特集論文

多様な戦略選択を可能にする事例ベースの政策表現とそのGAによる最適化

Exemplar-Based Policy with Selectable Strategies and its Optimization Using GA

池田 心[*1]
Kokolo Ikeda
京都大学
Kyoto University
kokolo@jaist.ac.jp, http://kokolo.info/www/

小林 重信
Shigenobu Kobayashi
東京工業大学
Tokyo Institute of Technology
Kobayashi@dis.titech.ac.jp, http://www.fe.dis.titech.ac.jp/

喜多 一
Hajime Kita
京都大学
Kyoto University
kita@media.kyoto-u.ac.jp, http://www.ipe.media.kyoto-u.ac.jp/

keywords: direct policy search, genetic algorithm, case-based reasoning, Markov decision process, exemplar

Summary

As an approach for dynamic control problems and decision making problems, usually formulated as Markov Decision Processes (MDPs), we focus direct policy search (DPS), where a policy is represented by a model with parameters, and the parameters are optimized so as to maximize the evaluation function by applying the parameterized policy to the problem.

In this paper, a novel framework for DPS, an exemplar-based policy optimization using genetic algorithm (EBP-GA) is presented and analyzed. In this approach, the policy is composed of a set of virtual exemplars and a case-based action selector, and the set of exemplars are selected and evolved by a genetic algorithm. Here, an exemplar is a real or virtual, free-styled and suggestive information such as “take the action A at the state S” or “the state S1 is better to attain than S2”.

One advantage of EBP-GA is the generalization and localization ability for policy expression, based on case-based reasoning methods. Another advantage is that both the introduction of prior knowledge and the extraction of knowledge after optimization are relatively straightforward. These advantages are confirmed through the proposal of two new policy expressions, experiments on two different problems and their analysis.

1. はじめに

定められた目標と現在の観測に基づき行動を決定する形式の意思決定問題はロボット制御や配送問題，市場投資問題など，広い工学的応用を持つ．このような問題に対してはさまざまな接近法が研究されており，例えばシステムの動作が既知あるいは単純で状態空間も小さい場合には，厳密解法である動的計画法や最適フィードバック制御などが利用可能である．一方実レベル問題への適用可能性を考慮すると，システムが未知のものであっても環境から与えられる報酬を手がかりとして試行錯誤を通して政策を学習する枠組みが必要であり，また広い状態空間を持つ問題に対してはその汎化を行うことで現実的な時間での満足化を図る必要がある．

代表的な学習手法の一つである TD 学習では，政策は状態または状態行動対の価値関数によって間接的に表現され，環境との相互作用を通して逐次更新される．その一つの実装である Q 学習 [Watkins 1992] は一定の条件下での最適収束性が保証されているが，現実的な時間での満足化のために状態表現に線型アーキテクチャなどの関数近似を用いる工夫もしばしば用いられる [Sutton 1998].

一方で本論文では，状態の汎化を行う関数モデルにより政策を直接的に表現し，これを一定期間環境に適用して評価値を計算し，進化計算などの手法によりパラメータの最適化を行う学習の枠組みである直接的政策探索（Direct Policy Search, 以下 DPS）[Moriarty 1999] に着目する．DPS は問題の特性や与えられた資源に応じて様々な政策モデルや最適化手法の利用が可能であるという特徴を持ち，また多段階の制御の最後に報酬が得られる問題などにおける報酬の遅れに頑健なこと，構成の単純さや多目

†1 2010 年より北陸先端科学技術大学院大学所属

的最適化の容易さとあいまって広い応用範囲を持つことが期待されている．

DPS における政策の関数モデルには，高次の状態空間を限られた変数で表現するための汎化能力と，状態の小さな差異によって細かく制御を切り替えるための局所化能力の両方が求められる．さらに，現実の問題への適用を考慮すると事前知識の導入や事後的な知識の抽出・検証が容易であることなども重要となる．また最適化アルゴリズムには，勾配情報や明示的な関数式がなくとも学習が可能であること，環境の持つランダム性に伴う評価値の揺らぎがあっても性能を大きく損なわないこと，大域的な探索を行い満足解を効率良く発見する能力を持つこと，などが求められる．

これらの要件を考慮に入れ，政策を状態と行動の対の集合で表現し，それを遺伝アルゴリズム（Genetic Algorithm, 以下 GA）を用いて最適化する手法として，SAP [Ikeda 2005]，SLIP[土谷 2006], FLIP[宮前 2009] 等が提案されている．ここでは行動の選択には事例ベース推論（Case-Based Reasoning, 以下 CBR）の一つである Nearest Neighbor 法が用いられ，現在の状態に最も近い状態と対になった行動が選択される．CBR は高次の汎化能力，非線型で局所的な推論に優れるとされ [Sheppard 1997]，GA の強力な探索能力とあいまって SAP 等は複数の制御問題に対して古典的な Q 学習などの手法を上回る良好な性能を実現している．

一方で，実応用を目指した場合，問題には多様な特性があり，また事前知識が利用できるとしてもその形式はさまざまであることに注意する必要がある．特に，行動の結果としての将来の状態がある程度予測可能な問題においては，それら同士を比較して最適な状態およびそこに至る行動を選択する戦略（先読み戦略，状態評価戦略などと呼ばれる）が有力となることが多いが [Knuth 1975]，現在の状態だけから行動を選択する SAP 等ではこのような推論を行うことはできない．

そこで本論文では，事例集合を用いて行動を選択するという枠組みを拡張・一般化し，**状態–行動型に限らない形式の事例集合と，形式に応じた CBR による行動選択を行う新しい枠組み**を提案する．このとき，事例はその形式によらずカプセル化することで，最適化を担う GA から見た場合に統一的に扱うことができる．これにより，問題の特性や与えられた資源に応じて様々な政策モデルや最適化手法を利用できるという DPS の特長を生かしたさまざまな応用を可能とすることを目指す．本論文では状態–価値型・状態–状態型という二つの新しい事例の形式と行動選択アルゴリズム，および状態–状態型への事前知識の導入法を提案し，特徴の異なる二つの問題に適用することでその有用性を示す．

本章に続き，第 2 章では対象とする問題クラスを示し，事例集合と CBR を用いた政策表現を提案する．第 3 章では事例集合の形式によらない GA の実装を示し，事前知識の導入法を提案する．第 4 章では実験により提案手法の有用性を示し，第 5 章では考察を行う．第 6 章はまとめである．

2. 事例集合による政策表現

2·1 対象とする問題クラス

マルコフ決定過程（Markov Decision Process, 以下 MDP）[van der Wal 1981] はエージェントの行動によって状態が確率的に遷移する環境のダイナミクスをモデル化したものであり，多くの動的制御・意思決定問題の定式化に用いられている．また，意思決定の時間間隔が任意であるような場合や，状態観測が不完全にならざるを得ない場合などには，それぞれ semi-MDP や部分観測 MDP などの拡張された MDP が用いられる．

本論文では，MDP に定式化できるシステムの中でも，複数の選択肢の中から行動を選択する状況で次状態が予測でき，評価が episodic な（始まりがあり，有限時間内に終わる）場合に着目し，この MDP のサブクラスを**予測可能 MDP** と呼んでこれを学習の対象とする．予測可能 MDP とその上での政策学習の目的を以下のように定義する．

- 環境のとりうる状態の集合を $\mathcal{S}$，状態 $s \in \mathcal{S}$ でエージェントのとりうる行動の集合を $\mathcal{A}_s$ で表す．
- 状態 $s \in \mathcal{S}$ で $|\mathcal{A}_s| \geq 2$ のとき，行動 $a \in \mathcal{A}_s$ を選択した場合の次状態および報酬が予測可能である，すなわち状態予測関数 $\mathcal{T}(s,a) : \mathcal{S} \times \mathcal{A}_s \to \mathcal{S}$ および報酬予測関数 $\mathcal{R}(s,a) : \mathcal{S} \times \mathcal{A}_s \to \mathbb{R}$ が定義・利用できる．
- エージェントは時刻に依存しない政策 π に従って行動を選択する．π は，現在の状態 s，状態予測関数 $\mathcal{T}$，報酬予測関数 $\mathcal{R}$ に基づき，行動 $a \in \mathcal{A}_s$ を決定する機構である．
- 一つの試行は episodic である．すなわち，時刻 $t = 0$ における初期状態の確率分布と，エピソード終了のための条件が定められており，有限時間内に終わる．
- 政策 π に従うエージェントがある試行で時刻 $t = 0$ から $t = t_{\mathrm{END}}$ まで行動し，各時刻に報酬 $r(t)$ を得たとすると，報酬の合計は $f(\pi) = \sum_{t=0}^{t_{\mathrm{END}}} r(t)$ で表せ，これを政策の評価値と呼ぶ．
- 学習の目的は，評価値の期待値 $E(f(\pi))$ を最大化するような政策を獲得することである．

予測可能 MDP の持つ制約は強いが，物理法則に基づく運動の制御や，囲碁・将棋のようにルールの定められた完全観測問題などではこれらが満たされており，多様な問題領域を含む．また，囲碁の例でも分かるように，次状態（この場合，自分の着手直後の状態）が予測可能であるからといって問題が容易であるとは限らない．一方で，episodic であることは，遷移ごとの逐次学習を行わずに，試行ごとの結果を評価値とする DPS には適した特徴である．これは強化学習では試行の最後だけに報酬が与えられるような場合に “報酬の遅れ” が課題になるのとは対照

的である．さらに重要なのは，予測可能 MDP においては，1 ステップ以上未来の状態を予測して比較・評価し，最も良い状態に到達できる行動を選ぶ**状態評価戦略**をとることが可能となることである．例えば，即時報酬が常に 0 であり，状態の好ましさを表す関数 $g(s):\mathcal{S}\to\mathbb{R}$ が存在するなら，最も好ましい次状態に遷移するような行動 $a^*=\arg\max_{a\in\mathcal{A}} g(s)$ を選択することが可能である．本研究で予測可能 MDP に着目する理由は，このように重要な問題領域を含みつつそこに適合した手法が考案できるためである．

2·2 事例集合による政策表現と一般化

SAP・SLIP 等では，政策を状態と行動の対の集合と，そこから行動を選択する Nearest Neighbor 法の組み合わせで表現している．本論文ではこれを拡張し，何らかの示唆的な情報，例えばある状態の好ましさなどを表す “事例” の集合と，そこからその形式に応じた CBR（広義には Instance-based 推論や Exemplar-based 推論，Memory-based 推論なども含む）の手法を用いて行動を選択するアルゴリズムを持つような政策を改めて**事例ベース政策**と定義する（図 1）．この枠組みで用いる事例とは，必ずしもエージェントが経験あるいは教師に教えられた現実の情報だけを指さずに，DPS の機構により作成された仮想的なものを含むものとする．事例の訳語には模範や典型という意味で **exemplar** を用い，事例ベース政策を Exemplar-Based Policy（EBP）と呼ぶことにする．

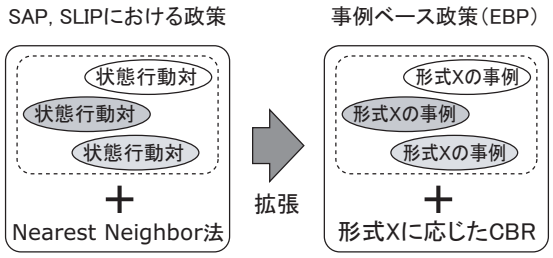

図 1 EBP のイメージ図

EBP の枠組みでは，多様な問題の特性に合わせ，また事前知識の導入を容易にするために，「状態 s では行動 a を取るのが良い」という形式だけではなく，「状態 x よりも状態 y のほうが良い」など様々な形式を利用可能とする．以下に，この枠組みにおける代表的な事例の形式とそれに対応した行動選択アルゴリズムの概略を例示する．ここで示す推論戦略のいくつかはすでに EBP という枠組みとは異なる視点では存在しているものであるが，事例の集合による政策表現という視点から捉え直すと，EBP の実現例と見ることができる．

§1 状態–行動型 EBP

状態–行動型 EBP は，SAP 等で用いられている事例・推論の形式であり，一つの事例は「状態 $s\in\mathcal{S}$ では行動 $a\in\mathcal{A}$ を取るのが良い」ことを直接表す．事例の集合 $\mathcal{E}=\{(s_j,a_j)\}_j$ が与えられ，これを「s_j のクラスは a_j である」と解釈すれば，現在の状態 s^{Now} のクラスを推論する CBR によって，取るべき行動を定めることができる．例えば，状態間に距離尺度が導入できれば，s^{Now} に対して最も近い状態 s_{j*} を持つ事例を探して，対応する行動 a_{j*} をとる Nearest Neighbor 型の行動選択アルゴリズムが利用できる [Sheppard 1997]（図 2）．

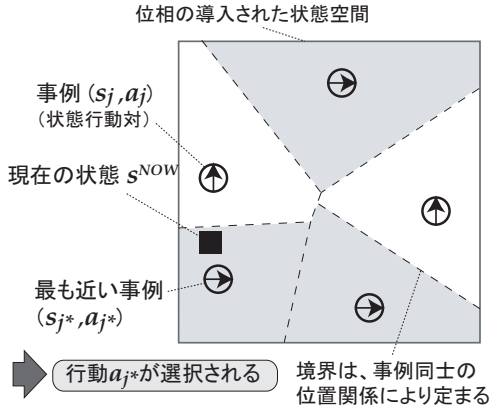

図 2 状態–行動型 EBP での Nearest Neighbor 型の行動選択

§2 状態–価値型 EBP

予測可能 MDP では行動の選択に状態評価戦略がしばしば用いられる．状態の好ましさを表す関数 $g:\mathcal{S}\to\mathbb{R}$ は多くの場合何らかの形でパラメータ化され，人手による調整のほか，進化計算などによる自動的な調整も提案されている [Graham 2001]．本論文では，事例集合をパラメータとしてこの関数を表現する**状態–価値型 EBP** を提案する．

状態–価値型 EBP では，一つの事例は「状態 $s\in\mathcal{S}$ の好ましさは $r\in\mathbb{R}$ である」ことを表す．状態空間 $\mathcal{S}$ に適当な位相を導入できる場合，何らかの関数近似手法を用いることにより，任意の状態 $s\in\mathcal{S}$ に対してその好ましさ $g(s)$ を事例の集合 $\mathcal{E}=\{(s_j,r_j)\}_j$ から推定することができ，状態評価戦略を実行することが可能となる（図 3）．関数近似手法の実装例は 4·2·2 節に示す．

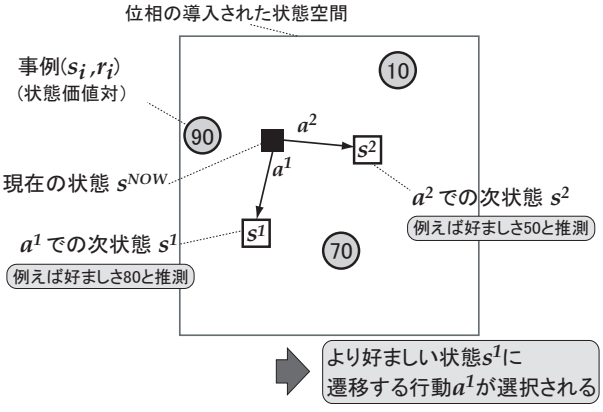

図 3 状態–価値型 EBP での行動選択

状態–価値型 EBP では，現在の状態に関する情報を用いて直接行動を選択するわけではなく，選択肢として与えられる行動の帰結としての未来の状態の好ましさを比較することで行動を選択する．そのため，状態空間が大きい場合にはその縮約したものとして特徴量空間 $\mathcal{W}$ と特徴量化関数 $\mathcal{S} \to \mathcal{W}$ をうまく設計することにより，事例を「特徴量ベクトル $x \in \mathcal{W}$ の好ましさは $r \in \mathbb{R}$ である」のように表すことで汎化能力を高めることが期待できる．

§3 状態–状態型 EBP

本論文ではさらに，一つの事例が「状態 $x \in \mathcal{S}$ は状態 $y \in \mathcal{S}$ よりも好ましい」ことを表す**状態–状態型 EBP** を提案する．好ましさの大小を推定することは空間 $\mathcal{S} \times \mathcal{S}$ から 2 値へのクラス分類に他ならないから，何らかのクラス分類手法を用いて任意の状態対に対してその好ましさの大小を推定することができる（実装例は 4·2·3 節に示す）．このとき，次状態の集合が予測可能で有限であるなら，その集合の中から最も好ましい状態をトーナメント選択等によって定め，そこに遷移するための行動を選択する戦略が実行できる．

状態–状態型 EBP と状態–価値型 EBP はどちらも状態評価戦略に基づくが，状態–状態型 EBP は相対的な序列のみを持つ点で，絶対的な好ましさという値を持つ状態–価値型 EBP と異なる．この特徴により，3·3·1 節で述べるように，状態–状態型 EBP では熟練者の行動履歴といった事前知識を導入することが容易になっている．

3. GA を用いた事例集合の最適化

3·1 EBP-GA と は

SAP や SLIP では，状態–行動型の事例集合をランダムに初期化し，GA を用いて精選，最適化している．本論文ではこれを一般の EBP に拡張し，**EBP-GA** と呼ぶ．EBP-GA では，行動選択アルゴリズムは固定して事例集合を最適化の対象とする．一つの個体は一つの事例集合を表し，複数の事例集合間で事例の交換等の操作を行い，MDP にそれぞれの政策を適用することで得た評価値を用いて淘汰選択を行う．

EBP-GA では，個体が保持する事例はエージェントが経験したり教師によって与えられた既定のものに限らず，ランダムに生成したり交叉操作により生成する等，それ自身が探索対象である．事前知識のない問題ではすべての個体のすべての事例はランダムに生成され，有益なもの（正例）と有害なもの（負例）を含むそれらの中から，交叉や淘汰といった進化の過程を経ることで有益なものだけが精選されていくことを期待する（図 4）．

GA は複数の解の生成保持，交叉，突然変異，世代交代といった操作からなる．もしその各操作が**事例の表現形式によらずに**定義できれば，どのような形式の EBP にも同じ枠組みを適用することができる．

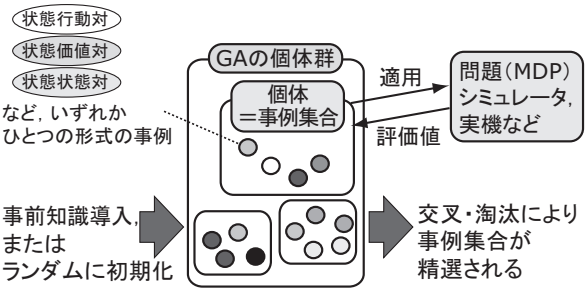

図 4 EBP-GA の概念図

3·2 EBP-GA の基準的な実装

EBP-GA を利用するためには，事例の形式や行動選択アルゴリズムいった政策モデルと，交叉や世代交代といった GA の操作を定める必要がある．これらの多くには CBR や GA の各領域の既存の技術を用いることができ，解くべき問題の特性や事前知識の形式，与えられた計算資源・評価可能回数や求められる精度によって適切な組み合わせを選ぶことが可能である．

本論文では，EBP のための GA の実装の一つとして，政策モデルにかかわらず利用することのできる Universal EBP-GA（以下 UEBPGA）を提案する．その構成は以下の通りである．

- 事前知識が与えられない場合，GA の個体の持つ事例集合はすべてランダムに初期化する．
- 世代交代モデルには，単純な家族選択モデルである MGG[Satoh 1996] を用いる．
- 交叉操作では，両親の持つ特徴を遺伝させるためにその事例集合を混合する操作を用う．
- 突然変異操作では，保持する事例を一定の確率で再初期化する．

表 1 UEBPGA における変数表記

変数	意味
N_{pop}	個体群に含まれる政策（個体）の数
N_{exem}	一つの政策が保持する事例の数
N_{child}	一回の交叉操作で作る，子個体の数
N_{gener}	探索を打ち切る世代交代回数
R_{mute}	突然変異操作で事例を再初期化する確率
$\mathcal{E}^i$	i 番目の個体（事例集合）
e^i_j	i 番目の個体の，j 番目の事例
$\mathcal{X}$	事例の空間．状態–行動型なら $\mathcal{X} = \mathcal{S} \times \mathcal{A}$

UEBPGA の具体的な手順は以下の通りである．パラメータと記号の意味を表 1 にまとめる．また，全体の処理の流れを図 5 に示す．

(1) 集団サイズ N_{pop} 等のパラメータを決定し，N_{pop} の事例集合からなる個体群を初期化する．一つの個体 $\mathcal{E}^i$ は N_{exem} の事例 $e^i_j \in \mathcal{X}, (j = 1, ..., N_{\mathrm{exem}})$ からなり，その事例はランダムに生成する．

(2) ランダムに 2 つの親個体 $\mathcal{E}^{p_1}, \mathcal{E}^{p_2}$ を選択する．

(3) N_{child} 回の交叉操作と突然変異操作によって子個体を生成する．これら，親 $\mathcal{E}^{p_1}, \mathcal{E}^{p_2}$ と子の $2 + N_{\mathrm{child}}$

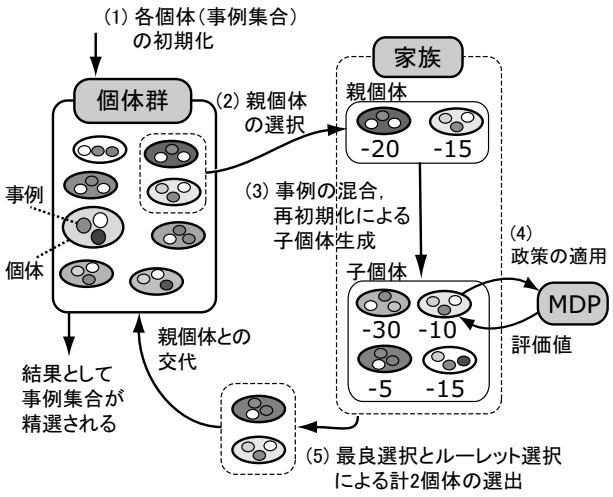

図 5 UEBPGA の基本的な流れ．EBP の形式とは独立に事例の精選を行う．

個体を家族と呼ぶ．

(4) 家族の各個体について，MDP に適用することで評価値を得る．なおもし親個体が評価済みで，かつ評価値に不確実性（ゆらぎ，ノイズ）のない問題であれば，親の再評価は不要である．

(5) 家族の中で最も評価値の優れた個体と，その他の個体の中から順位に従いルーレット選択 [Satoh 1996] により選んだ個体，計 2 個体を，親個体 $\mathcal{E}^{p_1}, \mathcal{E}^{p_2}$ の代わりに個体群に戻す．

(6) 手順 (2) から手順 (5) までを 1 世代とし，これを N_{gener} 世代繰り返す．

一つの子個体 $\mathcal{E}^c$ を生成するための交叉操作（混合）と突然変異操作（再初期化）は以下の手順で行う．事例は集合として扱うため,「遺伝子座」のような位置の概念は持たない．そのため，基本的なアルゴリズムとして 2 つの集合から要素をランダムに選ぶ手法をここでは用いる．

(1) 親個体 $\mathcal{E}^{p_1}, \mathcal{E}^{p_2}$ が与えられる．

(2) 子個体 $\mathcal{E}^c$ を空集合で初期化する．

(3) 事例 $e \in \mathcal{E}^{p_1} \cup \mathcal{E}^{p_2}$ をランダムに選択する．

(4) もし e が $\mathcal{E}^c$ に含まれていなければ $\mathcal{E}^c$ に加える．

(5) 手順 (3),(4) を $|\mathcal{E}^c| = N_{\mathrm{exem}}$ となるまで繰り返す．

(6) それぞれの事例 $e^c_j \in \mathcal{E}^c$ に対し，確率 R_{mute} で再初期化を行う．すなわち，$e \in \mathcal{X}$ をランダムに選択し e^c_j とする．

3·3 EBP-GA の特長と意義

1 章で述べた DPS に求められる要件を考慮しながら，EBP-GA の枠組み，ないしその実装の一つである UEBP GA に期待できる特長を以下にまとめる．

- **情報の表現形式が選択可能**　EBP-GA では多様な問題の特性に合わせて多くの政策の表現形式や行動決定方法が利用可能である．また，多様な形式の情報を事例という概念にカプセル化することで，それを精選する GA の実装とは独立して表現形式や行動決定方法を選択することが可能である．
- **CBR による政策の表現能力**　CBR の枠組みは，教師データを参考に未知の入力に対して適切な出力をするクラス分類問題に頻繁に用いられ，悪スケール性やいわゆる次元の呪いなどを考慮した多くの優れた手法が提案されている [Aha 1997]．CBR による行動選択アルゴリズムを適切に用いれば，事例集合が疎な領域ではおおまかな制御を，密な領域では細かい制御を行うことが期待できる．このような事例集合の粗密は，もしそれが政策の評価値を高くするのに必要であれば，事例の生成・交叉・選択の段階で自然に生じるはずである．
- **GA による政策最適化に適した特性**　政策最適化問題ではしばしば，勾配情報を必要とする手法が使えないこと，政策の評価が高コストなため効率の良い大域的探索が求められること，また環境などのランダム性に伴い評価値に揺らぎが発生することなどが課題となる．GA は勾配情報を必要とせず，複数の解を保持することで評価値の揺らぎにも比較的頑健であり，交叉操作によって情報交換を行うことで効率的な探索を行うことで知られ，政策最適化との親和性は高いと考えられる．
- **領域知識の導入が容易**　EBP-GA は事例集合そのものを最適化対象として生成していくため，事前知識が与えられない場合でも動作する．しかし，次節で示すように，多くの場合 EBP-GA には事前知識を初期個体として導入することが容易であり，これは最適化の速度・精度を向上させるために有効であると考えられる．

§1 事前知識の導入

一般的なエキスパートシステムでは知識は if-then のルールで記述する必要があるが，これが知識の保有者にとって必ずしも自然な知識の記述であるとは限らない．EBP-GA では多くの形式で事例を記述することができるため，知識の保有者が自分にとって自然な形で記述した知識をただちに事例集合として利用する可能性が高まる．

例えば,「状態 s では行動 a を取るのが良い」という形式の知識は，熟練者の行動記録や，囲碁・将棋などの場合は棋譜という履歴として，比較的入手しやすいものといえる．この状態–行動型の知識は，そのまま状態–行動型 EBP の初期個体の事例の一部として導入することができる．さらに予測可能 MDP を想定すると,「熟練者が状態 s で行動 a^1 をとった」という知識は,「(他の行動 a^2 や a^3 によって遷移する）状態 s^2 や s^3 よりも,(行動 a^1 によって遷移する）状態 s^1 のほうが良い」という解釈に立てば，状態–状態型の事例としても導入することができる（図 6）．

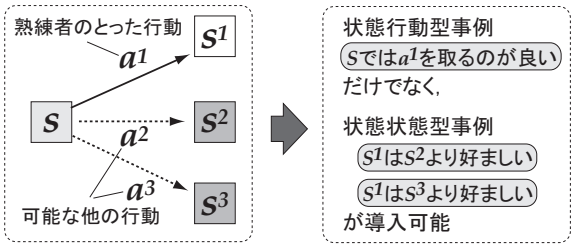

図 6 状態–行動対の，状態–状態型事例としての解釈

4. 実　　験

3·3 節で述べた EBP-GA の特長のうち，状態–行動型の EBP の表現能力や GA による効率的な探索については，単純な Q 学習との比較 [Ikeda 2005] および [土谷 2006] などでも実証されている．本章ではさらに，以下の点について確認する．

- UEBPGA が他の合理的な DPS と比べても十分な性能を示すこと．TD 学習などの逐次学習を行う手法と episodic な環境に適した UEBPGA を公平に比較することは困難であるため，3 層パーセプトロンと実数値 GA を用いた DPS を比較に用いる．
- 複数の EBP の形式それぞれに得手不得手があり，問題の特性に合わせた形式選択が有用であること．そのために本論文では特徴の異なる二つの問題を扱う．
- 事前知識の導入により，最適化序盤の性能が向上すること

状態–行動型の事例ベース政策については，SLIP[土谷 2006] や FLIP[宮前 2009] など，より高速に精度の高い事例集合を得るための高度な最適化実装が提案されている．本論文の主眼は事例ベース政策に状態–価値型，状態–状態型などの新しい形式を持ち込むことで多様な問題に取り組むことを可能にすることであり，「状態–行動型の戦略が適する問題でも UEBPGA の性能が SLIP や FLIP に勝る」ことを主張するものではない．この点については 5·2 節で詳述する．

4·1 対 象 問 題

実験は，N_{dim} 次元の連続な状態空間 $[0,1]^{N_{\mathrm{dim}}}$ と離散の行動空間を持つ MDP として，それぞれ ACROBOT [Spong 1994] および TETRIS を対象とする．どちらの問題も PC 上のシミュレータを用いて評価した．以下に，各々の問題の概要を説明する．

§1 ACROBOT 問題

ACROBOT は，直列に接続された 2 本のリンクを下垂状態から直立状態に振り上げる鉄棒運動を模した問題であり，トルクはリンクの接続部分にのみ加えることができる（図 7）．系の状態 $(x_1,x_2,x_3,x_4)\in\mathcal{S}=[0,1]^4$ はリンクの角度と角速度に対応し，x_1 は内側のリンクの角度 $\theta_1\in[0,2\pi]$（単位 rad），x_2 は外側のリンクの相対角度 $\theta_2\in[0,2\pi]$（単位 rad），x_3,x_4 はそれぞれの角速度 $\dot{\theta}_1\in[-4\pi,4\pi],\dot{\theta}_2\in[-9\pi,9\pi]$（単位 rad/s）を正規化したものである．行動 $a\in\mathcal{A}=\{-2,0,2\}$ は加えるトルク（単位 m・N）を表す．各リンクの長さは 1m，質量は 1kg とする．

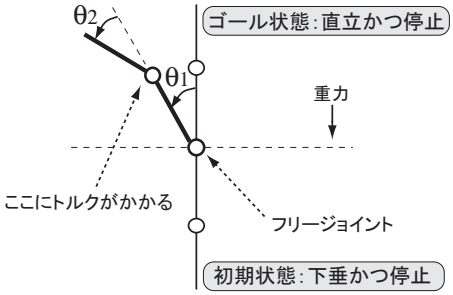

図 7 ACROBOT 問題

状態遷移ルールは http://www.cmap.polytechnique.fr/~munos/variable/acrobot.html で提供されているものを使用した．MDP としての意思決定と状態遷移（1 ステップ）は 0.05 秒 ごとに行い，物理シミュレーションは 0.01 秒 刻みで行う．

各エピソードは下垂停止状態 $s=(0.5,0.0,0.5,0.5)$ から開始し，直立停止状態 $s=(0.0,0.0,0.5,0.5)$ との誤差が各 0.01 以内（x_1,x_2 の値は 0.01 以下または 0.99 以上，x_3,x_4 の値は 0.49 以上 0.51 以下）となったときゴールとして終了する．300 ステップ（15 秒）経過した場合も終了する．各時刻にリンクの高さに応じて報酬 $\cos(\theta_1)+\cos(\theta_1+\theta_2)-3$ が与えられる．

§2 TETRIS 問題

TETRIS は Aleksey Pazhitnov により提案されたビデオゲームであり，ベンチマーク問題としてコンピュータプレイヤーを作ることが強化学習などの分野で研究されている [Szita 2006]．その接近法はさまざまであり，例えばカメラとキーボードを打鍵する指を備えた実ロボットを作成した例 [松井 2001] がある一方で，実験を PC 内で行えるよう MDP として定式化した例も多く，さまざまな特徴量とニューラルネットワークを用いるなどの学習戦略が用いられている [野村 2001]．

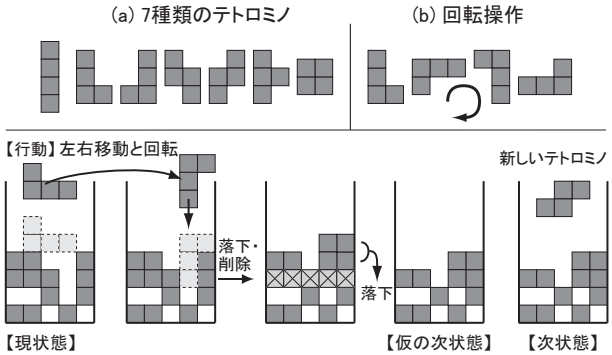

図 8 TETRIS 問題：テトロミノの種類，回転，状態遷移ルール

TETRIS ではプレイヤはテトロミノと呼ばれるブロッ

クの塊（図 8(a)）を操作し，$C_{\mathrm{width}} \times C_{\mathrm{height}}$ の枠内に落としていく．7 種あるテトロミノのうち一つがランダムに提示され，プレイヤはそれを左右に移動，回転（図 8(b)）して落下地点を定める．落下後，もし横方向一列が全てブロックで埋まるとその列は削除され，その列より上方のブロックは一列分落下する（図 8 下段）．TETRIS の目的は，枠内にブロックを落下できる場所がなくなる前に，C_{norm} 列を削除することである．本論文では $(C_{\mathrm{width}}, C_{\mathrm{height}}, C_{\mathrm{norm}}) = (6, 12, 10)$ とした．

i. 予測可能 MDP としての定式化

TETRIS を，予測可能 MDP の条件を満たすように定式化する．行動選択後，新しいテトロミノが与えられる直前までの遷移は一意であり，これを仮の次状態とする．仮の次状態からの遷移として新しいテトロミノはランダムに提示されるが，ここにはプレイヤの行動が関与しないため（$|\mathcal{A}_s| = 1$），予測可能 MDP の条件を違反しない．

TETRIS における状態は，枠内のブロックの有無を表す行列 M_{xy}（あれば 1，なければ 0）と提示されたテトロミノの種類番号によって，一意に定まる．この状態空間は極めて大きく（約 $2^{6\times12}$）効率的に扱うことが困難なため，以下では次のように特徴量を定め，これを政策に与えることにする．

(1) 特徴量 f_x はある縦一列のブロックの高さ，すなわち $f_x = \max(y \,|\, M_{xy} = 1)$ と定義する．ブロックがない場合は $f_x = 0$ とする．
(2) テトロミノが提示されている場合これに加え特徴量 f_0 としてテトロミノの種類番号を用いる．
(3) 合計で C_{width} または $C_{\mathrm{width}}+1$ の特徴量が与えられ，これらはそれぞれ $[0, 1]$ の範囲に正規化される．

TETRIS の操作は，左右への移動と回転である．これらはそれぞれ最大で C_{width}，4 種類あるため，行動空間 $\mathcal{A}$ はこれらの積とし，重複を含んで 24 の離散の行動を持つ．報酬はそのステップで削除した列の数 $\times 1.0$ とする．合計で 10 列以上を削除すると MDP は終了し，合計報酬は 10.0 となる．落下したブロックが枠内に収まらなかった場合も MDP は終了する．評価値の揺らぎを軽減するため，実験では一つの政策に対して 10 回のゲームを行い，その合計報酬の平均値を評価値として用いる．

ii. ヒューリスティックプレイヤ

3·3·1 節では EBP-GA に事前知識として熟練者の行動履歴を導入する方法を提示した．その有効性を確認するため，本節では比較的高性能なヒューリスティックプレイヤを導入する．このプレイヤは，全ての次状態に対する特徴量 $(f_1, ..., f_{C_{\mathrm{width}}})$ が与えられると，高さの二乗和 $g_h = \sum_{x=1}^{C_{\mathrm{width}}} f_x^2$ および表面の荒さを意味する $g_r = \sum_{x=2}^{C_{\mathrm{width}}} (f_x + f_{x-1}) |f_x - f_{x-1}|$ を計算し，$g = g_h + g_r$ が最も小さくなるような行動を選択する．これらの関数は TETRIS の特徴を十分考慮して設計されたものであり，平均で 9.8 の報酬を得ることができる．一方で，10 回に 1 回の割合でランダムな行動を取らせると平均報酬は 8.6 まで低下することから，TETRIS で高い報酬を得るためには連続して良い行動を選択しなければならないことがわかる．

4·2 政策最適化手法の実装

§1 状態–行動型 EBP の行動選択アルゴリズム

状態–行動型（SA 型と略す）EBP の実装の一つとして，実験では以下に示す k-Nearest Neighbor 法を用いて行動を選択する．$k_{\mathrm{NN_{SA}}}$ は局所性を制御するパラメータであり，小さいほど局所性が高い．行動選択のための計算量は $O(N_{\mathrm{dim}} N_{\mathrm{exem}} \log N_{\mathrm{exem}})$ 以下である．

(1) 事例集合 $\mathcal{E} = \{(s_j, a_j)\}_j$, $(s_j, a_j) \in \mathcal{S} \times \mathcal{A}$ が与えられている．
(2) TETRIS では，提示テトロミノが同じ（f_0 が等しい）事例集合のみを参照する．
(3) 現在の状態 s に対して，各事例の s_j との Euclid 距離を計算し，小さい順に入れ替える．
(4) 上位 $k_{\mathrm{NN_{SA}}}$ 個の事例が持つ行動の中で最も数が多いものを選択する．同数の場合はその中で最も上位の事例の行動を選択する．例えば順に行動 1, 行動 2, 行動 3, 行動 3, 行動 2 であれば，行動 2 を選択する．

§2 状態–価値型 EBP の行動選択アルゴリズム

状態–価値型（SR 型と略す）EBP の実装の一つとして，本論文では以下の手続きを提案し，実験に用いる．α_{local} は局所性を制御するパラメータであり，大きいほど局所性が高い．行動選択のための計算量は $O(|\mathcal{A}| N_{\mathrm{dim}} N_{\mathrm{exem}})$ 以下である．

(1) 事例集合 $\mathcal{E} = \{(e_j)\}_j = \{(s_j, r_j)\}_j$, $(s_j, r_j) \in \mathbb{R}^n \times \mathbb{R}$ が与えられている．
(2) 次状態の候補の集合 $\{s^{a_i}\}_i = \{\mathcal{T}(s, a_i) \,|\, a_i \in \mathcal{A}_s\}$ を計算する．
(3) 各 s^{a_i} について，事例との距離に基づいて重み付けした値 $g(s^{a_i})$ を計算する．

$$g(s^{a_i}) = \frac{\sum_{e_j \in E} r_j d(s_j, s^{a_i})}{\sum_{e_j \in E} d(s_j, s^{a_i})},$$

$$d(s_j, s^{a_i}) = \frac{1}{1 + \alpha_{\mathrm{local}} ||\overrightarrow{s_j - s^{a_i}}||}$$

(4) $g(s^{a_i})$ が最大となる行動を選択する．複数の場合はランダムに選ぶ．

§3 状態–状態型 EBP の行動選択アルゴリズム

状態–状態型（SS 型と略す）EBP の実装の一つとして，まず二つの状態 $x \in \mathbb{R}^n$ と $y \in \mathbb{R}^n$ が与えられた場合に優劣を定める以下の手続きを提案する（図 9）．$k_{\mathrm{NN_{SS}}}$ は局所性を制御するパラメータであり，小さいほど局所性が高い．

(1) 事例集合 $\mathcal{E} = \{(x_j, y_j)\}_j$, $(x_j, y_j) \in \mathbb{R}^n \times \mathbb{R}^n$ が与えられており，一つの事例は状態 x_j が状態 y_j より優れていることを意味する．

(2) 各事例に対し，比較したい二つの状態との重心同士の距離 $dist_j = |\frac{\overrightarrow{x_j + y_j}}{2} - \frac{\overrightarrow{x + y}}{2}|$ を計算する．
(3) E の中で $dist_j$ の小さいもの上位 $k_{\mathrm{NN_{SS}}}$ 個を新たに E_{local} とする．
(4) E_{local} の各事例に対し，$\overrightarrow{x_j - y_j}$ と $\overrightarrow{x - y}$ の内積を計算する．
(5) 内積が正となるものが $k_{\mathrm{NN_{SS}}}$ の半数以上である場合 x が優れているとし，半数未満の場合 y が優れているとする．

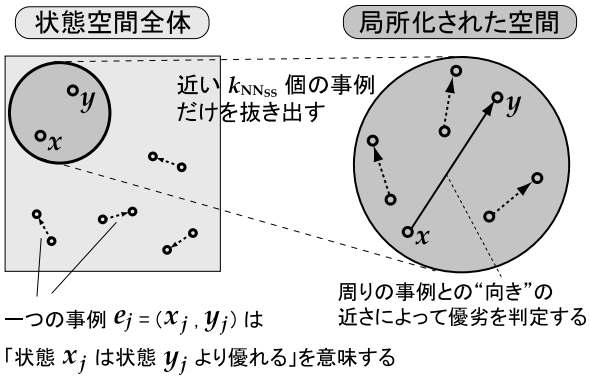

図 9 状態–状態型 EBP のための，状態の優劣判断アルゴリズム

この方法は優劣に関する遷移律を保証せず，全ての次状態から最も良い状態を選択するための比較法によってその結果が異なることがある．本論文では，ACROBOTでは行動集合 $\mathcal{A} = \{-2, 0, 2\}$ に対して行動 $a_1 = -2$（m･N）と行動 $a_2 = 0$（m･N）を最初に比較するようトーナメントを固定する．TETRISでは24通りの行動について比較数に2以上の偏りのないトーナメントを毎回ランダムに生成する．行動選択のための計算量は $O(|\mathcal{A}| N_{\mathrm{dim}} N_{\mathrm{exem}} \log N_{\mathrm{exem}})$ 以下である．

§4 3層パーセプトロンと実数値 GA による DPS

3層パーセプトロンは脳の神経細胞による情報処理の方法を模擬した入力－出力モデルであり，主に教師ありのパターン認識問題にバックプロパゲーション学習とともに用いられる．しかしその優れた汎化性能を教師なしの学習にも活用するために，結合重みを実数値遺伝子としてDPSを行う試みもなされており，二重倒立振子問題などへの適用も行われている [井口 2001]．本論文でもこの組合せ（以降NNGAと表記する）をとる．

本論文では，入力層 N_{dim}，中間層 N_{mid}，出力層 $|\mathcal{A}|$ の3層パーセプトロンを用いる（図10）．状態 $(x_1, ..., x_{N_{\mathrm{dim}}}) \in [0,1]^{N_{\mathrm{dim}}}$ が与えられると入力層は x_i を出力し，中間層はシグモイド関数またはガウス関数を用いた出力を行い，出力層は入力をそのまま出力し，その出力が最も大きい出力層素子に対応する行動が選択される．バイアスユニットを含めると最適化対象となる素子の結合重みの数は $(N_{\mathrm{dim}} + 1)(N_{\mathrm{mid}}) + (N_{\mathrm{mid}} + 1)|\mathcal{A}|$ である．その最適化には標準的な実数値GAの枠組みを用い，交叉には UNDX-m[喜多 2000]，突然変異は用いず，世代交代には UEBPGA と同じく MGG を用いる．各重みは $[-1, 1]$ の範囲でランダムに初期化し，交叉の結果 $[-1000, 1000]$ の範囲を外れる場合は動作の安定性を考慮して範囲内に収めた．シグモイド関数素子の数 N_{midS} とガウス関数素子 N_{midG} の数はパラメータである．

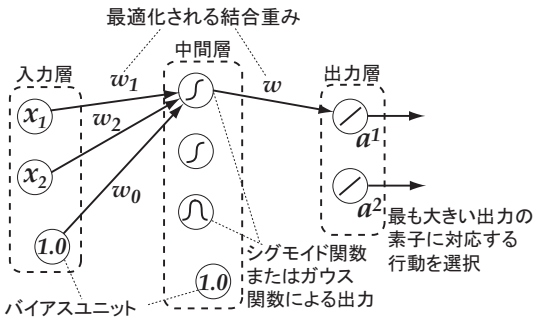

図 10 3層パーセプトロンによる行動選択

4·3 性能評価実験

本節では，SA型・SR型・SS型の UEBPGA，および NNGA について，その性能の特徴を調べる実験を行う．ACROBOT は評価値に揺らぎの入らない問題のため，集団内の最良個体の評価値と，ゴール状態に到達したかどうかを比較する．TETRIS は最良個体の選択が困難なため，集団内の平均評価値を比較する．すべての実験は30試行，$N_{\mathrm{gener}} = 10000$ 世代まで行い，その平均と必要に応じて標準偏差も用いて比較する．

表 2 各問題に用いた各手法の最良パラメータセット

手法	パラメータ	ACROBOT	TETRIS
SA 型 UEBPGA	R_{mute}	0.1	0.0
	$k_{\mathrm{NN_{SA}}}$	1	1
SR 型 UEBPGA	R_{mute}	0.2	0.0
	α_{local}	1000	10
SS 型 UEBPGA	R_{mute}	0.1	0.0
	$k_{\mathrm{NN_{SS}}}$	7	75
NNGA	N_{midS}	0	0
	N_{midG}	10	10

用いたパラメータセットを表2にまとめる．GAの集団サイズ N_{pop} は100，生成子個体数 N_{child} は10，事例数 N_{exem} は300に固定した．UEBPGAの事例再初期化確率 R_{mute} は $\{0.0, 0.01, 0.03, 0.1, 0.2, 0.3\}$ から，NNGAの中間層はシグモイド関数素子 N_{midS} とガウス関数素子 N_{midG} を5刻みで各0から20までかつ合計20まで，SA型の局所化パラメータ $k_{\mathrm{NN_{SA}}}$ は $\{1, 3, 5, 10\}$ から，SR型の局所化パラメータ α_{local} は $\{1, 3, 10, 30, 100, 300, 1000, 3000, 10000\}$ から，SS型の局所化パラメータ $k_{\mathrm{NN_{SS}}}$ は $\{3, 7, 13, 25, 49, 75\}$ から最も良いものを予備実験により選択した．

図11はACROBOTにおける学習曲線である．下図よ

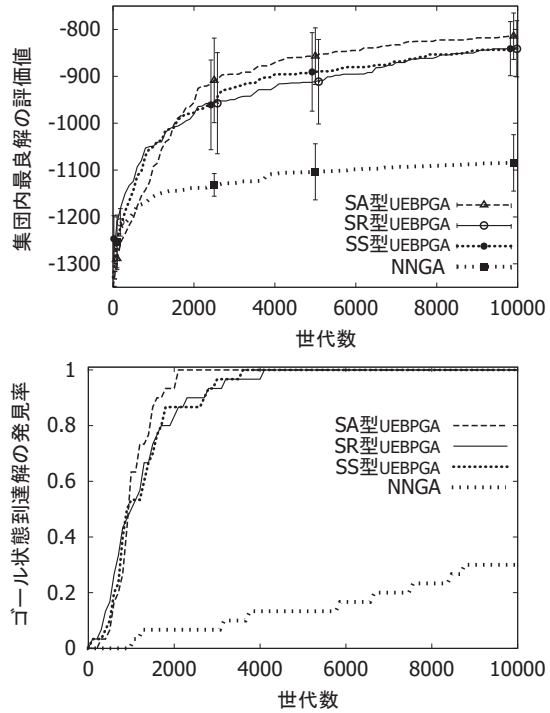

図 11 ACROBOT における学習曲線．最良解の評価値，30 試行の平均・分散（上），ゴール状態に到達する解の発見率（下）

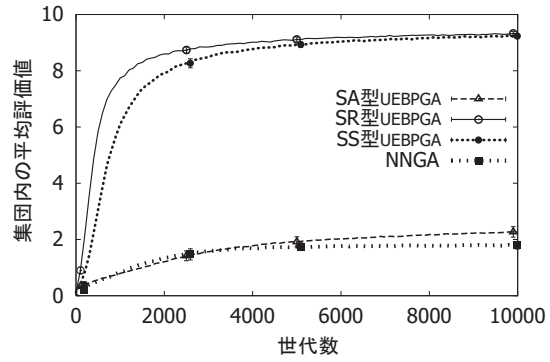

図 12 TETRIS における 4 手法の学習曲線．集団内の平均評価値，30 試行の平均・分散

り，UEBPGA は 3 つの推論形式で全く実装が異なるにもかかわらず，全ての試行でゴール状態に到達する解を発見している．その中では SA 型が最も発見が早く，また上図からそれらの解の中でもより効率の良い解を発見できていることがわかる．NNGA は 10000 世代の中ではゴール状態に到達する解を発見できない場合が多い．中間層のガウス関数素子の数を $N_{\mathrm{midG}} = 20$ とし，30000 世代の最適化を行うと半数以上の試行で発見に成功することは確認しているが，UEBPGA に比べればその性能は劣る．

図 12 は TETRIS をランダムな初期解から最適化した場合の学習曲線である．ACROBOT の場合と比べて相対的な分散が小さくなっているのは，集団内の平均評価値をとっているため試行ごとのばらつきが小さくなるからである．性能は SR 型と SS 型の UEBPGA が優れており，ヒューリスティックプレイヤが 20 回に 1 回ランダムな行動を選択した場合（平均評価値 9.4）とほぼ同じ性能を達成している．SA 型の UEBPGA と NNGA の性能は低く（同 2.0 前後），追加実験として SA 型 UEBPGA で事例数・探索世代数を 10 倍（$N_{\mathrm{exem}} = 3000$，$N_{\mathrm{gener}} = 10^6$）にしても平均評価値は 4.5 にしか上昇しなかった．これは本質的に SA 型の政策モデルがこの問題に親和性が低いことを意味しており，例え SLIP や FLIP のような高度な最適化を行っても SR 型や SS 型には及ばないと予想される．この点については 5·1 節で考察する．

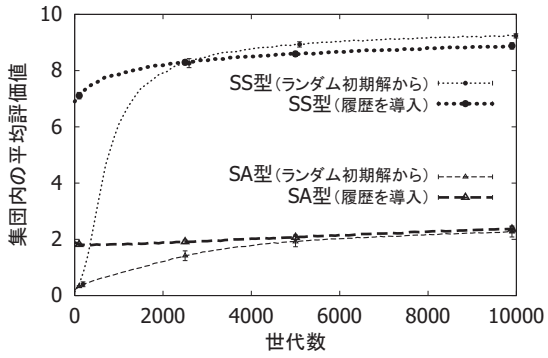

図 13 TETRIS における事例導入の効果．集団の平均評価値，30 試行の平均・分散

最後に図 13 は SA 型，SS 型の UEBPGA にヒューリスティックプレイヤが作成した履歴を初期個体の事例集合として導入した場合の性能を比較したものである．なお，試行ごとに異なる履歴を導入している．導入の結果，世代 0 から比較的良好な性能を持つこと，またそこからも性能の向上が見られより良い組み合わせの事例集合が探索されていることがわかる．一方で探索終盤にはランダムな初期解から最適化した場合とほぼ同じ性能になり，SS 型では性能の逆転も生じている．これは，比較的似た傾向を持つ事例を組み替えるよりも，多様な事例から適切な組み合わせを選んだ方が良い場合があるということであり，例えば UEBPGA で用いた単純な突然変異とは異なる操作による性能改善の余地があることを示している．

4·4 実験結果の分析

2 つの問題の MDP としての特徴，4 つの手法の結果，UEBPGA におけるパラメータと性能の関係を表 3 にまとめる．

まず，本論文で提案した SR 型・SS 型の UEBPGA はどちらの問題でも NNGA を上回る性能を示し，また問題のゴール状態に十分高い確率で到達することができた．また，TETRIS においては，SS 型は SR 型に比べわずかに性能が劣るものの，履歴を利用することで探索序盤の性能を大幅に向上させることに成功し，評価コストが高くまた熟練者の履歴が入手可能な問題での活用が期待できる．一方 SA 型の UEBPGA は ACROBOT では最も良

表 3 問題の特徴，用いた手法の性能，適切なパラメータの傾向

項目	ACROBOT	TETRIS
問題の特徴	状態遷移が連続的，評価値揺らぎなし，状態は 4 次元連続値，行動は離散値 3 行動	状態遷移が離散的，評価値揺らぎあり，状態は 7 次元離散値，行動は離散値 24 行動
UEBPGA SA 型	最も優れる	劣る，履歴の利用可
SR 型	優れる	最も優れる
SS 型	優れる	優れる，履歴の利用可
NNGA	中程度	劣る
突然変異率	0.1 程度が良い	低いほど良い
局所性	高めが良い	低めが良い

い性能を示し，また SR 型・SS 型のように次状態を予測する必要もない．これらから，問題の特性・履歴の有無や探索条件に合わせ，SA 型・SR 型・SS 型を使い分けることができることの有用性が確認できた．

最後に，表 2 に示す最良パラメータの傾向として，ACROBOT では突然変異率や局所性が高いほうが良く（$R_{\mathrm{mute}} \geq 0.1, \alpha_{\mathrm{local}} = 1000, k_{\mathrm{NN_{SS}}} = 7$ など），TETRIS ではその逆（$R_{\mathrm{mute}} = 0.0, \alpha_{\mathrm{local}} = 10, k_{\mathrm{NN_{SS}}} = 75$ など）であることがわかる．突然変異に関する説明としては，TETRIS には評価値の揺らぎがあるため，突然変異で悪い事例が導入されたときにその政策を排除できず停滞に陥る一方で，ACROBOT では積極的に新しい事例を導入し，多くの可能性を試験したほうが良い結果を導くという仮説を立てることができる．局所性に関する説明としては，TETRIS の状態空間は ACROBOT よりも大きいために，汎化の必要性がより大きいという仮説を立てることができる．ただし例えば ACROBOT では，R_{mute} については SA 型・SS 型では 0.01 から 0.2 まで，SR 型では 0.1 から 0.3 まで，$k_{\mathrm{NN_{SA}}}$ は 3 から 13 まで，α_{local} は 300 から 10000 までの間で，それぞれ 9 割以上の試行でゴール状態到達解を発見しており，少なくともこの問題ではパラメータに極めて敏感というわけではない．

5. 考　　察

5·1 TETRIS において状態評価戦略が成功した理由

性能を比較したときに最も顕著なのは，TETRIS では現在の状態から直接行動を導く戦略の性能が極めて低く，SR 型・SS 型 EBP のように状態評価戦略を持つ政策の性能が高い（図 12 参照）点である．これには，状態遷移の非連続性が大きく影響していると考えられる．SA 型 EBP では，「状態 s' では行動 a' が良い」「現在の状態 s は，状態 s' に近い」という二点から，「現在の状態 s でも行動 a' が良い」という結論を導いている．ここには暗黙的に，

- **仮定 1：** 近い二つの状態で同じ行動をとれば，近い状態に遷移する．
- **仮定 2：** 近い状態なら，好ましさも近い．

とが置かれている（図 14）．実際 ACROBOT のように連続的な系で 1 ステップが短い時間（0.05 秒）であれば，この仮定は合理的である．しかし，TETRIS のように離散的状態遷移を持つ問題では，仮定 1 は成立しない場合が多く（図 15），全く同じ状態を参照できる程度に事例が多くなければ適切な結論を導けない．

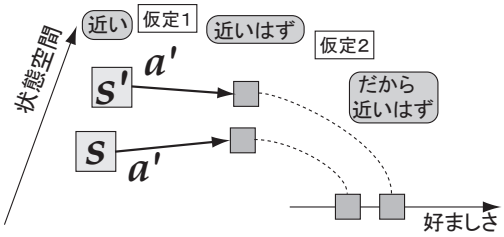

図 14 状態–行動型 EBP がうまくいく場合．近い状態で同じ行動をとると，遷移先の状態も近く，またその好ましさも近い．

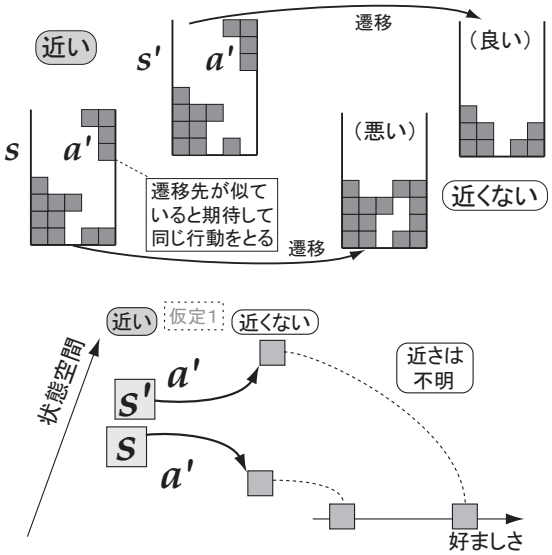

図 15 TETRIS で状態–行動型 EBP がうまくいかない理由．近い状態で同じ行動をとったとしても遷移先が大きく異なる場合がある．このような場合は好ましさも近いとは限らず，結果的に悪い行動をとってしまう可能性がある．

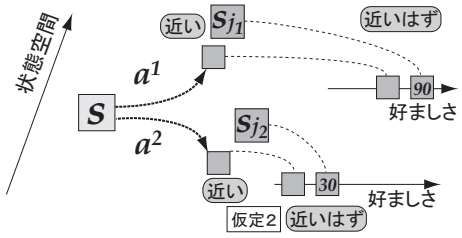

図 16 状態–価値型の戦略．次状態 s^1, s^2 を求め，それぞれに距離が近い事例 $(s_{j1}, 90)$ および $(s_{j2}, 30)$ を参照する．距離が近い事例とは好ましさも近いため，s^1, s^2 の好ましさ，ひいてはその優劣を予測できる．

従って，このように状態遷移に関する位相の連続性が低い問題では，次状態を計算することでその不連続性を緩和し，一方仮定 2（状態空間と評価の間の位相の連続

性は高いこと）を期待して状態評価戦略をとることが有望になるのである（図 16）．逆に言えば，一般に状態同士の距離は問題ごとにさまざまに定義できるが，CBR を用いた状態評価戦略にとっては仮定 2 が満たされるように距離を定義することが望ましい．

5·2 関連研究，今後の展開

本論文では，EBP-GA について，複数の表現形式が選択可能であること，知識の導入が容易であること，CBR による政策表現能力の高さ，GA による最適化能力の高さなどを実験を通じて示した．この他にも，多目的最適化が容易で自然に行えることは DPS の共通の特長であり，複数の競合する目的が生じやすい政策最適化には適した特徴である．また，EBP の実装例としては状態–状態型と状態–価値型の 2 つを提案し，最適化には UEBPGA を用いたが，TD 学習や Learning Classifier System[Holland 1986] の学習アルゴリズムの多くが政策の表現形式に依存しているのとは対照的に，EBP-GA では政策のモデルと事例集合の最適化方法を独立に選択可能であり，その可能性は広い．

まず政策のモデルについて考えてみると，事例が「状態 $s\in\mathcal{S}$ で行動 $a\in\mathcal{A}$ をとる好ましさは $r\in\mathbb{R}$ である」ことを表し Q 値と似た意味を持つ状態–行動–価値型 EBP なども利用可能であろう．また [Rosenstein 2001] では，ロボットに重量挙げ制御を行わせるために，初期状態からゴール状態までの軌道が経由する複数の点 (via-points) を直接探索している．このとき次の経由点までの行動選択は単純な PD 制御によって行われるが，適切な経由点が DPS によって求まれば，複雑な制御を達成することができる．このような接近法も，「第 i 番目には $s\in\mathcal{S}$ を通るべきである」という事例による EBP であると解釈することが可能である．さらに，例えば SAP, SLIP などでは Nearest Neighbor 法を用いているが，この行動選択は状態集合 $\mathcal{S}$ から取るべき行動集合 $\mathcal{A}$ へのクラス分類に他ならないから，Support Vector Machine などのより高度なクラス分類手法を用いることも可能である．

次に事例集合の最適化方法について考えてみると，例えば交叉操作で両親どちらかに近い解を生成することで多様性を維持するなどの工夫がすでに提案されている [土谷 2006]．また，行動集合 $\mathcal{A}$ の連続性・多次元性などにより，事例の混合による交叉だけでは効率的に新しい政策が作成できないような場合のために，親個体の事例から実数値交叉的な操作で新しい事例を導入する工夫も提案されている [宮前 2009]．多様性維持や評価の不確実性などは GA 一般に長く取り組まれてきた課題であり，それらの知見を生かした EBP-GA の最適化部分の実装が期待できる．

以上のように，事例ベースの政策最適化には，事例の表現形式・事例集合からの行動選択法・事例集合の最適化手法の 3 つの展開の軸がある（図 17）．この意味で本論文の貢献を位置づけると，事例の表現形式を拡張し，それぞれに行動選択法を提案し，汎用的な事例集合の最適化手法を示したことであるといえる．

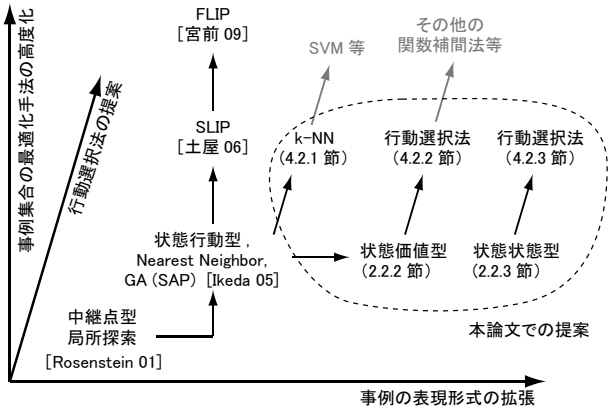

図 17 事例ベース政策最適化の 3 つの展開の軸と本論文の位置づけ：[Rosenstein 2001] では中継点集合と局所探索を用いての最適化が提案された．SAP[Ikeda 2005] では状態–行動対の集合と GA による最適化が提案された．[土谷 2006] や [宮前 2009] はその最適化部分を高度化させた提案である．他方，状態–行動対集合を用いた行動選択法として本論文では k-Nearest Neighbor を提案したが、SVM などの利用も今後提案されるだろう．本論文の最大の貢献は状態–価値対，状態–状態対を用いた事例ベース政策の可能性と，その行動選択法の実装例を示したことである．今までの発展と同様，行動選択法や最適化部分の高度化も今後期待できる．

最後に，本論文で提案した手法の応用範囲について考察する．本論文では議論展開の容易さのために対象として episodic で次状態が予測可能な MDP のみを取り上げたが，実際には予測可能 MDP の定義を厳密に満たさなくても提案手法（事例ベースの状態評価戦略と GA）が有効に機能する場合は多いと予想する．具体的には以下のような場合である．

- episodic でない場合．例えば 1 年中連続動作するシステムの制御であっても，1 時間分の評価で政策評価として十分だと判断できるならその評価値を用いて GA を行える．
- 次状態が唯一に定まらない場合．状態遷移が確率的な場合でも，その確率が分かっているなら，状態–価値型の EBP によってそれぞれの次状態の価値を推論し，その重みつき平均を求めることで行動を決定できる．
- 誤った次状態が予測される場合．その間違いの程度や確率に依存して性能は劣化する．ただし，アルゴリズムを適用できないという訳ではない．
- 次状態についての一部の情報しか得られない場合．状態評価戦略では，次状態そのものが得られる必要はなく，好ましさに影響する特徴量が得られればよい．したがって，情報が不完全であっても性能が劣化しない場合がありうる．

言い換えれば，何らかの意味での政策の評価が可能で，次状態に関して多少不正確であっても好ましさに影響する特徴量が得られる場合は，本手法を試す価値があると

考えている．なお，著者らはエレベータシステムの制御にも同様の戦略を用いて成果を上げており [池田 2006]，これは別論文で詳しく報告する予定である．

6. ま と め

本論文では，マルコフ決定過程に定式化される動的制御・意思決定問題に対する接近法として，政策を状態–行動対の集合で表し，現在の状態に最も近いものを参照するという政策表現とその GA による最適化に着目した．そしてこの枠組みを拡張・一般化し，様々な形式を持つ事例集合とそれに応じた事例ベース推論を用いた政策表現を用いることができる EBP-GA の枠組みを提案した．また具体的に状態–価値型 EBP，状態–状態型 EBP という二つの新しい戦略を提案し，これを特徴の異なる二つの問題に適用することで，問題の特徴や事前知識の有無などに応じた戦略の切り替えが有効であることを示した．EBP-GA は CBR の持つ汎化・局所化性能と GA の持つ強力な探索性能を備え，また多様な行動選択アルゴリズム利用できることから，DPS 型の政策最適化の実用的な枠組みとしての応用が期待できる．

◇ 参 考 文 献 ◇

[Aha 1997] David W. Aha: Lazy Learning, Kluwer Academic Publishers (1997)

[井口 2001] 井口 圭一，木村 元，小林 重信: GA による並列二重倒立振子の振り上げ安定化制御, 計測自動制御学会第 13 回自律分散システムシンポジウム, pp. 277–282 (2001).

[池田 2006] 池田 心，鈴木 裕通，喜多 一，マルコン シャンドル: マルチカーエレベータのスケジューリング問題, 計測自動制御学会 システム・情報部門学術講演会 2006，pp. 137–142 (2006).

[Ikeda 2005] Kokolo Ikeda: Exemplar-Based Direct Policy Search with Evolutionary Optimization, *2005 IEEE Congress on Evolutionary Computation*, pp. 2357–2364 (2005).

[Graham 2001] Graham Kendall, and Glenn Whitwell: An Evolutionary Approach for the Tuning of a Chess Evaluation Function using Population Dynamics, *Proceedings of the Genetic and Evolutionary Computation Conference*, pp. 995–1002 (2001).

[Holland 1986] Holland, J.H. : Escaping brittleness: the possibilities of general-purpose learning algorithmsapplied to parallel rule-based systems, *Machine Learning 2*, pp. 593–623 (1986).

[喜多 2000] 喜多 一，小野 功，小林 重信: 実数値 GA のための正規分布交叉の多数の親を用いた拡張法の提案, 計測自動制御学会論文集 , Vol.36 , No.10 , pp. 875–883 (2000).

[Knuth 1975] Knuth, D.E. and Moore, R.W. : An Analysis of Alpha-Beta Pruning, *Artificial Intelligence*, Vol. 6, No. 4, pp. 293–326 (1975).

[松井 2001] 松井純，土手斉，吉田和夫 : 論理及び直感の情報処理機構を併せもつゲームプレイ知的ロボット, 第 13 回自律分散システム・シンポジウム 2001, pp. 363–366 (2001).

[宮前 2009] 宮前惇，佐久間淳，小野功，小林重信 : インスタンスベース政策最適化のための実数値 GA と非ホロノミック系制御への適用，人工知能学会論文誌，24 巻 1 号 SP-J, pp. 104–115 (2009).

[Moriarty 1999] David E. Moriarty, Alan C. Schultz, and John J. Grefenstette: Evolutionary algorithms for reinforcement learning, *Journal of Artificial Intelligence Research 11*, pp. 241–276 (1999).

[野村 2001] 野村壮太郎, 吉田和夫 : 進化型強化学習のテトリスゲームへの適用，第 13 回自律分散システム・シンポジウム, 2001, pp. 265–270 (2001).

[Rosenstein 2001] Rosenstein, M.T. and Barto, A.G.: Robot weightlifting by direct policy search, *Proceedings of the 17th International Joint Conference on Artificial Intelligence*, vol. 2, pp. 839–844 (2001).

[Satoh 1996] H.Satoh, M.Yamamura, and S.Kobayashi: Minimal Generation Gap Model for GAs Considering Both Exploration and Exploitation, *Proc. of IIZUKA*, pp. 494–497 (1996).

[Sheppard 1997] Sheppard, J.W. and Salzberg, S.L.: A teaching strategy for memory-based control, *Artificial Intelligence Review 11*, pp. 343–370 (1997).

[Spong 1994] Spong, M.W.: Swing up control of the acrobot, *Proceedings of the 1994 IEEE Conference on Robotics and Automation*, pp. 2356–2361 (1994).

[Sutton 1998] Sutton, R. S. and Barto, A.: Reinforcement Learning: An Introduction, *A Bradford Book, The MIT Press* (1998).

[Szita 2006] Szita, I. and Lorincz, A.: Learning Tetris Using the Noisy Cross-Entropy Method, *Neural Computation 18*, pp. 2936–2941 (2006).

[土谷 2006] 土谷千加夫, 塩川裕介, 池田心, 佐久間淳, 小野功, 小林重信: ハイブリッド GA によるインスタンスベース政策学習 SLIP の提案と評価, 計測自動制御学会論文集, vol.42 no.12, (2006)

[van der Wal 1981] van der Wal, J.: Stochastic dynamic programming, *Mathematical Centre Tracts No. 139, Mathematisch Centrum, Amsterdam*, (1981).

[Watkins 1992] Watkins, C.J.C.H., Dayan, P.: Q-learning, *Machine Learning, 8*, pp. 179–292 (1992).

[Whitley 1988] Whitley, D. and Kauth, J.: GENITOR: A different genetic algorithm, *Proceedings of the Rocky Mountain Conference on Artificial Intelligence*, pp. 116–121 (1988).

〔担当委員：村田 忠彦〕

2009 年 7 月 31 日 受理

著 者 紹 介

池田 心

1999 年東京大学理学部数学科卒業，2003 年東京工業大学大学院総合理工学研究科博士課程修了，博士（工学）．同年京都大学学術情報メディアセンター助手，2007 年 4 月助教．2010 年 1 月より北陸先端科学技術大学院大学情報科学研究科准教授．主に遺伝アルゴリズムによる最適化，囲碁，ゲーム，エージェントシミュレーションなどの研究に従事．

小林 重信（正会員）

1969 年東京工業大学理工学部応用化学科卒業，1971 年理工学研究科化学工学専攻修士課程修了，1974 年理工学研究科経営学工学専攻博士課程修了，工学博士．同年東京工業大学工学部制御工学科助手，1975 年総合理工学研究科システム科学専攻助手，1981 年助教授．1990 年同研究科生命化学専攻教授，1991 年同研究科知能科学専攻教授，1996 年同研究科知能システム科学専攻教授，現在に至る．創発システム論，生物的適応システム，進化計算，強化学習などの研究に従事．

喜多 一

1982 年京都大学工学部電気工学科卒業．87 年同大学大学院工学研究科博士後期課程研究指導認定退学．同大学工学部助手，97 年東京工業大学大学院総合理工学研究科助教授，2000 年大学評価・学位授与機構教授．2003 年より京都大学学術情報メディアセンター教授．工学博士．進化的計算，エージェントシミュレーションなどの研究に従事．電気学会，計測自動制御学会，システム制御情報学会，日本シミュレーション学会，神経回路学会，日本教育工学会，進化経済学会，組織学会，社会経済システム学会会員，国際プロジェクト・プログラムマネジメント学会会員．