1D4-05

モバイルエージェントにおける SOM を用いた階層的強化学習

Hierarchical Reinforcement Learning with SOM in a Mobile Agent

下山佐助[*1] 末田直道[*2]
Sasuke Shimoyama Naomichi Sueda

[*1] 大分大学大学院工学研究科知能情報システム工学専攻
Department of Computer Science and Intelligent Systems, School of Engineering, Oita University

[*2] 大分大学工学部
School of Engineering, Oita University

We are researching an Active Content Architecture that enable flexible contents distribution. Now we develop an agent who visits user sites negotiates to sell his contents. The agent learns strategic visiting knowledge using a reinforcement learning to visit user sites effectively. There are two important ideas in this paper. (1) A continuous value is converted into the discrete value by Self Organizing Map(SOM). (2) In order to reduce the massive action space that the agent is able to select, the agent has an autonomous hierarchy clustering algorism for the action values.

1. はじめに

我々は総務省のプロジェクトの一環としてコンテンツ自体がポリシーを持つアクティブコンテンツ[吉岡 03]に関する研究を行っている。アクティブコンテンツとは、利用者の視点から見ると、コンテンツを保持しその提供側のポリシーを保持し、かつ、自らが流通機能を持ったモバイルエージェントである。現在はそのアプリケーションの一つとしてネットワーク上を自律的に巡回し、消費者に対して商品の販売交渉を行う機構を開発している。しかし、どの消費者に対してどの商品を販売すればよいかを予め設計者が定義するのは困難である。そこで、本論文では教師信号なしで自律的に行動を学習する強化学習[RL 98]に着目し、アクティブコンテンツへ適用することで消費者への商品の販売方法を学習させる機構を構築する。そのアイディアとして、学習の際に自己組織化マップ[Kohonen 96] (以下 SOM)を使用して行動空間の分割と階層化を行うことによって、学習性能の向上を図る方式を提案する。

2. 問題設定

最終的には実世界上で実験を行う予定である、現在はその前段階としてシミュレータを使った実験を行っている。シミュレータはプログラムによって自動生成した消費者モデルから構成される。エージェントには流通エージェントとコンテンツエージェントの二種類が存在する。

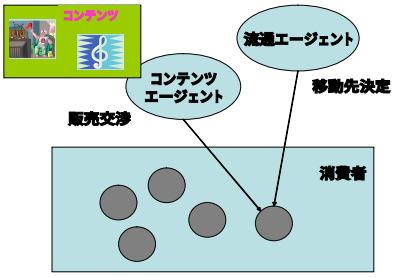


図 1 エージェントシステム構成図

- 流通エージェントは商品のリストを受け取り、商品の販売対象となる消費者を選択する。消費者を選択後、商品郡（コンテンツ）を持ったコンテンツエージェントを載せて、消費者の下へ移動する。学習には強化学習の代表的手法である Q-Learning[Watkins 92]を使用し、状態(商品の品揃え状態)における行動(移動先の選択)を学習する。
- コンテンツエージェントは流通エージェントに載って訪問した消費者との交渉を行い、商品を販売する。コンテンツエージェントは交渉戦略を学習する。
- 消費者モデルは離散値と連続値のパラメータを持っており、パラメータの値によって商品の好みが変わり、商品の購入確率が変動する。

本論文では流通エージェントに対象を絞って議論するため、以下で述べる実験では、コンテンツエージェントの交渉は、単純に全ての商品に対して交渉をする方式で行う(戦略的交渉は行わない)。

3. 学習手法

3.1 Q-Learning

Q-Learning とは、報酬に至るエピソードのステップごとに、以下の式を用いて、時刻 t における状態 s_t での行動 a_t の行動価値 Q を学習する。

$$Q(s_t, a_t) = Q(s_t, a_t) + \alpha [r_{t+1} + \gamma \max_a Q(s_{t+1}, a) - Q(s_t, a_t)]$$

ここで、γ (0≦ γ ≦1)は割引率、α (0≦ α ≦1)は学習率である。この手法は、環境がマルコフ性を満たすときに状態行動の価値が最適解へと収束することを保証している。

3.2 SOM

SOM は、入力層と出力層の二層構造からなるニューラルネットワークであり、入力層のノード(入力ノード)と出力層のノード(出力ノード)が完全結合している。また、各出力ノードは入力ベクトルと同次元の荷重ベクトル w を持つ出力ノード(勝者ノード) c を決定する。そして、勝者ノード c とその近傍ノード i に対して、荷重ベクトル w_i を次式によって更新する。

連絡先：下山佐助，大分大学大学院工学研究科，〒 870-1192 大分県大分市旦野原 700 番地，TEL:097-554-7866，FAX:097-554-7866，E-MAIL:pvf1242@csis.oita-u.ac.jp

$$w_i^{new} = (1 - \alpha(t))\, h_{ic}(t)\, w_i^{old} + \alpha(t)\, x$$

$$h_{ci}(t) = \exp\left(-\frac{\|r_c - r_i\|}{2\sigma^2(t)}\right)$$

ここで、t は時刻, $\alpha(t)$は学習係数 $0 \leqq \alpha(t) \leqq 1$ である。また、$\sigma(t)$は近傍の広がりを示し、$\alpha(t)$と $\sigma(t)$は学習の進展に伴い小さくする。学習の結果得られる荷重ベクトルの分布は、入力空間上の入力ベクトルの分布を近似する。つまり、入力ベクトルが一様に分布する場合は、SOM の荷重ベクトルも一様な広がりを持つようになり、入力ベクトルの分布に偏りがある場合は、荷重ベクトルもその偏りを持つ分布となる。

4. 提案手法

提案手法について述べる前に、今回の問題における状態と行動の定義について説明する。状態とは流通エージェントが持っている商品の品揃えであり、消費者に交渉して商品が売れると状態は変化する。行動は交渉対象となる消費者である。今回の問題設定においてネックとなるのが行動空間の大きさである。行動空間の大きさは消費者の数に等しいので、消費者の数が増えれば行動空間も増大し、学習が困難になることが予想される。

4.1 エージェントの構成

エージェントは状態認識器、学習器、行動選択器から成る。行動分割器は初期時間 t_0 において、選択可能な全行動の集合 $\boldsymbol{A}$ を受け取り行動選択器の構築を行う。状態認識器は時刻 t において環境から状態を観測し、一次元の離散値情報に変換して、S'_t を出力する。行動選択器は状態 S'_t を受け取り、行動 a_t を出力する。環境は a_t を受け取って変化し、報酬 r_t を返す。学習器は報酬を受け取り行動選択器の強化を行う。行動選択器には木構造の階層型行動選択を用いる。詳細は以降で述べる。

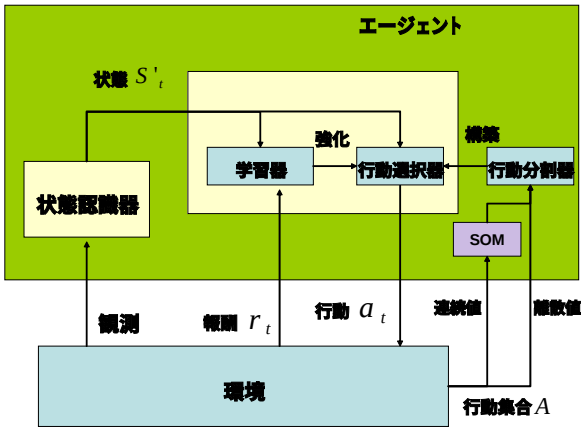


図 2 エージェントの構成

4.2 行動分割器

状態分割器は、消費者のパラメータを受け取ると、同一のパラメータを持つ消費者を同一のクラスに属するものとして、階層構造を構築する。パラメータによるクラスタリングは学習の前処理として行われる。つまり、消費者モデルが与えられた時点で行動空間が分割される。しかし、パラメータは連続値と離散値のものが混在しているため、このままではクラスタリングを行うことができない。そこで、連続値を離散値に変換することによってクラスタリングを行う。

4.3 クラスの自律的な分割

行動分割器を使ってクラスタリングを行ったとしても、消費者のパラメータの数が増えれば行動空間は巨大化してしまう。そこで、先ほどのクラスの上位にもう一つクラスを設定して三層の木構造で行動空間を構築する。上位クラス生成の為に離散の行動空間を自律的に分割し学習を行う自律分割機構を提案する。

行動空間の構成を行うために行動の類似度(similarity) σ を定義する。類似度は、全ての状態における行動価値の差の絶対値の総和から導かれる。行動 a_j と a_i の類似度は以下の式によって表される。

$$\sigma(a_i, a_j) = \sum_{s \in S''} \left| Q(s, a_i) - Q(s, a_j) \right|$$

ここで S'' は今までに経験した全状態である。木の構築アルゴリズムは以下の通りである。

(1)いままでに経験した全ての行動$a' \in A$から成る
行動リストALを生成する。
(2)*case*(a'が今までに一度でも選択されたことがある行動である){
$\min\sigma(a', a'')$となる$a'' \in A$を求める(ただし$a' \neq a''$)。
case(a'がクラスに属していない){
a'とa''を含むクラスcを生成する。
}
case(a'がクラスに属している){
a''をa'と同じクラスにする。
}
case(a'がまだ選択されていないか、a''が存在しない){
a'のみを含むクラスを生成する。
}
a'をALから取り除く。
(3)ALが空になるまで(2)を繰り返す。

木を構築する際には全ての行動について類似度を計算する必要があるため、行動空間が増えると学習毎に木が再構築され、実行時間が指数的に増えてしまう。しかし、一度しか構築しない場合、学習の結果が木の構成に大きく影響され、最初に構築した木が有効であるという保証がない。また、類似度を算出するためにはある程度学習が進んでいることが前提条件となる。そこで、更新率 η ($0 \leqq \eta \leqq 1$)を設定して一定の回数、学習を行った後に木の再構築を行うことにする。木が再構築されるサイクルは全エピソード数を T とした時 $T \cdot \eta$ で表される。更新率の値が多くなれば、生成した木の精度は高くなるが実行速度は低下してしまう。

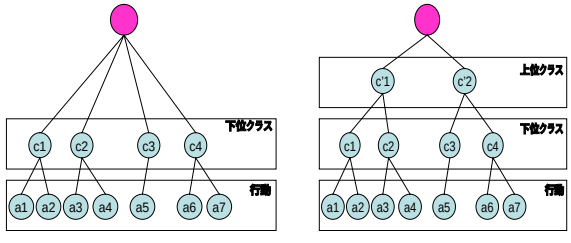


図 3 行動選択器の構造

図3は全行動数が 7 であると仮定した場合の行動選択器の構造を表している。左図は、消費者モデルのパラメータのみによるクラスタリングを行った階層構造であり、下位クラスと行動で構成される。右図は二層構造の行動選択器に対して、更に類似度によるクラスタリングを行ったものである。上位クラス、下位クラス、行動の三層で構成される。二層構造の行動選択器は予めクラスタリングが行われるので、構造が変化しないが、三層構造の場合は試行中に上位クラスの再構築が自律的に行われるため、構造が動的に変化する。

4.4 学習器

構造の違いによる学習方法を以下に示す。ただし、今回はどちらの場合でも、最下層の行動部分では学習を行っていない。

(1) 三層構造の場合の学習

下位クラスにおける学習は以下の式で表される。

$$Q(s_t,a_t)=\begin{cases} Q(s_t,a_t)+\alpha[r_{t+1}+\gamma\max_a Q(s_{t+1},a_{t+1})-Q(s_t,a_t)] \\ \quad a_t\text{が選択された行動である} \\ Q(s_t,a_t)+\alpha\zeta[r_{t+1}+\gamma\max_a Q(s_{t+1},a_{t+1})-Q(s_t,a_t)] \\ \quad a_t\text{が選択された行動と同一クラスに属する} \end{cases}$$

ここで ζ(0≦ ζ≦1)は選択された行動の影響度を表す。影響度は、実行された行動が同じクラスに属する別の行動に与える影響の度合いであり、この値が高くなれば学習の速度が増加する。このような学習を行う理由は、同一クラス内の行動は類似しており、選択された行動以外にも報酬を伝播することで、学習効率を向上させることができると考えたからである。ただし、学習の初期段階では、同一クラス内に属する行動が本当に類似した行動かどうかの判断が難しいため、なるべく低い値に設定する必要がある。

上位クラスの学習は以下の式によって行われる。

$$Q(s_t,a_t) = Q(s_t,a_t) + r_{t+1}$$

(2) 二層構造の場合の学習

二層構造では下位クラスのみの学習を行う。学習には通常の Q-Learning を使用する。

5. 実験

SOM による分割とグリッド分割の性能の違い、および自律的階層化を行う場合と行わない場合の性能の違いを調べる。

5.1 実験の設定

(1) 消費者モデル

実験にはプログラムによって発生させた 10000 人分の消費者データを使用する。消費者のパラメータは以下の通りである。

- 性別{男性,女性}(離散値)
- 年齢{15~65}(連続値)
- 年収{100~1000}(連続値)
- 職業{学生,会社員,無職}(離散値)
- 商品の嗜好{興味がある,普通,興味がない}(離散値)

ただし商品の嗜好は商品の数だけ存在する。今回は 10 個の商品データを使用するので商品の嗜好も 10 存在する。商品は以下の通りである。

- {ロック,クラシック,ジャズ,ブルース,メタル,演歌,ヒーリング,J-POP,トラッド,R&B}

消費者モデルのパラメータは流通エージェントが自由に見ることができる。ただし、見る事のできないパラメータに購入確率がある。購入確率は嗜好の度合いにより 5~95(%)の整数値で表され商品と同じ数だけ存在する。

(2) コンテンツエージェント

コンテンツエージェントは、流通エージェントから商品リストを渡され、消費者に販売を行う。商品リスト中の全ての商品が売れるかどうかを試し、商品が売れるか、全ての商品の販売に失敗した時点で交渉終了となる。商品が売れた場合は、商品リストの中から商品を取り除く。コンテンツエージェントは学習を行っていないので、エージェントの行動は一定である。

(3) 流通エージェント

流通エージェントは決められた試行回数だけ、エピソードを繰り返し学習する。エピソードが始まると、商品リストを与えられ、移動先を決定する。商品供給の方法は以下の二通りの方法を使用する。

- 固定選択:固定された5種類の商品を商品リストに加える。
- 無作為選択:10種類の商品の中から無作為に選ばれた 5 種類の商品を商品リストに加える。

固定選択は常に同じ商品が供給されるため、開始状態が一定である。それに対して無作為選択では、開始状態が一定ではないため、学習がより困難である。クラス選択には ε グリーディ方策を用いる。ただし、ε =0.2 である。1エピソードでは10回の交渉を行い、商品を全部売り切るか、最大試行回数に達した時点でエピソード終了となり報酬が与えられる。報酬には交渉を行うごとに-0.1 を受け取り、交渉が終了した時点で商品が一つでも売れていた場合 1.0 を受け取る。状態分割器には以下の二つを使用する。

- SOM(5*5)
- グリッド分割 (5*5)

グリッド分割では、与えられた連続値データを等分割して離散化する。自動分割アルゴリズムを使用する際の影響度 ζ=0.1、更新率 η =0.02 とする。

5.2 実験結果

(1) SOM とグリッド分割の比較

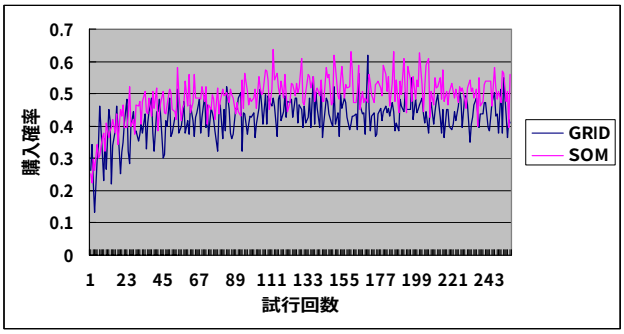


図 4 SOM とグリッド分割の比較

図４は行動分割器に、グリッド分割と SOM による分割を使用した場合の性能の違いを表したグラフである。自律分割機構は使用せず、試行は T=10000 回行った。グラフの縦軸は商品販売時の成功確率(*100%)であり、グラフの横軸は T/256 の間隔で実験結果の平均をとったものである。商品補充には固定選択を用いた。結果を見ると SOM を使用したほう収束が早く、交渉の成功率も高いことがわかる。これは、SOM によって多く存在

する部分のデータが密に、そうでない部分は疎に分割されることによって、グリッド分割と違い、無駄のないマッピングをしていることが原因であると思われる。しかし、それほど大きな差は見られない。これは、発生させた顧客データの偏りがあまり大きくなかったためであると考えられる。

(2) 自律分割機構の性能の検証

次に行動選択器に自律分割機構を使用した場合と使用しなかった場合とでの性能差の比較を行った。T=10000、行動分割には SOM を使用している。

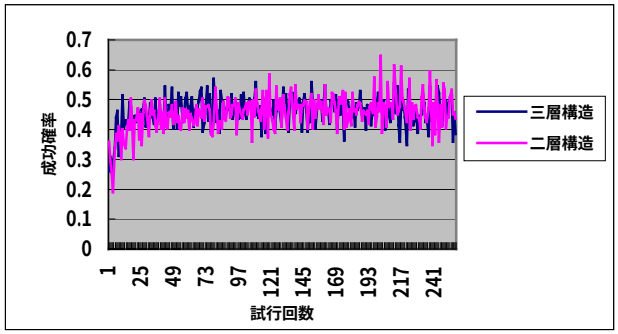


図 5 商品供給が固定

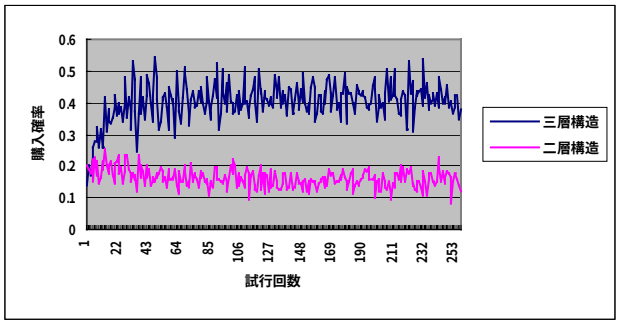
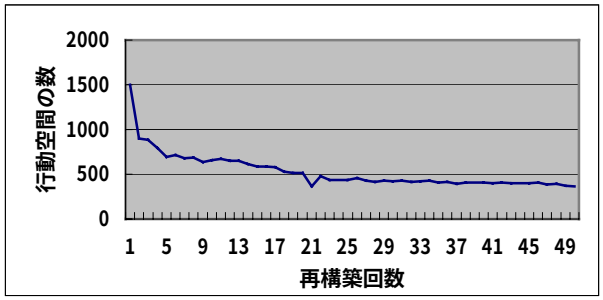


図 6 商品供給が無作為

実験の結果、シミュレータ上で商品リストに対する販売対象となる消費者の選択の学習に成功した。しかし、商品選択が固定的な場合は通常手法と階層化に大きな差異は見られなかった。これは、商品が固定のため環境が大きくならず、行動空間を分割するメリットがなかったためと思われる。しかし、商品選択が無作為の場合には、二層構造の手法では学習の成果が見られなかったのに対し、行動空間の自律的分割を行った手法では安定して学習を行えている。このことから、行動空間の自律的な分割が有効であることがわかる。



次に三層構造の場合、行動空間がどの程度まで削減されたかを見てみる。パラメータによるクラスタリングの結果、初期の 10000 の行動空間から 1500 まで減少する。更に自律分割機構によって 50 回の再構築のうち 20 回程で行動空間が 1/3 以下にまで減少し、その後はほとんど変化していない。

6. まとめ

今回の実験の結果、膨大な離散の行動空間において学習を行うことに成功した。連続値を離散値に変換する際には、グリッド分割を用いるよりも、データに応じて柔軟なマッピングが行える SOM が適していることがわかった。また、状態が複雑な場合には自律分割機構を使うことで学習効率を上げることができることもわかった。自律分割機構を使う際のデメリットは実行時間にある。しかし、木がかなり早い段階で構築されているので、行動空間の減少数が収束した時点で再構築を停止すれば、実行時間の短縮を図れるのではないかと考えている。

今後の課題としては、階層化手法の高速化、より現実に即した問題設定、Profit Sharing への自律分割アルゴリズムの適用、が挙げられる。また、今回自律分割機構の上位クラスでは非常に単純な学習機構を用いているため、学習の精度が悪くなっていることが考えられる。これに関しては、様々な学習機構による性能差を比較、検証することで、問題に適した学習機構を発見する必要がある。

参考文献

[Watkins 92] Watkins, C. J. C. H. & Dayan, P.: Technical Note: Q-Learning, Machine Learning , vol.8, pp.55-68(1992)

[RL 98] Richard S.Sutton,Andrew G.Barto: Reinforcement Learning, MIT press(1998):

[Kohonen 96] T. コホネン著, 徳高平蔵 岸田悟 藤村喜久朗訳: 自己組織化マップ: シュプリンガー・フェアラーク東京(1996)

[吉岡 03] 吉岡信和 田原康之 本位田真一 著: モバイルエージェントによる柔軟なコンテンツ流通を実現するアクティブコンテンツ: 情報処理学会論文誌(2003)