

## 人工社会を用いて創発現象を研究するためのルール学習の検討

西山 瑞紀† 山崎 誠治‡ 西尾 典晃‡ 武藤 敦子‡ 犬塚 信博‡

†名古屋工業大学 工学部情報工学科

‡名古屋工業大学 工学研究科 情報工学専攻

## 1 はじめに

人工社会とは、社会現象の創発を研究するためのマルチエージェントシミュレーションである。エージェント・環境・ルールによって構成されルールを変えることによってエージェント・環境の振る舞いが変わり、シミュレーションによって現象の創発を発見・研究する。

人工社会において、エージェントの行動ルールを学習する様々な研究がなされてきた。たとえば柴田ら [1] は、教師なし学習のうちの強化学習の一つである Q 学習を用いて人工社会でエージェントに自律的にルールを学習させた。また、松井ら [2] は論理的アプローチをとったエージェント学習への応用研究として、強化学習で得られた政策の一部を用い、環境が変わった際にも効率よく適応するために政策事前条件を定義し、概念学習によって事前条件を学習した。

本稿では、人工社会での実験に論理的アプローチを用い言語バイアスの制限下で一階述語論理によってルールを記述し、コントロールされた仮説空間内で網羅的にルールを変えて実験を行う枠組みを提案し、実装を行い適用例を用いて評価を行った。

## 2 SugarScape モデル

Sugarscape モデルとは人工社会の一つでエージェントが食べる食料を配置した空間分布のモデルである。環境は二次元座標で表され、座標それぞれに砂糖がある。エージェントは行動ルールに従い行動し移動先の座標から砂糖を集める。全エージェントが一度行動することを一ターンとしたとき、モデルでは一ターン経過毎に、エージェントはそれぞれ固有の消費率に従って砂糖を消費し、環境は各座標一斉にある比率で砂糖の保有量を回復する。

また、各座標毎に砂糖の最大容量が設定されており、エージェントは砂糖の消費率が定まっている。そして、エージェントは集めた砂糖を蓄積する。

ルールによっては公害による汚染物質やスパイスといった環境の要素や年齢や性別などのエージェントの要素を加える場合もある。様々なルールでエージェントや環境の複雑な動きがみられ、他にも多種多様なルールの記述により交配・戦争・取引・疾病の感染など様々な社会現象の創発を研究できる。

A Rule Learning Method for Study in Emergent Phenomenon in Artificial Society

†Mizuki NISHIYAMA, ‡Seiji YAMAZAKI, ‡Noriaki NISHIO, ‡Atsuko MUTOH, ‡Nobuhiro INUZUKA  
†Department of Computer Science, Faculty of Engineering, Nagoya Institute of Technology

‡Department of Computer Science and Engineering, Graduate School of Engineering, Nagoya Institute of Technology

## 3 ルール学習を用いた提案手法

人工社会を使った研究は環境を定め、その環境を観察することにより社会現象の創発を研究する。本稿では環境の定め方に対し論理的アプローチをとる。エージェントによる環境世界の観察や、環境への働きかけを述語で表現する。よって述語の組み合わせが環境を定めることとなる。

仮説は述語論理式で表され、仮説空間は仮説を記述する言語を決めることによって定まる。記述する言語の制約を決めるものを言語バイアスと呼び、これによって仮説空間に制限を加える。

また本稿での行動ルールとは、従来の Sugarscape モデルでのエージェントの行動ルールに加え、環境や身体的特性による行動への影響も述語によって表現したものである。本提案手法は、言語バイアスの制限による環境の設定、行動ルールの生成、シミュレーション、現象観察の順で機械的に人工社会を研究する枠組みである。



図 1: 枠組みの流れ

言語バイアスは使用できる述語の種類・引数のモード及び型に関する条件を与えることで、記述できるルールの範囲を制限する。ここでモードとは述語の各引数において入力、出力の別を表し、型は引数の取りうる領域を与える。例えば  $position(+Agent, -Point)$  は第一引数がエージェント型の入力、第二引数がポイント型の出力であることを表す。

言語バイアスに従って以下のようなエージェントの行動ルールを生成する。

$move(A, P) \leftarrow rule0(A, X_0), rule1(X_0), \dots, ruleN(X_k, P)$   
後件部はエージェントの番号 A と二次元座標の移動先 P を示す述語 move である。また前件部はバイアスの制限下で定められた述語の連言である。前件部によって移動先 P を定めるが、途中で fail した場合はそのエージェントはその場に留まることとする。

図 2 にルールの生成アルゴリズムを示す。背景知識集合は使用できる述語や、言語バイアスによる制限を表現したものである。アルゴリズムでは、バイアスによって制限された仮説空間内を幅優先探索で走査しており、出力条件を満たしたものを生成結果に追加していく。ここで出力条件とは後件部の第二引数であるポイント型の変数を、前件部の一番後ろに加えた述語の出力項と一致させる条件とする。

```

input      B : 背景知識集合
output    R : 生成結果

1. let R := ∅
2. let Queue = {move(A, P) ← .}
3. let Queue の先頭からルール r を取り出す
4. let L := ルール r の最後にバイアスの制限下で
   付加できるすべてのリテラルの集合
5. for each l ∈ L do
6.   add l をルール r の前件部の最後に追加
7.   if ルール r が出力条件を満たす
8.     then let R := R ∪ {r}
9.   let Queue の最後にルール r を追加
    
```

図 2: エージェントの行動ルール生成アルゴリズム

シミュレーションをする際には論理型言語を用いて Sugarscape モデルを実装しておき、生成したルールをすぐに実行できるように使用する述語の仕様をプログラム中に書き表す必要がある。

シミュレーションでは、生成した行動ルール集合からルールを一つずつ選び、選んだルールを全てのエージェントの行動ルールにして Sugarscape モデルを各ルール毎に実行することで機械的にシミュレーションを行うことができる。

## 4 提案手法を用いた評価

### 4.1 提案手法の適用例

実験では Sugarscape モデルと提案する実験環境を実装した。この環境を実際に動作させるための表 1 の述語をエージェントの行動ルール生成に用いた。ただし、fertile 以下の近傍とはマンハッタン距離による近傍であり、各述語は距離 1 の近傍を用いるものと距離 3 の近傍を用いるものの二つを用意した。たとえば fat の場合これを fat1, fat3 と表す。表 1 の A 及び P はエージェント型、ポイント型を表す。図 2 のアルゴリズムによって 100 個のルールを生成し、実験した。また実験での 100 個のルールについてそれぞれ経済の不平等

表 1: 実験に使用した述語一覧

| 述語名                      | 解釈                          |
|--------------------------|-----------------------------|
| <i>position(+A, -P)</i>  | A の座標 P を出力                 |
| <i>up(+P1, -P2)</i>      | 座標 P1 から各方向へ距離 1 の座標 P2 を計算 |
| <i>down, left, right</i> | up と同様                      |
| <i>empty(+P)</i>         | 座標 P が空か否か                  |
| <i>fertile(+A, -P)</i>   | A の近傍で最大肥沃座標 P を計算          |
| <i>harvest(+A, -P)</i>   | A の近傍で最大収穫座標 P を計算          |
| <i>rich(+A1, -A2)</i>    | A1 の近傍で最大財産エージェント A2 を計算    |
| <i>poor(+A1, -A2)</i>    | A1 の近傍で最小財産エージェント A2 を計算    |
| <i>fat(+A1, -A2)</i>     | A1 の近傍で最大消費エージェント A2 を計算    |
| <i>thin(+A1, -A2)</i>    | A1 の近傍で最小消費エージェント A2 を計算    |
| <i>hole(+P, -P)</i>      | P の近傍で周囲 9 マスが空の座標 P を計算    |

さを図る指数であるジニ係数及び全エージェント財産の和を軸にしたグラフを図 3 左に、各ルールでの 1000 ターン後のエージェント生存数の棒グラフを図 3 右に示す。

提案手法によるシミュレーションによって総資産が多い時

のジニ係数は 0.2 前後だということが観察できる。そして総資産やエージェントの生存数が一番多かったときの行動ルールをそれぞれの順位順に表 2 にまとめた。総資産順位 2 位のルールを例に挙げると、あるエージェントの近傍で一番低い代謝率をもったエージェントを探し、そのエージェントの近傍で現在砂糖量が一番多い場所を探すというルールである。いずれも harvest3,1 が行動ルールの前件部に入っており、有用なルールであることが予想される。

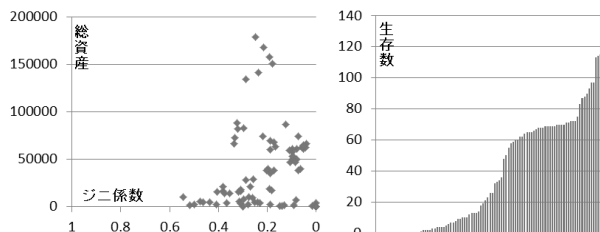


図 3: 左:ジニ係数と総資産の関係 右:生存数のグラフ

表 2: 高い性能を持った行動ルール

| 資産 | 生存数 | 行動ルール   |
|----|-----|---|
| 1位 | 1位  | <i>move(A, P) ← harvest3(A, P), empty(P)</i>                |
| 2位 | 4位  | <i>move(A, P) ← thin3(A, A1), harvest3(A1, P), empty(P)</i> |
| 3位 | 5位  | <i>move(A, P) ← fat3(A, A1), harvest3(A1, P), empty(P)</i>  |
| 4位 | 3位  | <i>move(A, P) ← poor3(A, A1), harvest3(A1, P), empty(P)</i> |
| 5位 | 6位  | <i>move(A, P) ← rich3(A, A1), harvest3(A1, P), empty(P)</i> |
| 6位 | 2位  | <i>move(A, P) ← harvest1(A, P), empty(P)</i>                |

### 4.2 利用に関する考察

100 個のルール群を生成するのに言語バイアスの制限下で述語長が 4 の深さまで探索をした。シミュレーションの結果、100 個のルールを 1000 ターン実行する実験は 45 分で終了した。述語の組み合わせから様々な行動ルールを容易に実験できる枠組みとなっていることを確認した。

## 5 おわりに

提案手法ではルール学習を用いて仮説空間内で網羅的に実験を行い、創発に有用なルールを発見することにより創発現象を研究する手法を検討した。仮説空間の制限により効果的な探索をすることができ、述語の組み合わせによって様々な振る舞いが見られる。

## 参考文献

[1] 柴田 淳子, 奥原 浩之 他, "Q 学習を組み込んだエージェントから構成される Sugarscape モデルによる人工社会", 電子情報通信学会, pp.35-43, 2007.

[2] 松井 藤五郎, 犬塚 信博, 世木 博久, 伊藤 英則, "強化学習結果の再構築への概念学習の適用", 人工知能学会論文誌, Vol.17, No.2, pp.135-144, 2002.

[3] Epstein, J. and Axtell, R., Growing Artificial Societies: social science from the bottom up, Brookings Institution Press, The MIT press, 1996.