対話的強化学習のための多様な準最適政策の探索
－巡回最適性に基づく LC-Learning－

佐藤 和宏† 山口 智浩‡

†奈良工業高等専門学校専攻科 〒639-1042 奈良県大和郡山市矢田町 22 番地
‡奈良工業高等専門学校情報工学科 〒639-1042 奈良県大和郡山市矢田町 22 番地
E-mail: †kazuhiro@info.nara-k.ac.jp, ‡yamaguch@info.nara-k.ac.jp

あらまし 本研究の目的は，HAI において教示側を支援する知的な学習エージェントの実現である．従来の強化学習手法は固定の学習目標に対する最適政策獲得を目的としてきたため，一般ユーザが対話的に目標を追加する対話的強化学習は想定外である．ユーザ教示の修正や曖昧な教示の明確化を対話的に解決するには，流動的な学習目標に対し，多様な学習結果の提示が有効である．そこで本研究では，新しい学習基準として**巡回最適性**を定義し，多様な巡回最適政策を求める手法を提案する．そして最適政策探索手法である Modified-PIA との比較実験により，提案手法が多様な巡回最適政策を求める手法として有効であることを示す．

キーワード 対話的強化学習，Shaping，巡回最適性，多様な政策

Preparing various policies for interactive reinforcement learning
－LC-Learning based on every visit optimality－

Kazuhiro SATOH† Tomohiro YAMAGUCHI‡

†Faculty of Adv. Eng., Nara national collage of techology 22 Yata-cho, Yamatokoriyama-shi, Nara, 639-1042, Japan
‡Dep.of Info. Eng., Nara national collage of techology 22 Yata-cho, Yamatokoriyama-shi, Nara, 639-1042, Japan
E-mail: †kazuhiro@info.nara-k.ac.jp, ‡yamaguch@info.nara-k.ac.jp

Abstract The purpose of this research is realizing the intelligent learning agent that support teaching in human agent interaction (HAI). Normally, reinforcement learning systems is intended to acquire an optimal policy for fixed goals. So reinforcement learning is not available to interactive reinforcement learning in which goals are added interactively by an end user. To adjust the user's teach or make clear the user's goal, it is available to show the user various result of learning for unclear user's goal. In this research, we propose the concept of *every-visit-optimality* (*ev-optimality*) and the reinforcement learning method that preparing various policies. And we show the proposal method is valid to preparing various ev-optimal policies by the experimental result.

Keyword Interactive reinforcement learning, Shaping, *Every-visit-optimality*，*Various policies*

1. はじめに

近年，家庭用ロボットが一般的になりつつある．例えば家庭用ロボットの有力な用途の 1 つとして AIBO のような人とのコミュニケーションを目的としたエンターテインメントロボットがある．このような家庭用ロボットは，使用される環境や要求される目的が各家庭やユーザによって異なる．そこで一般ユーザが各家庭において自ら家庭用ロボットの機能拡張やパーソナライズを行うことが必要になると考えられる．

家庭用ロボットのパーソナライズの先行研究の 1 つに，AIBO 等のペットロボットと人間とのコミュニケーション方法の獲得を目的として，ゴールを表す報酬を対話的に与える強化学習法[1]がある．この手法の特徴は，従来は固定かつ組み込みだった報酬関数を学習時に対話的に学習者に与える点である．これにより学習される行動系列を，逐次的に洗練することが可能となる．

この対話的な強化学習法の理論的な枠組みとして shaping[2]がある．shaping は，与えられたメインゴールに対し学習者を誘導するサブゴール系列となるように shaping 報酬関数を追加することで，メインゴールへ至る複雑な行動系列の形成を加速する手法である．従来の shaping は，報酬関数の設定に対し以下の 3 つの仮定が必要となる．

- メインゴールが既知.
- メインゴールへの単純な距離関数として，サブゴールを shaping 報酬として仮定.
- shaping 報酬は政策不変である[3].（最適政策の決定に影響しない）

従来手法では上述の仮定を満たす報酬関数を設計者が作成しシステムに組み込んでいた．しかしながらユーザが対話的に学習エージェントに報酬を与える対話的強化学習では，これらの仮定が成り立つとは保証できない．なぜなら shaping の知識を持たないユーザが報酬を設定した場合，上述の仮定 2,3 を満たさない報酬が設定される事があると考えられるからである．そのため対話的強化学習では対話的に入力された報酬を，上記の仮定を満たす報酬と満たさない報酬とに区別する必要がある．直接報酬を区別できないのであれば，間接的な解決策として，学習政策をユーザに提示して，学習すべき政策をユーザに選択してもらう方法が考えられる．

そこで本研究ではこの区別を対話的に行うための強化学習エージェントの機能拡張法を提案する．まず獲得政策の新しい評価基準として，全ての報酬を訪問し，かつ平均報酬が最も大きくなる政策を，**巡回最適政策**と定義する．次に与えられた報酬関数の全ての部分集合に対する巡回最適な政策を**多様な政策**と定義し，これらを効率よく求める強化学習法を提案する．この多様な政策をユーザに提示し，所望の政策を教示してもらうことにより，上述の shaping 報酬に関する仮定を満たす報酬と満たさない報酬を対話的に分離することが可能となる．

本手法の狙いはユーザの教示を部分的に満たす行動系列をユーザに提示する事が可能となる事である．これによりユーザの教示のインタラクティブな修正やあいまいな教示の明確化が期待される．

以下，2 章では提案する強化学習手法の枠組みを説明する．3 章では提案する手法の中の，特に多様な政策を探索するアルゴリズムについて詳しく説明する．4 章では提案手法の有効性を検証するための実験について述べる．5 章では結論として論文全体のまとめと今後の課題について述べる．

2. 提案システム

本章では提案する強化学習の枠組みについて述べる．図 1 に提案する強化学習システムの概要を示す．ここで s は観測された状態を表し，a は実行した行動を表す．そして Rw は獲得された報酬を表す．

図 1 に示すように，提案する強化学習エージェントは 3 つのブロック，モデル同定ブロック，政策の最適性を決定するブロック，政策探索ブロックから構成される．以下の節では，これらのブロックの概要について説明する．本提案手法の新規性は政策の最適性のブロックと政策探索のブロックにある．

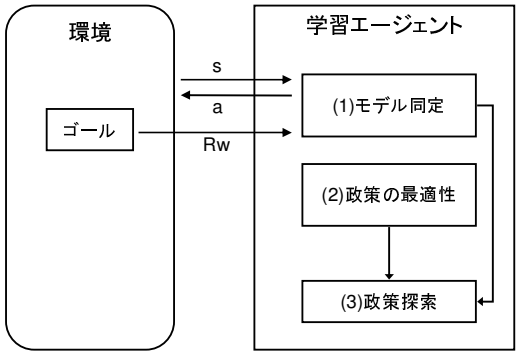

図 1 強化学習の枠組み

2.1. モデル同定

モデル同定ブロックは，観測した(s,a,Rw)の系列に基づきモデルの状態遷移確率 $p(s'|s,a)$と報酬関数 $Rw(s,a)$を逐次最尤推定する．本研究では環境を，エルゴード性を持つ単純 MDP(Markov Decision Process:マルコフ決定過程)としてモデル化する．

2.2. 政策の最適性

このブロックは強化学習で求める政策の最適性を定義する．政策とはエージェントの意思決定の基準となるもので，各状態における実行ルールの集合である．本研究では，平均報酬最大となる決定的政策を最適であると定義する．さらに巡回最適という概念を，平均報酬に基づいて定義する．この詳細は 3.1 節で述べる．

式(1)に初期状態 s から政策 π に基づき状態遷移を無限回繰り返した時の平均報酬 $g^{\pi}(s_0)$の定義を示す．

$$g^{\pi}(s_0) \equiv \lim_{N\to\infty} E\left(\frac{1}{N}\sum_{t=0}^{N-1} r_t^{\pi}(s_0)\right) \tag{1}$$

ここで N はステップ数，$r_t^{\pi}(s_0)$は初期状態 s_0 から政策 π に基づきエージェントが状態遷移をしたときにステップ t で獲得する報酬である．また，$E()$は期待値を表す．

2.3. 政策探索

このブロックは 2.1 節で同定したモデル上で，3.1 節で述べる**巡回最適性**を用いて定義する**多様な政策**を探索する．詳細については 3.3 節で説明する．

3. 多様な政策の探索

本章では，まず**多様な政策**の定義に必要となる概念である**巡回最適性**について定義する．そして**巡回最適性**の概念に基づき**多様な政策**を定義し，**多様な政策**を求める強化学習手法について説明する．

3.1. 巡回最適政策の定義

巡回最適性を定義するために，まず政策の定常サイクルと一時パスについて説明する．本研究では政策を定常サイクルと一時パスの和で表す．定常サイクルとは任意の初期状態から，ある政策の元でエージェントが状態遷移を繰り返した時，1 ステップあたりの生起確率が正となるルールの集合である．一時パスとは上記の生起確率が 0 に収束するルールの集合である．この定常サイクルの概念を用いて，巡回最適政策を定義する．巡回最適な政策とは，政策を構成する定常サイクルが報酬集合の全ての報酬を獲得する政策の中で，平均報酬を最大化する政策である．

3.2. 多様な政策の定義

多様な政策は以下の 3 ステップの処理で定義される．

(1) 報酬関数の全ての部分集合を列挙する．
(2) それぞれの部分集合に対して巡回最適政策を求める．
(3) ステップ 2 で求めた全ての巡回最適政策の集合を多様な政策とする．

図 2 にこれらのステップの処理の流れを図示する．図 2 において矢印が処理を，ブロックが入出力データを表す．

まず入力となる報酬関数が*{Rw1,Rw2}*と定義されている場合，ステップ 1 で列挙される部分集合は*{Rw1}*, *{Rw2}*, *{Rw1, Rw2}*となる．次にステップ 2 でそれぞれの部分集合に対する巡回最適政策を求める．最後にステップ 3 で全ての巡回最適政策を集めて，多様な政策集合として出力する．このとき多様な政策中の政策数の最大値は報酬数を r とすると 2^r-1になる．

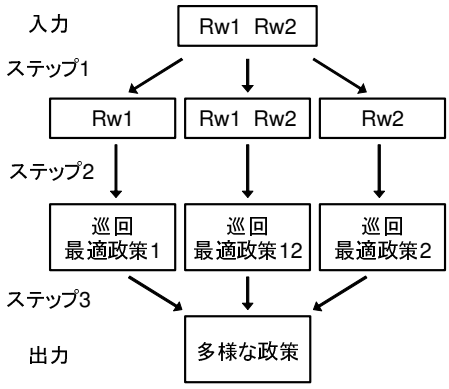

図 2 多様な政策探索の手順

3.3. 多様な政策の探索手法

本研究では LC-Learning[4][5]をベースとした，多様な政策を探索する手法を提案する．以下では提案する手法を**網羅 LC-Learning** と呼ぶ．LC-Learning とは，平均報酬を学習規範として用いるモデルあり強化学習法である．LC-Learning では以下の 3 ステップにより最適政策を求める．

(1) 報酬獲得ルールをルートとして MDP を木構造に展開する事で，報酬獲得政策を網羅的に探索．
(2) 状態の生起確率を用い，網羅的に求めた政策全ての平均報酬を計算．
(3) 平均報酬最大の政策を最適政策として決定．

網羅 LC-Learning では従来のステップ 3 を以下のステップ 3'に変更することで多様な政策を求める．

(3') 網羅的に求めた政策から，巡回最適政策を全て選択する．

網羅 LC-Learning のステップ 1,2,3'の詳細について，以下に説明する．

(1) 報酬獲得政策の探索

ステップ 1 では，全ての報酬獲得ルールをルートとして MDP を幅優先探索により木構造に展開する事で，報酬獲得定常サイクルを網羅的に探索する．そして求まった定常サイクルに対して一時パスを決定することで政策を求める．ただし政策の平均報酬最適性や巡回最適性は政策の定常サイクルのみによって決まるため，一時パスの決定方法についてはここでは省略する．

定常サイクルの探索方法を図 3，図 4，図 5 を用いて説明する．図 3 に示す 4 状態，6 ルール，2 報酬の MDP の場合，報酬獲得ルールが 2 つなので図 4 の(a), (b)に示す，各報酬獲得ルールをルートノードとする 2 つの木構造に展開される．まずルートノードでは報酬獲得ルールのみを展開し，以降のノードでは全てのルールを展開する．あるノードがルートノードからのパス上で既に展開されているなら，そのノード以下は展開しない．なぜなら，これらのパスは下記のいずれかのサイクルとなるからである．

この木構造において次の 2 種類のサイクルによって，1 つの定常サイクルが表現される．

(1) メインサイクル：ルートノードから，ルートノードと同じ状態を表現するノードへのパス．
(2) 部分サイクル：メインサイクル上のルールから確率的枝分れにより分岐し，メインサイクル上のノードへ戻るパス．

この様に，報酬獲得ルールごとに 1 つの木構造を求めた後に，複数の木構造間で同一な定常サイクルを取り除くことで，全ての報酬獲得定常サイクルが検出される．図 5 は図 3 の全ての報酬獲得定常サイクルを示す．

(2) 生起確率に基づく平均報酬の算出

このステップでは，全ての政策に対して政策を固定した時の各状態の生起確率から政策の平均報酬を求める．状態の生起確率とは，エージェントがある政策に基づいて状態遷移を無限回繰り返した時に，その状態が生起する確率の期待値である．状態 s_i から遷移を繰

り返した時の状態 s_j の生起確率の定義を式(2)に示す．

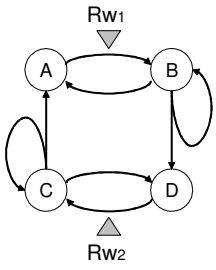

図 3 同定された MDP の例

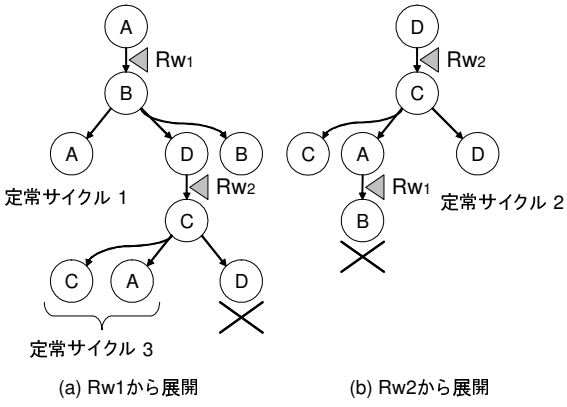

図 4 報酬獲得定常サイクルの探索

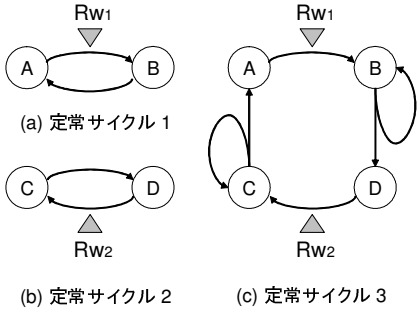

図 5 三種類の報酬獲得定常サイクル

$$P(s_j,s_i)=\begin{cases}1 & (j=i)\\ \sum_{s_k} P(s_k,s_i)p(s_j \mid s_k,a_k) & (j\neq i)\end{cases} \tag{2}$$

ここで s_i, s_j はともに状態集合の任意の要素を表し，a_k は政策を固定した時に状態 s_k で選択する行動を表す．式(2)を全ての状態について立てると，未知数が状態数，式の数が状態数の同次連立方程式となる．この同次連立方程式を解くことにより全ての状態の生起確率を求めることが出来る．本アルゴリズムでは，値反復により近似的に同次連立方程式を解き，状態の生起確率を求める．

そして値反復により求めた生起確率を用いて，以下の式(3)で政策の平均報酬 g^{π} を求める．

$$g^{\pi}(s_i)=\frac{\sum_{s_j} P(s_j,s_i)Rw(s_j,a_j)}{\sum_{s_j} P(s_j,s_i)} \tag{3}$$

(3) 巡回最適政策の決定

このステップでは木展開により検出した政策から全ての巡回最適政策を選択する．まず木展開により検出した政策を，獲得報酬集合の違いによりグループ分けする．次に各グループの中で平均報酬が最大になる政策を巡回最適な政策として決定する．全てのグループに対する巡回最適な政策の集合が多様な政策となる．

4. 実験

本章では，提案手法である網羅 LC-Learning が，従来手法に比べて，多様な政策を求める手法として有効であることを，比較実験により明らかにする．ここで平均報酬に基づく代表的な従来の手法として Modified-PIA[6]を用いる．Modified-PIA は平均報酬を最適性の評価基準に用いるモデルあり強化学習法である．比較のためここでは，最適政策のみを求める Modified-PIA に以下の処理を加えて多様な政策を求めている．

(1) 報酬関数の全ての部分報酬集合それぞれで定義された報酬関数の集合を求める．
(2) それぞれの報酬関数に対する最適政策を，Modified-PIA を用いて求める．
(3) 求めた最適政策の和集合を求め，巡回最適政策の集合を多様な政策とする．

本章ではこれらの処理を加えた Modified-PIA を**網羅 Modified-PIA** と呼ぶ．

4.1. 実験方法

以下の 3 項目に対して比較実験を行う．

(1) 環境モデルのパラメータの増減に対する計算コストの増減．
(2) 検出される全巡回最適政策数．
(3) 獲得報酬数別の検出される巡回最適政策数．

比較項目 1 に関しては，環境モデルのパラメータの増減に対する計算コストの増減が線形となることが望ましい．ここで増減させる環境のパラメータとしては，状態数，行動数，報酬数の 3 種類のパラメータを用いる．これら 3 種類のパラメータのうち，特に逐次的に報酬を追加する対話的強化学習では，報酬数の増加に対して計算コストが指数的に増加しない事が重要である．次に，比較項目 2 に関しては全体として多くの巡回最適政策が求まることが望ましい．最後に，項目 3 に関しては獲得報酬が多い，複雑な政策をより多く検出していることが望ましい．

実験は環境モデルを用いたシミュレーションによ

り行う．環境モデルとして状態遷移確率とルールに対する報酬値をランダムに設定した MDP を用いる．状態数，行動数，報酬数は表 1 に示す条件を用いる．

比較項目 1 の調査として，表 1 の条件 1，条件 2，条件 3 の環境モデルを用い，状態数，行動数，報酬数を独立に変化させたときの計算コストについて調査する．また条件 3 の環境モデルについては，比較項目 2 の調査として，検出巡回最適政策数についての調査も行う．また条件3のうち報酬数6のモデルについては，比較項目 3 の調査として，獲得報酬数別の検出巡回最適政策数についての調査も行う．

ここで計算コストの指標として，網羅 LC-Learning では状態の生起確率を，網羅 Modified-PIA では状態価値を求めるための反復計算の回数を用いる．

表 1 実験に用いた環境モデルのパラメータ

	状態数	行動数	報酬数
条件 1	3～10	2	4
条件 2	10	2～4	4
条件 3	10	4	1～10

4.2. 実験結果

実験結果を以下にまとめる．

- 計算コストに関しては，提案手法と従来手法は同程度の傾向を示した．
- 但し，報酬数増加に対する計算コストは提案手法に優位性が見られた．
- 検出した政策の多様性は，提案手法に優位性が見られた．

以下，実験結果について詳しく述べる．まず条件 1～3 の環境モデルに対する計算コスト増加の傾向を表 2 に示す．特に条件 3 の結果については図 6 に詳細を示す．また条件 3 に対する検出巡回最適政策数を図 7 に示す．また条件 3 の報酬数 6 のモデルに対する，獲得報酬数別の巡回最適政策の検出数を図 8 に示す．図 8 の縦軸は網羅 LC-Learning で検出した巡回最適政策数に対する，網羅 Modified-PIA で検出した巡回最適政策数の割合である．網羅 LC-Learning では全巡回最適政策が求まることが保証されているため，この値は全巡回最適政策中の網羅 Modified-PIA が検出した政策数の割合となる．

図 7 の結果より，網羅 Modified-PIA で検出する巡回最適政策は網羅 LC-Learning で検出する数に比べ少ないことが分かる．そして図 8 の結果より獲得報酬数が 2 の巡回最適政策がもっとも検出する割合が多く，獲得報酬数 1 や 4,5,6 になると検出する割合が少なくなることが分かる．特に獲得報酬数 4,5,6 の場合は巡回最適政策を見つけることが出来ていないことが分かる．

表 2 環境の増減に対する計算コストの増減の性質

	Modified-PIA	LC-Learning
条件 1	非線形	非線形
条件 2	線形	非線形
条件 3	非線形	線形

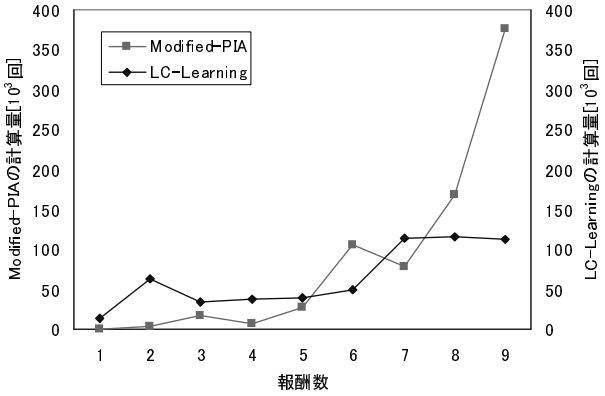

図 6 報酬数増減に対する計算コストの増減

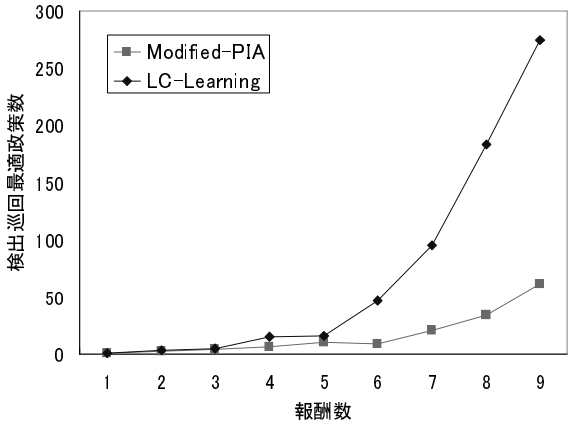

図 7 報酬数増減に対する検出巡回最適政策数

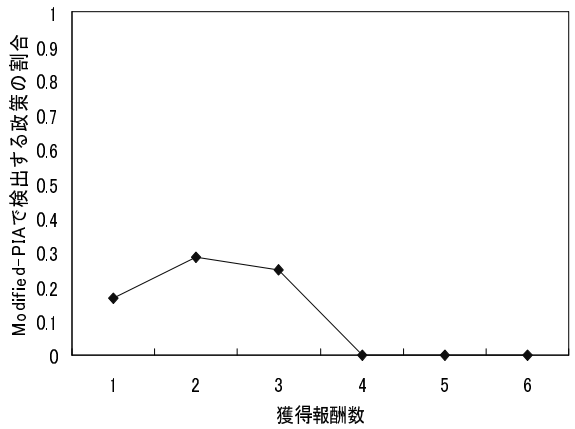

図 8 報酬数 6 の時の検出巡回最適政策数の比

5. 考察

4 章の実験結果に対し，以下の 3 点について議論する．

5.1. 報酬数増減に対する両手法の計算コスト

網羅 Modified-PIA では対象となる MDP の報酬関数に対して全ての部分報酬集合を求め，それぞれに対して通常の Modified-PIA を実行している．そのため報酬数を r とすると，2^r-1が部分報酬関数の場合の数となるため 2^r-1回通常の Modified-PIA を実行する事になる．そのため網羅 Modified-PIA の計算コストは報酬数 r の増加に対して指数的に増加すると考えられる．これに対し網羅 LC-Learning では展開する木構造の数が報酬数と等しくなる．そのため網羅 LC-Learning の計算コストは報酬数の増加に対して線形に増加すると考えられる．

5.2. 網羅 Modified-PIA の検出巡回最適政策数

第 1 に，網羅 Modified-PIA による全検出巡回最適政策数について解析する．図 7 より，網羅 Modified-PIA で検出する巡回最適政策は網羅 LC-Learning で検出する数に比べ少ないことが分かる．この理由として，網羅 Modified-PIA では報酬関数の全部分集合に対して最適政策を求めているが，異なる部分集合に対する最適政策が同じ政策になる事があるため，全巡回最適政策を求める事が出来ないと考えられる．

第 2 に，獲得報酬別の網羅 Modified-PIA による検出巡回最適政策数について解析する．図 8 より獲得報酬数が 2 の巡回最適政策が最も検出する割合が多く，獲得報酬数 1 や 4,5,6 になると検出する割合が少なくなることが分かる．特に獲得報酬数 4,5,6 の場合は巡回最適政策を見つけられない．これらは上述した報酬関数の複数の部分集合に対する最適政策が同じ政策となるために起こると考えられる．網羅 Modified-PIA では，最適政策中のある報酬獲得ルールが政策不変な場合，その報酬を獲得しない政策を検出することが出来ない．そのため獲得報酬数が 2 や 3 の巡回最適政策は検出率が高く，それよりも獲得報酬が多い政策や，少ない政策は検出率が落ちたと考えられる．これに対して網羅 LC-Learning では，意図的に Rw_0 を獲得する政策や，Rw_1 を獲得しない政策を求めるため，全ての巡回最適政策を検出することが出来る．

5.3. 行動数増加に対する計算コストの削減

網羅 LC-Learning の問題点である，行動数増加に対する計算コスト増加の問題点とその解決方法について議論する．この問題の原因は，提案手法が行動をアークとして MDP を木構造に展開することにある．解決方法の 1 つに，巡回最適政策になる見込みのない枝の枝刈りが考えられる．ただしそのためには巡回最適政策になるかを枝の展開時に判断する必要があり，早い段階で十分な枝刈りを行うことは難しいと予想される．他の方法として，木構造に展開する際に巡回最適政策になる可能性の高い枝 N 本のみを展開し，それ以外の枝を全て枝刈りする方法が考えられる．この方法は一般的にビームサーチと呼ばれる手法である．この方法を用いれば木構造に展開する際の計算コストが行動数と無関係になるため全体の計算コストは行動数に対して線形な性質になると考えられる．ただしこの手法を用いると，提案手法の重要な利点である，全巡回最適政策の検出という性質が失われてしまう．そのためビームサーチを網羅 LC-Learning に応用するための課題として，ビームサーチを用いた際に，全巡回最適政策中の何割を検出可能であるか，また対話的強化学習において巡回最適政策の何割を求める必要があるかなどを実験的に検証することなどがあげられる．

6. おわりに

本研究では，家庭用ロボットをパーソナライズするための対話的強化学習に，多様な解を用いることを提案した．そして多様な解を求める強化学習手法として LC-Learningの拡張を提案し，その有効性を検証する実験を行った．

今後の課題として，行動数の変化に対する提案手法の計算コストを削減すること，また本手法を対話的強化学習システムに適応することが挙げられる．

文　献

[1] F. Kaplan, P-Y. Oudeyer, E. Kubinyi and A. Miklosi, “Robotic clicker training,” Robotics and Autonomous Systems, Vol. 38, No. 3-4, pp.197-206, 2002

[2] George Konidaris, Andrew Barto, “Automonous Shaping: Knowledge Transfer in Reinforcement Learning,” Proc. of 23rd International Conference on Machine Learning, pp.489-496, 2006

[3] A.Y.Ng, D.Harada and S.Russell: "Policy invariance under reward transformations:Theory and application to reward shaping", Proc. of 17th International Conference on Machine Learning, pp.278-287, 1999.

[4] Taro Konda, Tomohiro Yamaguchi, “ LC-Learning, Phased Method for Average Reward Reinforcement Learning - Analysis of Optimal Criteria -,” PRICAI2002: Trends in Artificial Intelligence, M.Ishizuka and A.Sattar (Eds.), Lecture notes in Artificial Intelligence 2413, Springer, pp.198-207, 2002

[5] Taro Konda, Shinjiro Tensyo, Tomohiro Yamaguchi, “LC-Learning : Phased Method for Average Reward Reinforcement Learning - Preliminary Results -,” PRICAI2002: Trends in Artificial Intelligence, M.Ishizuka and A.Sattar (Eds.), Lecture notes in Artificial Intelligence 2417, Springer, pp.208-217, 2002

[6] M.L.Puterman, “Markov Decision Processes: Discrete Stochastic Dynamic Programming,” JOHN WILEY & SONS, INC, pp.385-388, 1994