェントの動作

の周囲を探し, 空いている場所が見つかるまで繰り返す. ここで, 区域を移動してきたエージェントは元の区域に戻るという選択はしないこととする. なぜなら, 元の区域に戻るためには再度通信を行う必要があり, エージェントの密集度が高い場合には頻繁に通信が行われてしまうという状況が生じるためである.

図 2 を用いて衝突した場合のエージェントの具体的な動きを説明する. 図 2 はある区域の様子を表している. 図中の座標(1, 2)で, 区域を移動してきたエージェントと元々その区域にいたエージェントが衝突している. したがって, 区域を移動してきたエージェントを移動させる. (1, 2)の近隣である(1, 3)にはすでに他のエージェントが存在するため, それ以外の近隣である(0, 2), (1, 1), もしくは(2, 2)を選択し, 移動させる. 仮に4箇所とも空いていない場合には, (1, 0), (3, 2)から空いている場所を探すことになる.

### 3. 4 シミュレーションの流れ

3. 1 から 3. 3 までの処理を踏まえたシミュレーションの流れを示す. なお, 下線部分は, 区域間 (計算資源間) の通信を行っていることを表す.

- Step 1 隣接区域の環境情報を受信
- Step 2 エージェントの移動
- Step 3 資源収集, 代謝  
(区域に残っているエージェントのみ)

- Step 4 汚染物質の発生
- Step 5 区域を出たエージェントの情報を隣接区域に送信
- Step 6 区域を移動してきたエージェントを配置
- Step 7 資源収集, 代謝  
(区域を移動してきたエージェントのみ)
- Step 8 汚染物質の発生
- Step 9 募金や慈善くじによる資金調達
- Step 10 汚染物質削減
- Step 11 資源再生

Step 1 から Step 11 までを規定回数繰り返す。

#### 4. 比較実験

本実験では、分散処理を行わない場合と分散処理を行う場合で環境改善資金調達シミュレーションを行い、実験結果に類似性を示せるかを検証する。

##### 4. 1 実験内容

150×150 の Sugarscape モデルとそれを9区域に分割した 50×50×9 の Sugarscape モデルを用いてシミュレーションを行う。ここでは、150×150 の Sugarscape でのシミュレーションが分散処理を行わない場合で、50×50×9 の Sugarscape でのシミュレーションが分散処理を行う場合である。つまり、150×150 の Sugarscape を1台のコンピュータで処理させ、それを9分割してできる 50×50 の2次元空間をそれぞれ仮想的に9台のコンピュータで処理させる。各 Sugarscape モデルでの初期エージェント数は3600、タイプ内訳はタイプ1が20%、タイプ2が60%、タイプ3が20%とし、資源の現在量が最大容量に満たない場合は1期間に最大容量×再生率分まで回復するというルールを用いる。また、各地点(x,y)の資源の最大容量 $R_{xy}$ は式(7)によって決定するものとする。 $Max = 0.5$ ,  $range = 0.00083$ ,  $ctr = 45$ である。

$$R_{xy} = 10Max \times \exp \left[ -range \times \left\{ (x - ctr)^2 + (y - 149 + ctr)^2 \right\} \right] + 10Max \times \exp \left[ -range \times \left\{ (x - 149 + ctr)^2 + (y - ctr)^2 \right\} \right] \quad (7)$$

環境改善のための資金調達システムを導入しない場合 (Open access : OA), 募金システムを導入する場合 (Voluntary contributions : VC), 慈善くじシステムを導入する場合 (Lotteries : LO) の3種類のシミュレーションを前述の2

つの Sugarscape モデル毎に 50 試行ずつ行い、平均をとる。なお、それぞれのシミュレーション期間は 2000 期間である。

実験におけるパラメータ設定は以下のとおりである。エージェントの内部状態としては、視力は 1 から 6 まで、代謝率は 15 から 35 まで、寿命は 60 から 100 まで、交配開始年齢は 12 から 15 まで、交配終了年齢は男なら 40 から 50 まで、女なら 30 から 40 までとする。初期エージェントの初期財産は代謝率の 1 倍から 3 倍とする。募金システムや慈善くじシステムで用いられるパラメータとしては、効用値を計算する際の財産に応じた効用  $\alpha$  は 0.1、環境からの効用  $\beta$  は 10、初期還元率  $\gamma$  は 0.15 とし、くじの当籤確率  $\delta$  は 0.03 と設定する。

#### 4. 2 実験結果

分散処理を行わない場合 (Not distributed : ND) と分散処理を行う場合 (Distributed : D) のエージェント人口、汚染物質量、平均財産、総資源量、募金またはくじへの参加率の推移を図 3 から図 7 に示す。また、それぞれ 200 期間毎の 2 乗誤差を計算し、50 試行に対して得られた平均 2 乗誤差とその平方根を表 6 から表 10 に示す。

分散処理を行わない場合と分散処理を行う場合とを比べると、どの図においてもほとんど差がない。また、2 乗誤差も低い値となっている。表 7 の汚染物質量の平均 2 乗誤差の平方根を見ると、一見値が高いように見えるが、図 4 の Open access での汚染物質量の最大値が約 270000 であるので、それと比較すると低い値であると言える。それぞれの推移の図と 2 乗誤差から類似性が高いと言える。このことから、本論文で提案した分散方法は有効性があると考えられる。

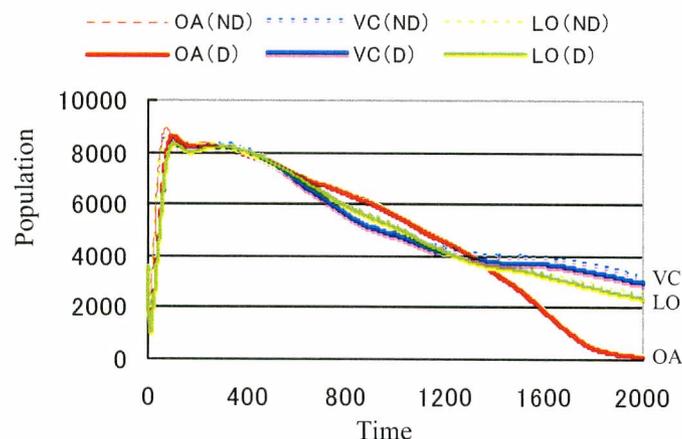


図 3: エージェント人口の推移

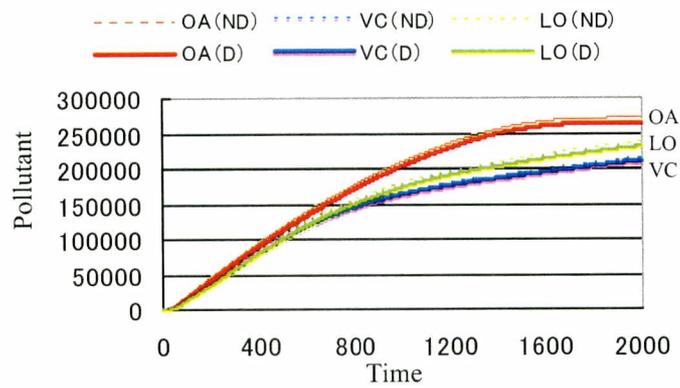


図 4: 汚染物質量の推移

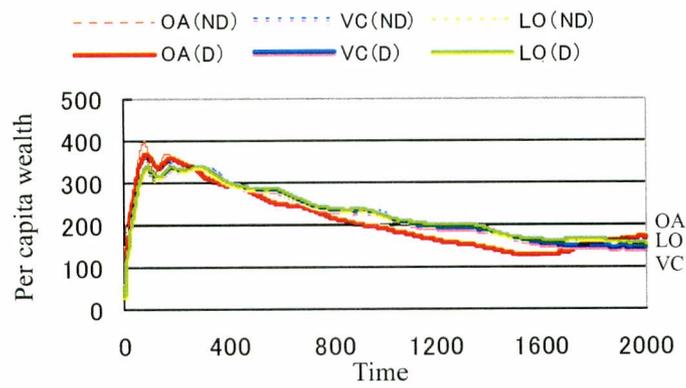


図 5: 平均財産の推移

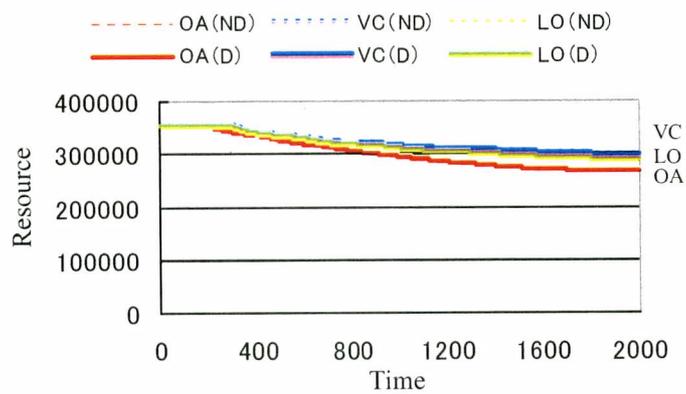


図 6: 総資源量の推移

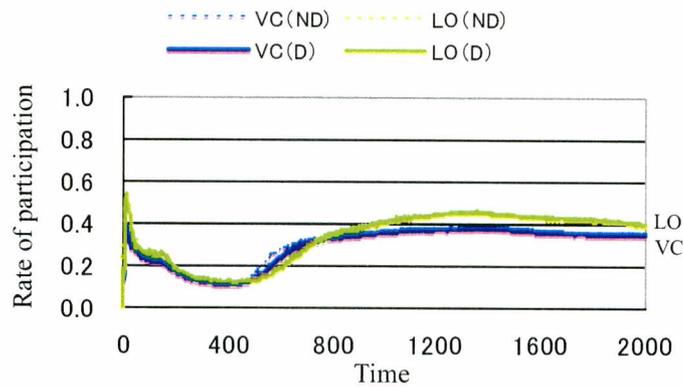


図 7: 参加率の推移

表 6: 人口推移の平均 2 乗誤差

| Simulation              | 平均 2 乗誤差 | 平方根   |
|-------------------------|----------|-------|
| Open access             | 6376.0   | 79.8  |
| Voluntary contributions | 33568.6  | 183.2 |
| Lotteries               | 20720.0  | 143.9 |

表 7: 汚染物質量の平均 2 乗誤差

| Simulation              | 平均 2 乗誤差   | 平方根    |
|-------------------------|------------|--------|
| Open access             | 33756261.0 | 5810.0 |
| Voluntary contributions | 7320382.2  | 2705.6 |
| Lotteries               | 25855154.8 | 5084.8 |

表 8: 平均財産の平均 2 乗誤差

| Simulation              | 平均 2 乗誤差 | 平方根 |
|-------------------------|----------|-----|
| Open access             | 4.6      | 2.1 |
| Voluntary contributions | 9.8      | 3.1 |
| Lotteries               | 20.0     | 4.5 |

表 9 : 総資源量の平均 2 乗誤差

| Simulation              | 平均 2 乗誤差  | 平方根    |
|-------------------------|-----------|--------|
| Open access             | 3280813.3 | 1811.3 |
| Voluntary contributions | 521041.7  | 721.8  |
| Lotteries               | 2559047.1 | 1599.7 |

表 10 : 参加率 (%) の平均 2 乗誤差

| Simulation              | 平均 2 乗誤差 | 平方根 |
|-------------------------|----------|-----|
| Voluntary contributions | 0.0      | 1.3 |
| Lotteries               | 0.0      | 0.9 |

## 5. おわりに

本論文では、グリッド・コンピューティングによって MAS を行う前段階として、Sugarscape の分割、エージェントの移動に必要な情報の通信をすることにより、仮想的にシミュレーションの分散処理を行った。数値実験により、分散処理を行った場合と分散処理を行わない場合とで、類似した推移を示すことが確認された。

今後は、通信回数の削減や他の MAS への適用を行う必要がある。また、本論文では、シミュレーションのみで、グリッド化の手法における実験データの比較を行ったが、提案手法を実際のグリッド・コンピューティング・システムに導入し、実験データを検証する必要がある。将来的には、大規模な MAS のグリッド・コンピューティングによる分散処理を実現する。

## 参考文献

- [1] Joshua M. Epstein, Robert Axtell (服部 正太, 木村 香代子訳), 人工社会 - 複雑系とマルチエージェント・シミュレーション -, 共立出版(東京), 220 pages (1999).
- [2] 並列分散処理研究室 (筑波大学大学院システム情報工学研究科), <http://www.padc.cs.tsukuba.ac.jp/index.html> (2004).
- [3] 日本アイ・ビー・エム システムズ・エンジニアリング株式会社, グリ

ッド・コンピューティングとは何か, ソフトバンク・パブリッシング株式会社 (東京), 275 pages (2004).

- [4] 溝口 文雄, グリッドコンピューティング 情報処理の新しい基盤技術, 岩波書店 (東京), 145 pages (2005).
- [5] 西崎 一郎, 上田 良文, 佐々木 智彦, “慈善くじによるグローバル・コモنزの保全のための資金調達と人工社会モデルを用いたシミュレーション分析”, システム制御情報学会論文誌, Vol.17, No.7, pp. 288-296 (July, 2004).