人　工　知　能　92－8
（1994．1．19）

入力属性空間の区分による強化学習法

幸島明男　仁木和久
電子技術総合研究所情報科学部認知科学研究室

本論文では、強化学習 (Reinforcement Learning) においてエージェントが学習する対象は、状況に応じた行動選択のためのポリシーというより、むしろ、「状況の見え方」と呼ぶ認識スキーマであると仮定する。この観点に基づいて強化学習を行なう方法について考察する。具体的には、入力属性空間の区分による一般化を行なう事例ベースの学習の導入によって、強化学習を実現する。

この学習法を「強化学習における入力の一般化」と呼ばれる問題に適用する実験を行ない、実用性を検討した。

Reinforcement Learning by Partitioning Input Feature Space

Akio Sashima　Kazuhisa Niki
Electrotechnical Laboratory
1-1-4,Umezono,Tsukuba-shi,Ibaraki,305,Japan

In this paper, we assume that an agent should learn a recognition schema by reinforcement learning. We describe a new reinforcement learning method from this point of view. In this learning method, we adopt "Method of seeing the situation" rather than a policy to select an action in each situation. We implement it by exampler based learning which generalize examples by partitioning input feature space.

We make an experiment to apply this learning method to "Input generalization problem in reinforcement learning" and report the result.

1 はじめに

人間や動物は試行錯誤を繰り返しながら、適切な行動を自律的に学習する。同時に、危険なものや役にたつものの存在を学習する。この学習過程においては、模範となる親や教師の行動の模倣という側面は弱い。自分の行動に対して環境から得られる信号（快・不快）によって、その行動と環境とが自分にとってどのような意味をもつのか、学習していく過程であると考えられる。

直接的な模倣例となるような教師が存在しない試行錯誤による学習のうち、環境から受けとる正・負の強化信号によって、状況に対する適切な行動の学習が自律的に進められていくものを、**強化による学習** (Reinforcement Learning)[1] と呼ぶ。

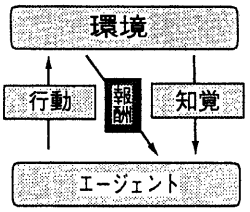

図 1: エージェントと環境の関係

従来の強化学習の研究は、環境に対する適応メカニズムという点が強調されており、エージェントの行動やそのパフォーマンスがどう変化するかばかりが重視されていた。そのため、エージェントがどのように状況の認識を行なっているのかという観点からは、あまり検討されてこなかった。そこで、本論文では、強化学習における状況の認識という点に着目し、その枠組の検討を行なった。

2 背景

強化学習は、その枠組の単純性やリアクティブな行動生成との融和性から、自律エージェントに行動を学習させる有望な方法として研究が進められている。

AIにおける強化学習の一般的アルゴリズムは、次のようなものである。

強化学習の一般的アルゴリズム

1. 現在の状況を認識したエージェントは、ポリシー（選択確率パラメータ）にしたがって、またはランダムに選択可能な複数の行動の中から1つの行動を選択する。
2. 選択した行動を実行し、その結果として強化信号を受けとる[1]。
3. その強化信号のフィードバックにより、選択に利用したポリシーを変更する。
4. 1にに戻る。

以上のように、強化学習は、この行動選択のためのポリシーである選択確率パラメータの学習と見られてきた。しかし、研究が進むにつれ、確率パラメータの学習だけを行なったのでは実世界での自律エージェントの行動の学習のような複雑な問題には適用が難しいことが明らかになってきた。

例えば、入力の一般化[2] (Input generalization)[2] や知覚的見せかけ[3] (Perceputual aliasing)[3] とよばれる問題が生じた。これらの問題は、簡単にいえば、現在の状況と過去に経験した状況との識別や類似性の判断をするにはどうしたら良いかという問題である。

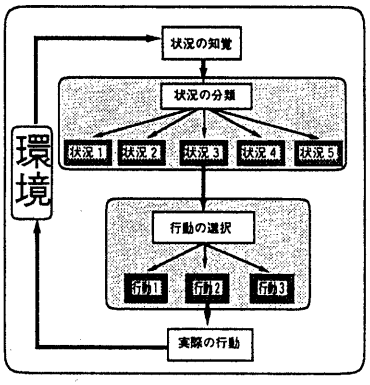

図 2: 従来の強化学習のモデル

従来の強化学習の枠組では、「エージェントは、現在の状況を知覚的に分類できさえすれば、意味的にも正しく分類できる」と想定する。そして、分類された状況に対応するポリシー（選択確率パラメータ）の学習だけを考える。ところが、知覚的類似性

[1]毎回、必ず強化信号を得られるとは限らない。ゴールに到達したときだけ、報酬信号が得られるような環境での強化学習を Delayed reinforcement Learning[4] と呼ぶ。本論文で扱う強化学習は Delayed reinforcement Learning である。

[2]はじめて経験する状況におかれた場合でも、過去に経験した状況を利用して、エージェントは行動を生成することができるかを問題にする。

[3]見かけ上は似ているが実際には異なる状況にであった時、エージェントは、それを区別できるかを問題にする。

によって状況を単に分類するだけでは解けないような問題は数多く存在する。状況の類似性は、知覚的な類似性だけに基づくものではないため、状況が知覚的に似通っていても、行なうべき行動は異なることは多い。そのような場合に対しても、従来の枠組を適用しようとすれば上記の問題が生じる。

これらの問題を解決するため、状況の類似性や一般性を判断する機構を強化学習に持たせようとする研究はすでに幾つか存在する [2][3]。しかし、そこで提案された方法では、状況を認識する機能と行動を決定する機能とは別々なモジュールとして存在しており、強化学習の枠組自体は従来のものと特に変わりはない。

このことから、本論文では、強化学習を「状況の見え方」と呼ぶ認識スキーマの獲得と仮定する観点から、その枠組自体の再検討を行なった。そして、行動の学習と状況の認識機構の獲得とを統一的に扱うための枠組を提案する。この枠組では、行動の学習は、状況の認識機構の獲得の結果として間接的に実現される。

3 状況の見え方の獲得による強化学習

3.1 状況の見え方の具体例

まず、「状況の見え方」がどのようなものか、既に「状況の見え方」を獲得している自律エージェントが、それを利用して行動する場合を例題にして述べる。

自律エージェントの移動 前と左右に距離センサーを持った自律エージェントを想定する。この自律エージェントは、行動に応じて起動する「状況の見え方」によって、状況を知覚し行動を制御する。この自律エージェントは、壁に当たると負の報酬、当たらずに移動できると正の報酬を得るとする。

行動に応じた「状況の見え方」 自律エージェントの目前に壁があるとする。「前に進む」という行動を行なおうとすると、この行動に対応して起動する「状況の見え方」によって、過去の類似状況において「前に進」んだ場合に負の報酬を得た経験を思いだす。そのため、現在の「壁」が負の報酬と結び付くものと見える。言い変えれば、現在の状況は「壁が障害となるから進めない」となる。

一方、ここで「右に曲がる」という行動を行なおうとした場合、過去の類似状況において、「右に曲が」った場合にも負の報酬を得た経験はないので、負の報酬とは結びつかない。言い変えれば、現在の状況は「障害物がないから曲がれる」となる。

自律エージェントは、行動に応じて異なる「状況の見え方」によって、結果的に適切な行動（この場合は「右に曲がる」）を行なう。

「状況の見え方」における属性の重み 自律エージェントは、行動に応じて起動する「状況の見え方」によって、重視するセンサーの重みを制御し、状況の類似性を判断する。例えば、「前に進む」時は対応する「状況の見え方」により、前向きのセンサーが敏感になる。右向きのセンサーの値が大きく異なるような状況でも、前向きのセンサーの値が類似していれば類似状況と判断する。センサーに対する敏感性は、他のセンサーの重みとの相対的な比率によって決定される。

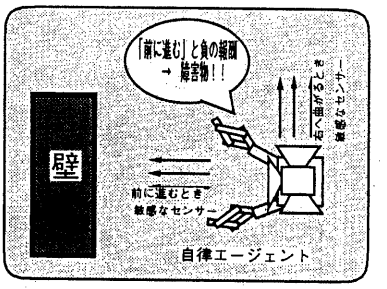

図 3: 自律エージェントの移動と「状況の見え方」

この例に示したように、「状況の見え方」は、行動によって得られる報酬という観点から状況を認識するためのものである。ある行動を行なおうとした時に、各行動に対応して自動的に起動される一種の認識スキーマとして考える。このような認識スキーマが獲得できれば得られる報酬が多く期待できる行動を選択するだけでエージェントは適切に行動できる。したがって、エージェントが、環境から得られる強化信号によってこの認識スキーマを獲得するならば、そのエージェントは強化学習を行なったことになる。

なお、例に示したような、行動に応じて重視するセンサーを変化させる機能を持つロボットはすでに実現 [5] されている。そのロボットは、あらかじめプログラムされた注意機構によって、現在の行動に応じた安全性の面から受け入れるセンサーを選択する。

本研究では、これを自律的に獲得しようというものである。したがって、自律的な学習によってプログラムが行なわれるため、プログラマが意図しなかったようなセンサーの重み付けを発見する可能性がある。

3.2 状況の見え方のモデル

例にあげた自律エージェントのように、得られる報酬という観点から状況を認識する機構のモデルを考える。

3.2.1 行動に応じた状況の認識

自律エージェントと「壁」の例で示したように、得られる報酬と言う観点から状況の認識を行なう。行動に応じて得られる報酬は変化するから、行動と独立に状況の認識だけを行なうことは不可能である。正しい状況認識をするためには、そこで行なう行動とペアで状況の認識を行なう必要がある。

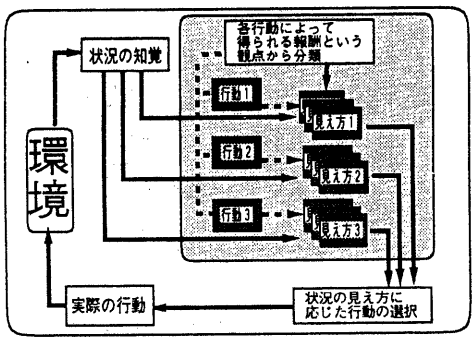

図 4: 「状況の見え方」による強化学習のモデル

そこで、図4のようなモデルを考える。このモデルでは、図2で示した従来の枠組のように状況の認識と行動の選択とを分割可能な別々なモジュールとは考えない。知覚した状況は、各行動に応じて異なる「状況の見え方」に変換される。その見え方のうちで、最も報酬を多く得られる見え方をした行動を選択する。

ここで提案した、行動に応じて認識する状況が変化する機構は、従来の強化学習の枠組では扱われてこなかったものである。

3.2.2 期待できる報酬の値に基づく類似性

図5で示すように、状況の類似性には、知覚的類似性と報酬に関する類似性がある。

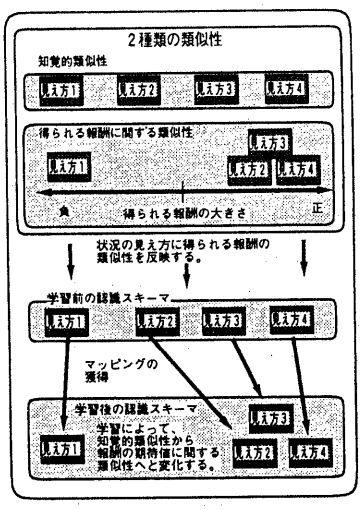

図 5: 状況の見え方

認識スキーマは、得られる報酬と言う観点から状況の認識を行なう。そのためには、行動とその結果得られる報酬の類似性を反映した分類を行なうことが必要である。そこで、従来は知覚的な類似性という面からだけ考慮されてきた「状況の類似性」を「行動によって期待できる報酬の値による類似性」という形で定義する。

図5で示すように、認識スキーマは、はじめは知覚的な類似性によって状況の類似性を判断する。しかし、学習が進むにつれ、得られる報酬の値が近い状況は類似した状況として分類するように変化する。これは、例えば、自律エージェントが「段差」の落差により負の報酬を得た後には、「壁」と「段差」が共に類似した状況に見えるようにするものである。

3.2.3 属性の重み付け

自律エージェントの例で述べたように「状況の見え方」は、行動に対応して状況のどの属性に着目すべきかを決定する。分類に役立たない属性がいくら知覚的に似通っていても、それは無視する必要がある。そのためには、分類基準の調整を行ない、分類に役立たない属性にはあまり注意を向けないようにする。類似性の判断を行なう時、属性の重み付けを行ない、分類に役立つ属性に関しては重みを大きくして、微妙な違いも見つけられるようにする。

この点で、本研究で考える属性の重み付けは、属性選択メカニズムとは異なる。ここで行なう属性

の重み付けは、属性値の相違に対する敏感性を制御するものである。値の相違に対する敏感性が鋭くなる場合は、通常の属性選択メカニズムには無い特性である。

4 属性空間の区分による強化学習法

「状況の見え方」の獲得を実現するために、属性空間の区分による方法をを提案する。

ここで提案する学習法では、事例ベースの学習 (Exampler based Learning) [6] で行なわれる方法を参考にして状況の分類を行ない、Q-Learning [7] と呼ばれる強化学習法を参考にして報酬の見積り値の生成を行なう。両者の融合することにより、状況の類似性の判断や報酬との関係を計算する「認識スキーマ」の機能を実現する。以下では、その融合の方法を具体的に述べる。

4.1 状況の分類問題としての強化学習

一般的な強化学習アルゴリズムは、強化信号 R によって、ある状況 x_i において、選択可能な行動の集合 a_n から 1 つの a_i を決定する最適なポリシーを表す関数 a_i=$\pi(x_i)$ を学習するものである。

Q-Learning[7] と呼ばれるアルゴリズムでは、$\pi(x_i)$ は、x_i で行動 a_k を行なったときの効用関数 $Q(x_i, a_k)$ が最大の値 $max(Q(x_i, a_k))(k = 0, \ldots n)$ となる行動を出力する関数であるとする。

$Q(x_i, a_k)$ の値は、行動と状況とが定まることによって決定される効用の見積り値を表す。ここでは、$Q(x_i, a_k)$ の a_k を固定し、効用の見積もり値をラベルと見なすことにより、状況 x_i を見積り値のクラスに分類する分類関数 $Q_{a_k}(x_i)$ として見る。ここで、$Q_{a_k}(x_i)$ は行動ごとに異なる効用関数を持つ[4]ものとして実現することを示す。つまり、Q-Learning の効用関数の式を、行動に応じて状況の見え方が定まることを表現する式であると考える。強化学習を状況の分類の問題とみることができた。

このように問題を定式化し、行動ごとに効用関数を持つ表現にすることにより、行動に応じて属性の重みを変えたり、状況の類似性を変化させることが容易になる。例えば、2 つの状況を一般化し、1 つにする時、従来の表現では、状況ごとに各行動に関する見積もり値を持っているので、各行動に関する値をそれぞれマージしなければならない。しかし、この表現では、行動ごとに状況の見積り値を持っているので状況に対応する見積もり値どうしをマージするだけで済む。

[4]従来の Q-Learning によるシステムの多くは、$Q(a_k, x_i)$ は状況ごとに異なる効用関数 $Q_{x_i}(a_k)$ を持つものとして実現されたものがほとんどである。

4.2 状況の分類と一般化

状況の分類と一般化は、入力された事例に簡単な一般化をほどこしてから (不可能ならばそのまま) 記憶しておき、推論時に利用すると言う事例ベースの学習 [6] (Exampler based Learning) として実現されている。

事例ベースの学習では、ある事例をどのクラスに分類するかを、すでに記憶してある事例との類似度 (属性空間上での距離) によって決定する。事例間に距離を定義し、分類すべき事例ともっとも近い距離の事例のクラスを、その事例のクラスとする。

本研究では、Salzberg によって提案された事例ベースの学習の 1 つである、Nearest HyperRectangle Learning[6] と呼ばれる学習法を修正して利用する。

4.2.1 Nearest HyperRectangle Learning

Nearest HyperRectangle Learning は、すべての事例をそのまま記憶しておくのではなく、隣あう 2 つの事例が同じラベルを持つ場合は、その二つの事例の持つ属性値による区間でもって超長方形 (Hyper Rectangle) を生成し、属性空間を区分する。新しく入ってきた事例の属性値がその超長方形の中に位置し、かつ同じラベルを持っている場合は、その事例は記憶しない。つまり、超長方形は事例の一般化の役割を果たしている。

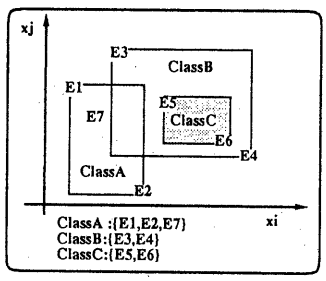

図 6: Nearest HyperRectangle Learning Method

新しく入ってきた事例の属性値がその超長方形の中に位置し、かつ異なるラベルを持っている場合は、その事例はそのまま記憶しておく。そして、その隣に同じラベルを持つ事例が入ってきた時に、超

長方形を生成する。そのため、重なりや入れ子構造をもった超長方形が作られる。

事例間の距離は以下のようにして計算する [6]。

$$D_{EH} = w_H \sum_{i=1}^{m} w_i \frac{diff_i}{m(max_i - min_i)}$$

$$\begin{aligned} diff_i &= E_{f_i} - H_{i_{upper}} & E_{f_i} > H_{i_{upper}} \\ diff_i &= H_{i_{lower}} - E_{f_i} & E_{f_i} < H_{i_{lower}} \\ diff_i &= 0 & otherwise \end{aligned}$$

E は新たな事例、H はすでに記憶されている事例を表す。$[H_{i_{lower}}, H_{i_{upper}}]$ が各属性の区分化された区間を表す。max_i, min_i は今まで現れた属性値の最大、最小値を表し、距離の正規化を行なっている。ここで、w_H は事例の重み、w_i は属性に対する重みを表す。

4.2.2 Nearest HyperRectangle Learning の強化学習への組み込み

Nearest HyperRectangle Learning は、クラスのラベルと属性値リストのペアが明示的に与えられる問題に対して適用するアルゴリズムである。ここでは、強化学習における報酬の見積り値をラベルとして、状況の知覚パターンを属性値リストとして与え、これを分類する。新しく入ってきた事例とそれに最も近い事例とのラベル（報酬の見積り値）を比較し、その差がある閾値 $K(0 < K < 1.0)$ 以下の場合は同じラベルと見なし、一般化を行なう。Salzberg の方法ではクラスのラベルは記号的表現しか扱わないので、これは本学習法における拡張である。

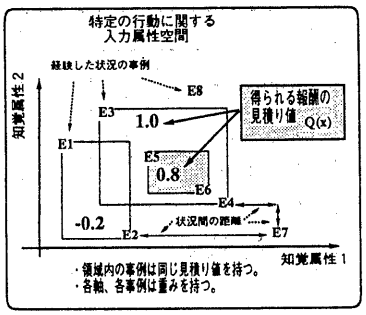

図 7:「状況の見え方」を表す入力属性空間

行動に対応して異なる属性空間において、状況を表す属性ベクトルがどの区分領域上の点なのか、またはどの事例と近いのか計算することによって分類を行なう。この状況の分類関数を $f_{a_k}(x_i)$ と表現すれば、状況の分類＝報酬の見積り：$f_{a_k}(x_i) = Q_{a_k}(x_i)$ となり、強化学習に Nearest HyperRectangle Learning を組み込むことが実現できる。

4.2.3 重み付けの方法

Nearest HyperRectangle Learning の距離の計算式で示したように、本学習法でも属性の重み付けと事例の重み付けを行なう。この重みを調整することにより、状況の類似性の調整ができる。

事例の重み付け

強化学習においては、属性値ベクトルと共に入力される報酬の見積りが、適切なラベルであるかどうかは保証されていない。したがって、適切でない超長方形が生成された場合は、事例の重みづけ [6] によってその影響を除く必要がある。

エージェントが行動を行なう時、その行動によって得られる報酬の見積り値を計算する。先に述べたように、この計算は過去の最も類似した事例 H のラベルから得る。

この計算が正確ならば、次の状況における行動に対する報酬の見積り値のうち、少なくともどれか1つの値は現在計算した値よりも大きくなるはずである。ところが、事例 H が過剰な一般化をしていた時などは、報酬の見積り値が逆に小さくなってしまうことがある。このような場合、事例 H を誤った不適切な事例であると考える。

不適切な事例（この場合事例 H）は、他のどの事例とも、その距離が離れるように事例の重みを大きくする。

具体的には、以下の式 [6] を用いる。

$$if(Q_{(t)a_k}(x_i) > max(Q_{(t+1)a_n}(x_j))) \quad then$$
$$WH(t+1) = WH(t) * K \;\; K(1.0 < K) \text{ は定数}$$

これにより、不適切な事例の影響を小さくすることができる。

属性の重み付け

行動に対応して作られる属性空間の属性の重みは、行動に応じて異なる値を持たせる。

状況の類似性に関係ない属性の距離は小さくし、関係ある属性の距離は大きくすれば、関係ない属性の距離に影響されることが少なくなり、正しく分類できると考えられる。そこで、新しい事例を記憶する場合に、記憶する事例 E と同じラベルを持つ区分化された事例 H とが最も近い距離に存在し、か

つ H の区分領域内には無いような場合に以下の式 [6] にしたがって、属性の重み付けを行なう。

$$if(H_{i_{lower}} \leq E_{f_i} \leq H_{i_{upper}}) \quad then$$
$$w_i = w_i * (1.0 - \nabla f)$$
$$else \quad w_i = w_i * (1.0 + \nabla f)$$

∇f は $0.0 \leq \nabla f \leq 1.0$ となる定数である。

この式は一見すると、＋ー が逆のようだが、類似性に関係無い属性が、関係ある属性よりも先に過剰な一般化を行なってしまうことが多いので、マッチした属性の重みは減らした方が良い結果が得られる。

4.3 ラベルの生成：報酬の見積り値の計算

事例は、状況を表現する属性ベクトルと報酬の見積り値のペアとして記憶する。ここでは、過去に経験した状況に対応するラベルとなる報酬の見積り値を、時系列を遡って作り出す方法について述べる。

Q-Learning と呼ばれる強化学習アルゴリズムでは、報酬の見積りを間接的に時系列に沿って伝播させるために以下のような方法を用いる。

時刻 t における $Q_t(x_i, a_k)$ の値は、

$$Q_{(t)}(x_i, a_k) = R(t+1) + \gamma * max(Q_{(t+1)}(x_j, a_n))$$

と表せる。ここで γ は、$0 \leq \gamma \leq 1.0$ になるような学習パラメータである。つまり、現時刻 t の見積り値は、次時刻 $t+1$ における直接的報酬 $R(t+1)$ と報酬の見積もりの最大値 $max(Q_{(t+1)}(x_j, a_n))$ とによって表せる。

通常の強化学習のアルゴリズムでは、現在の状況の報酬の見積もりの最大値 $max(Q(x_i, a_k))$ がわかった時点で、1つ前の時刻の状況における報酬の見積り値を計算し、記憶する。同様に、ここで提案する学習法でも、現在の報酬の見積り値がわかった時点で、間接的な報酬の見積り値を計算し、それをラベルとする事例として1つ前の状況を記憶する。

しかし、1つ前の時刻の状況の事例だけを生成したのでは学習が遅い。そこで、報酬を得た時点で、報酬の見積り値の絶対値が特定の閾値 (ここでは、0.2) を下回るまで、過去の履歴を遡って事例の生成を行なう [8]。ゴールから遡るにつれ、見積りは不確かになるので、閾値以下の値の事例は生成しない。

なお、事例の生成は、報酬を得た時点でバッチ的に行ない、インクリメンタルには行なわない。行動のフェーズと事例の生成のフェースとは分離している。事例の生成と行動の学習をインクリメンタルに行なう方法は、今後の課題である。

4.4 行動の選択確率の計算

行動 a_k に対する選択確率 $E(a_k)$ は次式 [8] によって計算する。

$$E(a_k) = \frac{\exp(Q_{a_k}(x_j)/\alpha)}{\sum_{i=1}^{n} \exp(Q_{a_n}(x_j)/\alpha)}$$

α は行動のランダムネスを制御するためのパラメータである。ここでは、0.25 とした。

5 音源を探索する課題

提案した学習法の実用性を調べるため、音源を探索する課題を行なった。

この課題を採用した理由は、**強化学習における入力の一般化の問題** [2] の具体例となっているという点である。

例えば、

- 状況の記述が実数ベクトルで与えられるため、全く同じ状況を二度体験するということはほとんどなく、新しい状況でも行動を選択できる必要がある。
- すべての状況を記憶することは無駄であり、分類する状態数の削減を行なう必要がある。

などの点である。

5.1 課題の内容

壁で囲まれた2次元空間の中を、エージェントが自由に移動する。そして、その空間の中に存在する音源を見つけるという課題である。エージェントは音源を見つけた時と壁に当たった時に強化信号を得る。音源は発見された時点で移動する。

5.2 学習環境

学習環境は壁で囲われた正方形の2次元実数空間である。エージェントは、この空間内を自由に移動する。図7において、A からスタートして音源 B を発見する。

エージェントは壁に激突した時、負の報酬 (-0.2) を受けとり、音源との距離が十分近く (空間を示す正方形の一辺の 1/10 の距離) になると、正の報酬 (1.0) を受けとる。正の報酬を受けとったら、自動的にスタート値点に戻される。200 ステップ[5]で音源を発見できなければ、やはり自動的にスタート地点に戻される。音源の位置は、エージェントがスタート地点に戻されるた時には必ず移動させる。

[5]一回の行動の実行が1ステップである。

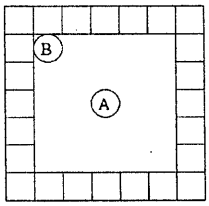

図 8: エージェントの学習する環境

5.3 エージェントの性能

エージェントは計5つのセンサーを持っている。センサーはすべて実数値を出力する。

障害物と自分との間の距離を計測する距離センサー 斜め右前向き、前向き、斜め左前向きの3つの距離センサーがある。このセンサーの計測範囲は、空間を示す正方形の一辺の 1/5 の範囲までである。それ以上の距離は、最大値（実例では 0.9999）を示す。

音波センサー 左右のセンサーに到達するまでの音の時間遅れを計算する方向センサーと音圧を測定する音圧センサーを持っている。この音に関するセンサーに関してエージェントは何の知識も持たない。方向センサーは左側から先に聞こえると＋、右側だと－の値をとる。音圧センサーは音源に近付くほど大きくなり、空間上のどこにいても観測可能である。

エージェントの知覚データは、以下の例のような属性値のリストとして与えられる。

内容：[音の方向、音の強さ、右 45 度の距離、前の距離、左 45 度の距離]

具体例：$[0.2365, 4.7184, 0.4802, 0.3388, 0.4779]$]

これが状況を表現する。このセンサーの値は、エージェントが1ステップの行動に対応したサイクルで更新される。

エージェントが1ステップで取れる行動は、

- その場で左に 30 度回転する行動：[left]
- その場で右に 30 度回転する行動：[right]
- 前に距離 D だけ進む行動：[go]

の3つである。距離 D は、空間領域の一辺の長さの 1/20 の距離としてある。つまり、一辺が $10m$ の空間では歩幅が $50cm$ であるようなエージェントを想定できる。エージェントは、各ステップで3つの行動のうちのどれか1つを必ず選択する。

5.4 エージェントの行動サイクル

エージェントの行動サイクルは、行動フェーズと学習フェーズに分かれる。その過程を簡単に説明する。

5.4.1 行動フェーズ

1. **知覚センサーからの入力を読みとる** ここで強化信号を読みとったら、学習フェーズへ入る。

2. **入力の分類** 得られた属性値ベクトルを、各行動それぞれに対応する入力属性空間上で、もっとも近い事例のクラスに分類する。ある閾値以内 (0.5) の距離に事例がない時は、ラベルが 0.0 のクラスに属するものとする。

3. **選択確率の計算と選択** 先ほどの式に基づいて、行動選択確率をそれぞれ計算する。

4. **事例の重み付け** 前ステップで予想した見積り値が誤っていた場合、計算に利用した事例の重みを下げる。

5. **実際の行動** [go,left,right] のどれかの行動を実行する。センサーの値と実行した行動は履歴として記憶しておく。

5.4.2 学習フェーズ

1. **報酬の見積り値の計算** 報酬の見積もり値を履歴を遡って計算し、その状況の属性リストとペアにする。

2. **事例の生成** 属性値リストを報酬の見積り値をラベルとしたによる事例を、その状況で行なった行動に対応する事例ベースに記憶する。属性空間上のある閾値以内 (0.5) の距離に、同じラベルの事例がない時は、点のまま記憶する。事例の一般化が可能な場合は超長方形を作る。

3. **属性の重み付け** 先に示した式にしたがって、属性の重みづけをする。ここでは、マッチした属性の重みは 0.8 倍し、マッチしない属性の重みは 1.2 倍する。

4. **履歴の記憶の消去**

6 実験結果

音源探索課題に関する実験を行なった。その結果を示す。

6.1 ステップ数の学習による変化

まず、本学習法によるエージェントが学習をしていることを示す。これは、エージェントが音源を見つけるまでにかかったステップ数（縦軸）の1トライアルごと[6]の変化（横軸）を示している。比較のためにランダムウォークによる平均も示した。

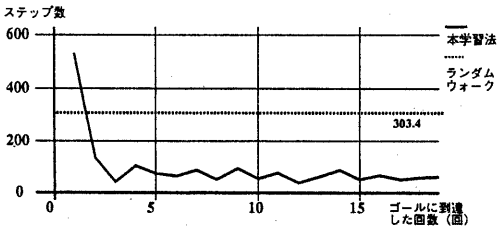

図 9: ゴールまでのステップ数（平均）の変化

学習後のゴールまでのステップ数の平均は、ランダムウォークを上回るようになっている。

実験結果は、強化信号に基づいて適切な行動の学習ができたことを示している。認識スキーマの獲得という観点からの強化学習法が、単に考え方に留まらず、実用性のあるものであることが示せた。

このグラフにおいて、最初のゴールに至るまでのステップ数の平均がランダムウォークによるものよりも劣る原因は、最初の試行だけは、音源の位置がいつも決まった位置に固定されているからである。グラフに示したランダムウォークの平均は、最初の試行は除いて計算している。

最初の試行に関してだけランダムウォークのステップ数の平均をとると 474.5 になり、本学習法によるステップ数の平均とほぼ同様である。

6.2 類似度の調整の効果

ランダムウォークするエージェントと、区分値による一般化も重み付けも行なわない事例を記憶するエージェント、事例と属性の重み付けを行なわない（区分値による一般化だけ行なう）エージェント、区分値による一般化と属性の重み付けだけ（事例の重み付けを行なわない）を行なうエージェント、そして、本学習法のエージェントの5つの比較をした。

1400 ステップの行動（横軸）によってゴールに到達した回数を累積し、その平均（縦軸）を比較した。

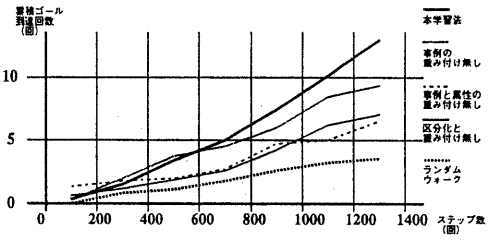

図 10: 累積ゴール到達数（平均）の比較

このグラフを見ると、単に区分値による一般化だけを行なっただけでは、事例をそのまま記憶するものと比較して効果がないことが分かる。

逆に属性、事例の重みを変化させた場合は、学習効果があることがわかる。特に事例の重み付けを行なうものは、誤った事例を除く効果があるため、属性の重み付けだけを行なったものに比べて到達回数の落ち込みが少なくなっている。

事例、属性の重み付けは、報酬に関する状況の類似性を「状況の見え方」にフィードバックするためのものであった。この実験結果は、事例、属性の重み付けが「状況の見え方」を変化させるのに有効であることを示している。

6.3 獲得した事例

ある試行において獲得した事例の一例を示す。

$[left]$: [[0.09, 0.46], [0.58, 1.45],
[0.97, 0.99], [0.50, 0.99], [0.07, 0.99]]
$[right]$: [[−0.42, −0.10], [0.50, 1.32],
[0.09, 0.58], [0.23, 0.99], [0.99, 0.99]]
$[go]$: [[−0.07, 0.24], [0.50, 0.89],
[0.64, 0.99], [0.99, 0.99], [0.65, 0.99]]

リストの第1番目の区分値は音源の方向を表すセンサーだが、方向センサーの値と行動とが一致するようにうまく区分されている。

ただし、この学習法では、誤った事例も削除されることは無いので、このように明確に表現されていない事例も残っている。削除すべき事例かどうかの判別は難しく、これを自動的に行なうことは今後の課題である。

[6] スタートしてから音源を見つけるまでを1トライアルとする。

6.4 属性の重み付け

ある試行において獲得した属性の重み付けの一例を示す。横軸に各行動における各センサー属性を

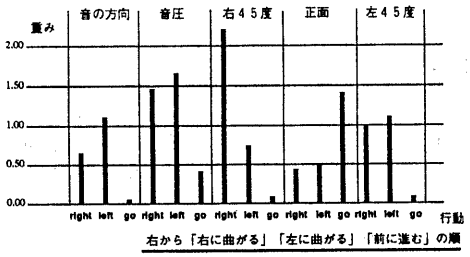

図 11: 属性の重み付け

示し、縦軸がその重みの値である。図 11 から、「前に進む」行動 [*go*] の属性の重みを見ると、「正面の距離センサー」の属性と「音圧センサー」の属性の重みが大きくなっており、他は小さい。これらのセンサーの値を基準に状況を分類する傾向を示している。これは望ましい傾向である。

しかし、音の方向センサーも重要であるはずなのに無視しており、必ずしも重み付けがうまくいっていないようである。他の行動に対する各属性の重みの意味も理解し難いものが多い。

図 10 に示したように属性の重み付けによる学習能力の向上は明らかである。したがって、重み付けがでたらめなものとは考えられない。事例ベースの学習の特長の 1 つは獲得した事例の透明性（可読性）である。その特長を生かすためにも、この事例の不透明さは改善していく必要がある。

6.5 一般化による事例数の削減の効果

区分値を用いずに事例をそのまま記憶し続けた場合の事例の数と本学習法における事例の数を比較した。各行動に対応して記憶された事例数を示す。こ

表 1: 獲得した事例の数（平均）

行動	[go]	[left]	[right]	合計
本学習法	30.25	12	14.25	56.5
区分化無し	84	18.5	18.5	121

の表から、事例の数が合計ではおおよそ 1/2 で済んでいることがわかる。区分による一般化が事例の増加を押える効果があることが確かめられた。事例の数が減れば類似度の計算が短縮でき、現実的な学習時間の短縮にも有効である。

7 まとめ

本論文では強化学習における認識の問題について検討した。エージェントが学習する対象は、「状況の見え方」と呼ぶ認識スキーマであると仮定し、この観点に基づく強化学習法を提案した。「状況の見え方」は入力属性空間の区分による一般化を行なう事例ベースの学習の導入によって実現した。強化学習における入力の一般化の問題を解くことで、本学習法の実用性を示した。

今後は、他の学習法との比較や、適切な応用課題を見つけることにより、本論文で提案した、行動に応じた適切な状況認識の獲得というアプローチの持つ利点や特徴などを明らかにしていきたいと考える。

参考文献

[1] Waltz,M.D.& Fu,K.S,A heuristic approach to reinforcement learning control systems, IEEE Transactions on Automatic Control, AC-10, 390-398,1965

[2] D.Chapman & L.P.Kaelbling,Input generalization in delayed reinforcement learning, Proc. IJCAI-1991,726-731,1991

[3] D.Whitehead & D.H.Ballard,Active Perception and reinforcment learning , Proc. of the Sixth International Workshop on Machine Learning,1989

[4] R.S.Sutton, Learning to predict by the methods of temporal differences,Machine Learning 3,9-44,1988

[5] 開 一夫, 佐藤 倫太, 安西 祐一郎, パーソナルロボットのための音声対話インターフェース, 情報処理学会 HI 研究会資料,93-HI-47,1993

[6] S.Salzberg, A Nearest Hyperrectangle Learning Method,Machine Learning,6,251-276,1991

[7] C.J.C.H.Watkins, & P.Dayan, Q-Learning, Machine Learning 8,279-292,1992

[8] R.S.Sutton,Integrated Architecture for Learning Planning, and Reacting Based on Approximating Dynamic Programing , Proc. of the Sixth International Workshop on Machine Learning,1989