バイオメディカル・ファジィ・システム学会誌 Vol.15, No.1, pp.69−81 (2013)

[Original article]

(2013年4月17日 Accepted)

マルコフ決定過程で表現されたロールプレイングゲームにおける攻略法の能動学習

前田 康成[1]，後藤 文太朗[1]，升井 洋志[1]，
桝井 文人[1]，鈴木 正清[1]，松嶋 敏泰[2]
1）北見工業大学・情報システム工学科 2）早稲田大学・応用数理学科

要約： 従来からマルコフ決定過程（MDP）を用いたロールプレイングゲーム（RPG）のモデル化が行われている．しかし，RPGの攻略法を能動的に学習する研究は行われていない．そこで，本研究では，真のパラメータが未知のMDPで表現されたRPGにおける期待総利得をベイズ基準のもとで最大にする攻略法を求める能動的な学習方法を提案する．シミュレーションをとおして，提案方法の有効性を確認する．
キーワード： ロールプレイングゲーム，マルコフ決定過程，能動学習

Active Learning Strategy for Role-playing Game Modeled by Markov Decision Processes

Yasunari MAEDA[1], Fumitaro GOTO[1], Hiroshi MASUI[1],
Fumito MASUI[1], Masakiyo SUZUKI[1], Toshiyasu MATSUSHIMA[2]
1) *Department of Computer Science, Kitami Institute of Technology*
2) *Department of Applied Mathematics, Waseda University*

Abstract: *In previous research a role-playing game(RPG) is represented with Markov decision processes(MDP). But active learning method for RPG has not been studied yet. In this research we propose an active learning method which maximizes an expected total reward with respect to a Bayes criterion under the condition that the true parameter of MDP is unknown. We recognize the effectiveness of our proposed method by some simulations.*

Keywords: *role-playing game, Markov decision processes, active learning*

Yasunari MAEDA
165 Koen-cho, Kitami-shi, Hokkaido, 090-8507, Japan
Phone: +81-157-26-9328, Fax: +81-157-26-9344, E-mail: maeda@cs.kitami-it.ac.jp

1. はじめに

近年，コンピュータの低価格化に伴い，テレビゲーム機が広く普及し，ゲームの一分野としてロールプレイングゲーム（以下，RPG と表記する）が広く普及している．また，工学分野ではRPG をマルコフ決定過程（以下, MDP と表記する）[1][2][3]を用いて表現して，ゲームを攻略する戦略（攻略法）を数理工学的に扱う研究[4][5][6]が行われている．

RPG にはいろいろな種類があるが，本研究では従来研究[6]同様にマップモードにおいてマップ上のプレイヤを移動させて, 敵と遭遇すると戦闘モードになり，敵を倒すと何らかの報酬（お金や経験値など）をプレイヤが得る冒険型の RPG を対象とする．

問題設定として, MDP の状態遷移確率を支配する真のパラメータが既知の場合と未知の場合の両方が考えられる．真のパラメータは開発者のみが知っている情報なので，一般的にゲームをプレイする側のユーザの遊び方（戦略）は真のパラメータ未知の場合に相当する．

また，真のパラメータ未知の場合には，真のパラメータに関する事前学習なしに報酬を最大化したい評価期間のみの問題設定，真のパラメータに関する事前学習用にプレイの履歴データを受動的に与えられたもとでの報酬を最大化したい評価期間の問題設定，真のパラメータに関する事前学習を行うために練習期間を与えられた練習期間と評価期間を合わせた問題設定が考えられる．1 番目は事前情報なしの問題設定，2 番目は事前情報を受動的に得る問題設定，3 番目は事前情報（評価期間に対する事前情報）を能動的に得る問題設定である．事前情報は，ユーザが本格的にプレイする前に行う練習によるユーザの経験（知識獲得）に相当する．ユーザの練習の際には，ユーザは何も考えずにランダムにプレイするわけではなく，何らかの戦略に従ってプレイしていると考えられる．よって，一般的にゲームをプレイする側のユーザの遊び方に最も近い問題設定は，上記の中であれば3 番目の問題設定である．

従来研究[6]では，真のパラメータ既知の場合，未知の場合の 1 番目の問題設定と 2 番目の問題設定が検討されているが，3 番目の問題設定は未検討である．そこで，本研究では真のパラメータ未知の MDP で表現される冒険型の RPG に対して, 報酬を最大化したい長さが有限の評価期間の前に練習用の長さが有限の練習期間を設けた問題設定で定式化し，統計的決定理論[7][8]に基づいて評価期間の報酬をベイズ基準のもとで最大化する攻略法を求める能動的な学習方法を提案する．また，シミュレーションをとおして提案方法の有効性を確認する．

従来研究[4][5][6]では, RPG を MDP を用いてモデル化することの目的として, RPG をユーザがプレイする際に楽しいと感じる要素を工学的に把握することや，RPG の自動開発などが挙げられている．これらの従来研究は基礎研究であり，RPG の自動開発などは直近の目的ではなく，将来的な目的である．本研究も従来研究と同様に基礎研究であり, 現時点で実際の RPG の開発現場に貢献することは出来ないが，将来的には実際の開発現場におけるコスト軽減などに貢献できる可能性がある．

2． MDP を用いた RPG

2.1 本研究で研究対象とする RPG

以下で, 本研究で研究対象とする RPG について説明する．なお, 本研究で研究対象とする RPG は従来研究[6]で想定されている RPG と同様のものである．

プレイヤはヒットポイント（以下，HP と表記する）と呼ばれる数値を持ち，HP が 0 になると次の期にマップ上のスタート位置から再開する．再開時には HP はスタート時と同じ最大値 M_{hp} まで回復する．

sm_i はマップ上の位置を示し，SM，$SM=\{sm_1, sm_2, \ldots, sm_{|SM|}\}$ はマップ上の位置の集合である．ゲーム開始時のスタート位置を sm_1 とする．スタート位置や現在のプレイヤの位置は既知である．f_i はマップ上の地形の種類を示し，F，$F=\{f_1, f_2, \ldots, f_{|F|}\}$ はマップ上の地形の種類の集合である．マップ上の各位置がどの地形に該当するかは，関数 $F(sm_i)\in F$ でわかる．

e_i は敵の種類を示し，E，$E=\{e_1, e_2, \ldots, e_{|E|}\}$ は敵の種類の集合である．$M(e_i)$ は敵 e_i 出現時の敵 e_i の HP を示す. プレイヤは敵を攻撃することによって敵の HP

を0以下にすると，その敵を倒し，その敵に該当する報酬$G(e_i)$を得る．

プレイヤが選択できる行動（コマンド）はマップモードと戦闘モードで異なり，マップモードではa_1からa_4が選択可能で，戦闘モードではa_5とa_6が選択可能である．a_1，a_2，a_3，a_4はそれぞれマップ上で右，左，上，下に移動するための行動である．$mv(sm_i, a_j)$はプレイヤが位置sm_iで行動a_jを選択した際の移動先の位置である．プレイヤの移動に際して，確率$p(e_k|F(mv(sm_i,a_j)),\theta^*)$で移動先$mv(sm_i,a_j)$に敵$e_k$が出現し戦闘モードになる．敵は同時に複数出現することはなく，確率$1-\sum_{e_k\in F} p(e_k|F(mv(sm_i,a_j)),\theta^*)$で何も出現せずにマップモードが続く．

戦闘モードの行動a_5はプレイヤが戦うための行動で，確率$p(C(e_i)|a_5,e_i,\theta^*)$で敵$e_i$への攻撃に成功し，敵$e_i$のHPが$C(e_i)$減少する．プレイヤは確率$1-p(C(e_i)|a_5,e_i,\theta^*)$で敵$e_i$への攻撃に失敗する．また，戦闘モードでは敵もプレイヤに対して攻撃し，確率$p(B(e_i)|e_i,\theta^*)$で敵e_iがプレイヤへの攻撃に成功し，プレイヤのHPが$B(e_i)$減少する．攻撃はプレイヤが常に先攻と仮定する．敵e_iは確率$1-p(B(e_i)|e_i,\theta^*)$でプレイヤへの攻撃に失敗する．行動a_6はプレイヤが敵から逃げるための行動で，確率$p(map|a_6,\theta^*)$でプレイヤは次の期にマップモードに移動し，確率$1-p(map|a_6,\theta^*)$で戦闘モードが続く．行動a_6を選択した場合も，敵は攻撃してくる．よって，プレイヤが逃げることに失敗し，かつ敵が攻撃に成功するとプレイヤはダメージを受ける．

θ^*，$\theta^*\in\Theta$は上記の各確率分布を支配する真のパラメータで未知である．Θは連続パラメータの集合とする．

2.2 MDPとRPGの対応

最初に一般的なMDPの概要について説明する．

MDPは，状態s_i，$s_i\in S$，$S=\{s_1,s_2,\ldots,s_{|S|}\}$（$|S|$は有限），各状態で選択できる行動$a_i$，$a_i\in A$，$A=\{a_1,a_2,\ldots,a_{|A|}\}$（$|A|$は有限），状態$s_i$で行動$a_j$を選択したもとで状態$s_k$へ遷移する遷移確率$p(s_k|s_i,a_j,\xi^*)$（$\xi^*$は遷移確率分布を支配する真のパラメータ），遷移に伴い発生する利得$r(s_i,a_j,s_k)$で構成される．図1に状態数と行動数が2のMDPの例を示す．状態s_1から4本の矢印が出ているが，これらは行動a_1を選択した場合の遷移確率$p(s_1|s_1,a_1,\xi^*)$による状態s_1への遷移と遷移確率$p(s_2|s_1,a_1,\xi^*)$による状態s_2への遷移と，行動a_2を選択した場合の遷移確率$p(s_1|s_1,a_2,\xi^*)$による状態s_1への遷移と遷移確率$p(s_2|s_1,a_2,\xi^*)$による状態s_2への遷移の計4パターンの状態遷移を表現している．

MDPを確率モデルとして採用した問題設定では，行動を選び，状態が遷移し，利得を得るという一連のプロセスを繰り返しながら総利得を最大化することを目的にする場合が多い．プロセスの繰り返し回数が有限の場合には，総利得の期待値（期待総利得）を最大化する最適な決定関数（選択する行動を決める関数）を動的計画法（以下，DPと表記する）によって算出できる．具体的には真のパラメータξ^*既知の場合であれば，式(1)を用いて，t期の状態がs_iという条件下におけるt期以降の期待総利得の最大値$V(s_i,t)$を逐次的に計算できる．

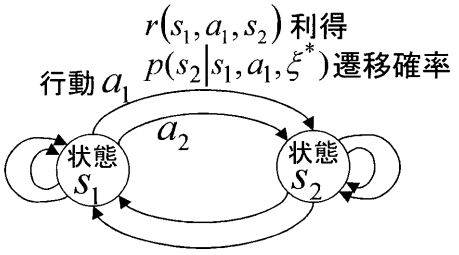


図1　MDPの例

$$V(s_i,t)=\max_{a_j\in A}\sum_{s_k\in S}p(s_k|s_i,a_j,\xi^*)\left(r(s_i,a_j,s_k)+V(s_k,t+1)\right), \quad (1)$$

ただし，$V(s_i,t)$はt期以降の利得の総和の期待値の最大値である．

次に，MDPと前節で説明したRPGの対応について説明する．なお，MDPとRPGの対応についても従来研究[6]と同様である．

x_tはMDPにおけるt期の状態を示す変数で，次式のように構成される．

$$x_t=(x_{t,1},x_{t,2},x_{t,3},x_{t,4}), \quad (2)$$

ただし，$x_{t,1}$はt期におけるプレイヤのHP，$x_{t,2}$はt期におけるプレイヤのマップ上での位置，$x_{t,3}$はt期における敵の種類，$x_{t,4}$はt期における敵のHPを示し，マップモードの場合には敵は存在せず$x_{t,3}=x_{t,4}=0$とする．

$A(x_t)$は状態x_tにおいて選択可能なMDPの行動集合を示す．y_tはMDPにおけるt期に選択した行動を示す変数である．

次にt期の状態x_tで行動y_tを選択したときの状態遷移について説明する．マップモードにおける状態遷移（a_1，a_2，a_3，a_4を選択したとき）を表1（本稿末尾に掲載）に，戦闘モードにおいて行動a_5（戦う）を選択したときの状態遷移を表2（本稿末尾に掲載）に，戦闘モードにおいて行動a_6（逃げる）を選択したときの状態遷移を表3（本稿末尾に掲載）に示す．

$$x_{t+1}=(M_{hp},sm_1,0,0), \quad (3)$$
$$x_{t+1}=(x_{t,1},mv(x_{t,2},y_t),e_i,M(e_i)), \quad (4)$$
$$x_{t+1}=(x_{t,1},mv(x_{t,2},y_t),x_{t,3},x_{t,4}), \quad (5)$$
$$x_{t+1}=(x_{t,1},x_{t,2},x_{t,3},x_{t,4}), \quad (6)$$
$$x_{t+1}=(x_{t,1}-B(x_{t,3}),x_{t,2},x_{t,3},x_{t,4}), \quad (7)$$
$$x_{t+1}=(x_{t,1},x_{t,2},x_{t,3},x_{t,4}-C(x_{t,3})), \quad (8)$$
$$x_{t+1}=(x_{t,1},x_{t,2},0,0), \quad (9)$$
$$x_{t+1}=(x_{t,1}-B(x_{t,3}),x_{t,2},x_{t,3},x_{t,4}-C(x_{t,3})), \quad (10)$$

マップモードの式(4)は敵e_iが出現した場合で，式(5)は敵が出現しなかった場合である．また，ゲームのスタート位置であるsm_1が移動先$mv(x_{t,2},y_t)$の場合には敵は出現せず，プレイヤがスタート位置sm_1に戻り，HPが最大値M_{hp}まで回復する．戦闘モードで行動a_5を選択したときの式(9)は，敵$x_{t,3}$を倒すことに成功したことを示し，この状態遷移に伴い利得$r(x_t,a_5,x_{t+1})=G(x_{t,3})$が得られる．この状態遷移以外のマップモードおよび戦闘モードの状態遷移に伴う利得は0である．戦闘モードでの式(3)は，プレイヤが敵に倒されて，ゲームのスタート位置sm_1からの再開を示す．

状態遷移確率を支配する真のパラメータθ^*のみ未知で，プレイヤや敵の攻撃力$C(e_i)$，$B(e_i)$および敵を倒したときの報酬$G(e_i)$などは全て既知とする．ここまでは，従来研究[6]の真のパラメータが未知の場合と同様である．以下で，本研究における問題設定の追加部分を説明する．

本研究では，当該期間中の総利得を最大化したいT期間の評価期間の前に，当該期間中の利得を考慮せずに練習（学習）に専念するT'期間の練習期間を追加で設定する．本研究の目的は評価期間の総利得を統計的決定理論に基づいてベイズ基準のもとで最大化することである．練習期間の初期状態x'_1と評価期間の初期状態$x_{T'+1}$は，$x'_1=x_{T'+1}=(M_{hp},sm_1,0,0)$で各期の状態は観測可能とする．

ユーザが本格的にプレイする前に行う練習を想定すると，ユーザは単に続けて練習するだけではなく，リセットして初期状態からやり直すこともある．本研究ではこのリセットも練習期間に実施できるようにMDPによるモデルを拡張する．具体的には，練習期間のみ，マップモードおよび戦闘モードで選択できる行動にa_7，a_8，a_9，a_{10}を追加する．練習期間中のt期の状態x'_tにおいて添え字が7以上の行動a_iが選択された場合には，状態x'_tを状態$(M_{hp},sm_1,0,0)$で読み替え，行動a_iをマップモードの行動a_{i-6}で読み替えて解釈する．通常，練習期間の系列として$x'_ty'_tx'_{t+1}$という系列があれば，これは状態x'_tにおいて行動y'_tを選択して，状態x'_{t+1}へ遷移したことを示し，その状態遷移確率は$p(x'_{t+1}|x'_t,y'_t,\theta^*)$である．しかし，$y'_t\in\{a_7,a_8,a_9,a_{10}\}$の場合には，$t$期にリセットの行動選択をしたことによ

り，状態遷移確率 $p\left(x'_{t+1}\middle|\left(M_{hp},sm_1,0,0\right),y''_t,\theta^*\right)$ に従って遷移したことになる．ただし，y''_t は，$y'_t=a_i$ とした場合，マップモードの行動 a_{i-6} である．また，状態遷移確率の具体的な値は表1，表2，表3による．リセットは練習期間の期を最初に戻すことではなく，状態を初期状態に戻すことである．練習期間の状態の初期状態への読み替えおよびマップモードでの行動への読み替えを実施する行動 a_7，a_8，a_9，a_{10} の選択も他の行動同様に1回の行動選択として扱う．この拡張に伴い，従来研究と同様に定義した状態 x_t において選択可能な行動集合 $A\left(x_t\right)$ を修正する．期を示す t も引数に含めて $A\left(x_t,t\right)$ とし，評価期間の場合には $A\left(x_t,t\right)$ は $\{a_1,a_2,a_3,a_4,a_5,a_6\}$ の部分集合であり，練習期間の場合には $A\left(x_t,t\right)$ は $\{a_1,a_2,a_3,a_4,a_5,a_6,a_7,a_8,a_9,a_{10}\}$ の部分集合である．

3．提案アルゴリズム

統計的決定理論[7][8]に基づいて定式化し，評価期間の総利得をベイズ基準のもとで最大化するという点で最適な行動選択の仕方（決定関数）を求めるアルゴリズムを提案する．練習期間の行動選択の仕方は練習の際の戦略，評価期間の行動選択の仕方は本格的にプレイする際の戦略（攻略法）に相当する．

最初に効用関数 $U\left(d\left(\cdot,\cdot,\cdot,\cdot\right),x'^{T'+1}y'^{T'},x^{T+1}y^{T},\theta^*\right)$ を次式で定義する．

$$U\left(d\left(\cdot,\cdot,\cdot,\cdot\right),x'^{T'+1}y'^{T'},x^{T+1}y^{T},\theta^*\right)=\sum_{t=T'+1}^{T'+T}r\left(x_t,y_t,x_{t+1}\right),\quad(11)$$

ただし，$d\left(\cdot,\cdot,\cdot,\cdot\right)$ は当該の期の状態と，当該の期までの練習期間の系列と評価期間の系列と，当該の期（当該の期を示す整数）を受け取って，選択する行動を返す決定関数を示す．$x'^{T'+1}y'^{T'}$ は $x'_1y'_1x'_2y'_2\cdots x'_{T'}y'_{T'}x'_{T'+1}$ という練習期間の系列を示し，$x^{T+1}y^{T}$ は $x_{T'+1}y_{T'+1}x_{T'+2}y_{T'+2}\cdots x_{T'+T}y_{T'+T}x_{T'+T+1}$ という評価期間の系列を示す．効用関数 $U\left(d\left(\cdot,\cdot,\cdot,\cdot\right),x'^{T'+1}y'^{T'},x^{T+1}y^{T},\theta^*\right)$ は真のパラメータ θ^* のもとで，ある決定関数 $d\left(\cdot,\cdot,\cdot,\cdot\right)$ を用いて，練習期間に $x'^{T'+1}y'^{T'}$，評価期間に $x^{T+1}y^{T}$ と遷移した場合の総利得である．

次に期待効用 $EU\left(d\left(\cdot,\cdot,\cdot,\cdot\right),\theta^*\right)$ を次式で定義する．

$$\begin{aligned}EU\left(d\left(\cdot,\cdot,\cdot,\cdot\right),\theta^*\right)&=E\left(\sum_{t=T'+1}^{T'+T}r\left(x_t,y_t,x_{t+1}\right)\right)\\&=\sum_{x'^{T'}y'^{T'}}\sum_{x^{T}y^{T}}\prod_{t=1}^{T'}p\left(x'_{t+1}\middle|x'_t,y'_t,\theta^*\right)p\left(x_{T'+2}\middle|x_{T'+1},y_{T'+1},\theta^*\right)\\&\quad\left(r\left(x_{T'+1},y_{T'+1},x_{T'+2}\right)+p\left(x_{T'+3}\middle|x_{T'+2},y_{T'+2},\theta^*\right)\right.\\&\quad\left(r\left(x_{T'+2},y_{T'+2},x_{T'+3}\right)+\cdots+p\left(x_{T'+T+1}\middle|x_{T'+T},y_{T'+T},\theta^*\right)\right.\\&\quad\left.\left.r\left(x_{T'+T},y_{T'+T},x_{T'+T+1}\right)\cdots\right)\right),\end{aligned}\quad(12)$$

ただし，期待効用 $EU\left(d\left(\cdot,\cdot,\cdot,\cdot\right),\theta^*\right)$ は真のパラメータ θ^* のもとで，ある決定関数 $d\left(\cdot,\cdot,\cdot,\cdot\right)$ を用いた場合の総利得の期待値である．練習期間の系列中にリセットの行動 a_i（$i\geq7$）が含まれる場合には，その期の状態遷移については前述のとおり読み替えて解釈する．リセットの行動に関する読み替えは，これ以降の各種数式についても同様である．

次にベイズ期待効用を定義する．真のパラメータ θ^* が未知なので，パラメータ θ の事前分布 $p(\theta)$ を導入し，事前分布は既知とする．事前分布に対してパラメータ空間で期待効用の期待値をとるベイズ期待効用を次式で定義する．

$$\begin{aligned}&BEU\left(d\left(\cdot,\cdot,\cdot,\cdot\right),p(\theta)\right)\\&\quad=\int_{\theta\in\Theta}p(\theta)EU\left(d\left(\cdot,\cdot,\cdot,\cdot\right),\theta\right)d\theta.\end{aligned}\quad(13)$$

真のパラメータ θ^* 未知の場合には，次式で定義されるベイズ期待効用を最大にする決定関数が最適な決定関数（行動選択の仕方）である．

$$d^*\left(\cdot,\cdot,\cdot,\cdot\right)=\arg\max_{d\left(\cdot,\cdot,\cdot,\cdot\right)}BEU\left(d\left(\cdot,\cdot,\cdot,\cdot\right),p(\theta)\right).\quad(14)$$

式(13)のベイズ期待効用を書き下すと以下のようになる．

$$
\begin{aligned}
&BEU\left(d\left(\cdot,\cdot,\cdot,\cdot\right),p(\theta)\right)=\int_{\theta\in\Theta}p(\theta)EU\left(d\left(\cdot,\cdot,\cdot,\cdot\right),\theta\right)d\theta.\\
&=\sum_{x_2'}\int_{\theta\in\Theta}p(\theta)p\left(x_2'|x_1',y_1',\theta\right)d\theta\\
&\sum_{x_3'}\int_{\theta\in\Theta}p\left(\theta|x_1'y_1'x_2'\right)p\left(x_3'|x_2',y_2',\theta\right)d\theta\\
&\cdots\sum_{x_{T'+1}'}\int_{\theta\in\Theta}p\left(\theta|x'^{T'}y'^{T'-1}\right)p\left(x_{T'+1}'|x_{T'}',y_{T'}',\theta\right)d\theta\\
&\sum_{x_{T'+2}}\int_{\theta\in\Theta}p\left(\theta|x'^{T'+1}y'^{T'}\right)p\left(x_{T'+2}|x_{T'+1},y_{T'+1},\theta\right)d\theta\\
&\left(r\left(x_{T'+1},y_{T'+1},x_{T'+2}\right)+\right.\\
&\sum_{x_{T'+3}}\int_{\theta\in\Theta}p\left(\theta|x'^{T'+1}y'^{T'},x^2y\right)p\left(x_{T'+3}|x_{T'+2},y_{T'+2},\theta\right)d\theta\\
&\left(r\left(x_{T'+2},y_{T'+2},x_{T'+3}\right)+\cdots\right.\\
&\sum_{x_{T'+T+1}}\int_{\theta\in\Theta}p\left(\theta|x'^{T'+1}y'^{T'},x^Ty^{T-1}\right)p\left(x_{T'+T+1}|x_{T'+T},y_{T'+T},\theta\right)d\theta\\
&r\left(x_{T'+T},y_{T'+T},x_{T'+T+1}\right)\cdots\Big),
\end{aligned}
\tag{15}
$$

ただし，y_t' および y_t は決定関数 $d\left(\cdot,\cdot,\cdot,\cdot\right)$ によって決まる t 期の行動である．上記のようにベイズ期待効用は逐次的に事後分布を更新する入れ子構造の形になる．ベイズ期待効用の最大化は，この入れ子構造に DP を適用することによって可能である．

以下で，DP を用いてベイズ期待効用を最大化するという点で最適な決定関数を求めるアルゴリズムを示す．提案アルゴリズムでは，DP で $T'+T$ 期から遡りながら，各期の各状態と1期からその期に至るまでの各遷移系列（練習期間の遷移系列と評価期間の遷移系列）の組に対して行動選択を行う．

$T'+T$ 期目の状態 $x_{T'+T}$（全ての状態の候補）と練習期間の遷移系列 $x'^{T'+1}y'^{T'}$（練習期間の遷移系列の全ての候補）と評価期間の遷移系列 x^Ty^{T-1}（評価期間の遷移系列の全ての候補）の組に対する処理は以下のとおりである．

$$
\begin{aligned}
&d^*\left(x_{T'+T},x'^{T'+1}y'^{T'},x^Ty^{T-1},T'+T\right)=\arg\max_{y_{T'+T}\in A\left(x_{T'+T},T'+T\right)}\sum_{x_{T'+T+1}}\\
&\int_{\theta\in\Theta}p\left(\theta|x'^{T'+1}y'^{T'},x^Ty^{T-1}\right)p\left(x_{T'+T+1}|x_{T'+T},y_{T'+T},\theta\right)d\theta\\
&r\left(x_{T'+T},y_{T'+T},x_{T'+T+1}\right),
\end{aligned}
\tag{16}
$$

$$
\begin{aligned}
&V\left(x_{T'+T},x'^{T'+1}y'^{T'},x^Ty^{T-1},T'+T\right)=\max_{y_{T'+T}\in A\left(x_{T'+T},T'+T\right)}\sum_{x_{T'+T+1}}\\
&\int_{\theta\in\Theta}p\left(\theta|x'^{T'+1}y'^{T'},x^Ty^{T-1}\right)p\left(x_{T'+T+1}|x_{T'+T},y_{T'+T},\theta\right)d\theta\\
&r\left(x_{T'+T},y_{T'+T},x_{T'+T+1}\right),
\end{aligned}
\tag{17}
$$

ただし，$p\left(\theta|x'^{T'+1}y'^{T'},x^Ty^{T-1}\right)$ は1期から $T'+T$ 期の間の練習期間に遷移系列 $x'^{T'+1}y'^{T'}$，評価期間に遷移系列 x^Ty^{T-1} のように遷移した場合の事後分布である．式(16)は最後の1期間の期待利得を最大化する行動を算出し，式(17)は最後の1期間の期待利得の最大値を算出する．

t 期目（$T'<t<T'+T$）の状態 x_t（全ての状態の候補）と練習期間の遷移系列 $x'^{T'+1}y'^{T'}$（練習期間の遷移系列の全ての候補）と評価期間の遷移系列 $x'^{t-T'}y'^{t-T'-1}$（評価期間の遷移系列の全ての候補）の組に対する処理は以下のとおりである．

$$
\begin{aligned}
&d^*\left(x_t,x'^{T'+1}y'^{T'},x'^{t-T'}y'^{t-T'-1},t\right)=\arg\max_{y_t\in A\left(x_t,t\right)}\sum_{x_{t+1}}\\
&\int_{\theta\in\Theta}p\left(\theta|x'^{T'+1}y'^{T'},x'^{t-T'}y'^{t-T'-1}\right)p\left(x_{t+1}|x_t,y_t,\theta\right)d\theta\\
&\left(r\left(x_t,y_t,x_{t+1}\right)+V\left(x_{t+1},x'^{T'+1}y'^{T'},x'^{t-T'+1}y'^{t-T'},t+1\right)\right),
\end{aligned}
\tag{18}
$$

$$
\begin{aligned}
&V\left(x_t,x'^{T'+1}y'^{T'},x'^{t-T'}y'^{t-T'-1},t\right)=\max_{y_t\in A\left(x_t,t\right)}\sum_{x_{t+1}}\\
&\int_{\theta\in\Theta}p\left(\theta|x'^{T'+1}y'^{T'},x'^{t-T'}y'^{t-T'-1}\right)p\left(x_{t+1}|x_t,y_t,\theta\right)d\theta\\
&\left(r\left(x_t,y_t,x_{t+1}\right)+V\left(x_{t+1},x'^{T'+1}y'^{T'},x'^{t-T'+1}y'^{t-T'},t+1\right)\right).
\end{aligned}
\tag{19}
$$

式(18)は t 期以降の期待総利得を最大化するための t 期の行動を算出し，式(19) は t 期以降の期待総利得の最大値を算出する．

T' 期目（練習期間の最後の期）の状態 $x_{T'}'$（全ての状態の候補）と練習期間の遷移系列 $x'^{T'}y'^{T'-1}$（練習期間の遷移系列の全ての候補）の組に対する処理は以下のとおりである．

$$
\begin{aligned}
&d^*\left(x_{T'}',x'^{T'}y'^{T'-1},T'\right)=\arg\max_{y_{T'}'\in A\left(x_{T'}',T'\right)}\sum_{x_{T'+1}'}\\
&\int_{\theta\in\Theta}p\left(\theta|x'^{T'}y'^{T'-1}\right)p\left(x_{T'+1}'|x_{T'}',y_{T'}',\theta\right)d\theta\\
&V\left(x_{T'+1},x'^{T'+1}y'^{T'},x_{T'+1},T'+1\right),
\end{aligned}
\tag{20}
$$

$$
\begin{aligned}
&V\left(x_{T'}',x'^{T'}y'^{T'-1},T'\right)=\max_{y_{T'}'\in A\left(x_{T'}',T'\right)}\sum_{x_{T'+1}'}\\
&\int_{\theta\in\Theta}p\left(\theta|x'^{T'}y'^{T'-1}\right)p\left(x_{T'+1}'|x_{T'}',y_{T'}',\theta\right)d\theta\\
&V\left(x_{T'+1},x'^{T'+1}y'^{T'},x_{T'+1},T'+1\right),
\end{aligned}
\tag{21}
$$

ただし，練習期間中はまだ評価期間の遷移系列はないので，練習期間中の $d^*\left(\cdot,\cdot,\cdot\right)$ や $V\left(\cdot,\cdot,\cdot\right)$ の引数は評価期間の遷移系列を含まない．また，練習期間中の利得は考慮しないので，当該期の利得を考慮しない処理になっている．式(20)は評価期間の期待総利得を最大化するための練習期間の最後の行動を算出し，式(21) は評価期間の期待総利得の最大値を算出する．

t 期目（$1\le t<T'$）の状態 x_t'（全ての状態の候補）と

練習期間の遷移系列$x''y''^{t-1}$（練習期間の遷移系列の全ての候補）の組に対する処理は以下のとおりである.

$$d^*\left(x'_t, x''y''^{t-1}, t\right) = \arg \max_{y'_t \in A(x'_t,t)} \sum_{x'_{t+1}} \int_{\theta\in\Theta} p\left(\theta \middle| x''y''^{t-1}\right) p\left(x'_{t+1} \middle| x'_t, y'_t, \theta\right) d\theta \quad (22)$$
$$V\left(x'_{t+1}, x''^{t+1}y'', t+1\right),$$

$$V\left(x'_t, x''y''^{t-1}, t\right) = \max_{y'_t \in A(x'_t,t)} \sum_{x'_{t+1}} \int_{\theta\in\Theta} p\left(\theta \middle| x''y''^{t-1}\right) p\left(x'_{t+1} \middle| x'_t, y'_t, \theta\right) d\theta \quad (23)$$
$$V\left(x'_{t+1}, x''^{t+1}y'', t+1\right),$$

式(22)は評価期間の期待総利得を最大化するためのt期の練習用の行動を算出し，式(23)は評価期間の期待総利得の最大値を算出する．式(16)から式(23)を用いて，$d\left(x_{T+T'}, x'^{T+1}y'^{T'}, x^T y^{T-1}, T'+T\right)$から$d\left(x'_1, x'_1, 1\right)$まで求めることによって，1期目から$T'+T$期目までの全ての状態と遷移系列の組に対して，ベイズ基準のもとで評価期間の総利得を最大にするという点で最適な行動選択の仕方を算出できる.

式(16)から式(23)には積分計算が含まれており，一般的に積分計算の計算量は大きいが，二項分布（敵の出現以外の確率分布）の事前分布としてベータ分布，多項分布（敵の出現の確率分布）の事前分布としてディリクレ分布を仮定すると，積分計算は四則演算で実施できる[6][9][10].

四則演算の一例として，評価期間中のマップモードのt期の状態x_tにおいて行動y_tを選択したもとで，敵e_iが出現し，戦闘モードの状態x_{t+1}に遷移する場合の$\int_{\theta\in\Theta} p\left(\theta \middle| x'^{T'+1}y'^{T'}, x^{t-T'}y^{t-T'-1}\right) p\left(x_{t+1} \middle| x_t, y_t, \theta\right) d\theta$の計算を以下に示す.

$$\begin{aligned}
&\int_{\theta\in\Theta} p\left(\theta \middle| x'^{T'+1}y'^{T'}, x^{t-T'}y^{t-T'-1}\right) p\left(x_{t+1} \middle| x_t, y_t, \theta\right) d\theta \\
&= \int_{\theta\in\Theta} p\left(\theta \middle| x'^{T'+1}y'^{T'}, x^{t-T'}y^{t-T'-1}\right) p\left(e_i \middle| F\left(mv\left(x_{t,2}, y_t\right)\right), \theta\right) d\theta \\
&= \frac{N\left(F\left(mv\left(x_{t,2}, y_t\right)\right), e_i \middle| x'^{T'+1}y'^{T'}, x^{t-T'}y^{t-T'-1}\right) + \alpha_1}{N\left(F\left(mv\left(x_{t,2}, y_t\right)\right) \middle| x'^{T'+1}y'^{T'}, x^{t-T'}y^{t-T'-1}\right) + \alpha_2},
\end{aligned} \quad (24)$$

ただし，

$$\alpha_1 = \alpha\left(e_i \middle| F\left(mv\left(x_{t,2}, y_t\right)\right)\right), \quad (25)$$

$$\alpha_2 = \sum_{e_j \in E} \alpha\left(e_j \middle| F\left(mv\left(x_{t,2}, y_t\right)\right)\right) + \alpha\left(F\left(mv\left(x_{t,2}, y_t\right)\right) \middle| F\left(mv\left(x_{t,2}, y_t\right)\right)\right), \quad (26)$$

$N\left(F\left(mv\left(x_{t,2}, y_t\right)\right), e_i \middle| x'^{T'+1}y'^{T'}, x^{t-T'}y^{t-T'-1}\right)$は練習期間の遷移系列$x'^{T'+1}y'^{T'}$および評価期間の遷移系列$x^{t-T'}y^{t-T'-1}$中で地形の種類が$F\left(mv\left(x_{t,2}, y_t\right)\right)$の位置で敵$e_i$が出現した回数，$N\left(F\left(mv\left(x_{t,2}, y_t\right)\right) \middle| x'^{T'+1}y'^{T'}, x^{t-T'}y^{t-T'-1}\right)$は練習期間の遷移系列$x'^{T'+1}y'^{T'}$および評価期間の遷移系列$x^{t-T'}y^{t-T'-1}$中で移動先の位置の地形の種類が$F\left(mv\left(x_{t,2}, y_t\right)\right)$だった回数，$\alpha\left(e_i \middle| F\left(mv\left(x_{t,2}, y_t\right)\right)\right)$は$p\left(e_i \middle| F\left(mv\left(x_{t,2}, y_t\right)\right), \theta\right)$に対するディリクレ分布（事前分布）のパラメータ，$\alpha\left(F\left(mv\left(x_{t,2}, y_t\right)\right) \middle| F\left(mv\left(x_{t,2}, y_t\right)\right)\right)$は$1-\sum_{e_i\in E} p\left(e_i \middle| F\left(mv\left(x_{t,2}, y_t\right)\right), \theta\right)$に対するディリクレ分布（事前分布）のパラメータを示す．このように，事前分布としてディリクレ分布やベータ分布を採用することにより，積分計算を四則演算で置き換えることができる.

ディリクレ分布やベータ分布のパラメータの設定が事前分布の設定に相当するが，事前に何も情報が無い場合の設定の仕方についてはベイズ統計学やその応用分野でいろいろな方法が検討されている．本研究のシミュレーションの際には，多くの分野で良好な性質が報告されているジェフリーズの事前分布[6][7][8][9][10]を採用し，具体的には各パラメータを0.5に設定した.

事前分布にディリクレ分布やベータ分布を採用してジェフリーズの事前分布に設定し，式(16)から式(23)で処理することにより，真のパラメータ未知の場合にベイズ基準のもとで評価期間の総利得を最大化する練習期間と評価期間の行動選択の仕方を算出できる.

本研究では，RPGという具体的な適用分野を想定しているが，特定の適用分野を想定せずにMDPという確率モデルそのものを研究対象とした従来研究[11]において，真のパラメータ未知の場合のMDPに関して本研究同様に利得を考慮せずに学習に専念する期間を設けた検討も行われている．しかし，本研究のRPGにおけるリセットの行動に相当するようなものは検討さ

れていない．リセット自体はRPG特有であるが，リセットの考え方はRPG以外にも適用可能である．学習データを取得する系列を1本に限定しているのが従来研究[11]であり，学習データ数は固定のもとで学習データ系列は複数になることを許容（初期状態に戻ることを許容）しているのが本研究である．本研究では表現形式を簡易にするため便宜上は練習期間の遷移系列を1本に見せているが，各種処理の際にはリセットの行動部分では状態や行動を他の状態や行動に読み替えており，練習期間の遷移系列が複数本になることを許容していることと同様である．よって，本研究は従来研究[11]で提案されている未知パラメータをともなうMDPに対する能動的な学習方法を，RPGを例に，より一般化したと解釈できる．

4．提案アルゴリズムの有効性の確認

提案アルゴリズムの有効性を確認するために行ったシミュレーションの結果について報告する．ここでは，2つのシミュレーションパターンについてシミュレーションを行った．図2がパターン1のマップ，図3がパターン2のマップである．その他の設定を表4（本稿末尾に掲載），表5（本稿末尾に掲載），表6（本稿末尾に掲載），表7（本稿末尾に掲載），表8（本稿末尾に掲載），表9（本稿末尾に掲載），表10（本稿末尾に掲載），表11（本稿末尾に掲載），表12（本稿末尾に掲載），表13（本稿末尾に掲載）に示す．なお，パターン1のマップは従来研究[6]で報告されているシミュレーションと同様のマップである．

sm_7	sm_8	sm_9
sm_4	sm_5	sm_6
sm_1	sm_2	sm_3

図2　パターン1のマップ

sm_5		sm_3
sm_4	sm_1	sm_2

図3　パターン2のマップ

上記の設定のもとで，パターン1については$T'=5$期間の練習期間と$T=5$期間の評価期間を合わせた10期間の問題，パターン2については$T'=4$期間の練習期間と$T=4$期間の評価期間を合わせた8期間の問題について，提案アルゴリズムで行動の選択の仕方を算出し，算出した行動の選択の仕方の性質を100回のシミュレーションで確認した．

比較相手1として，練習期間には等確率でランダムに行動を選択し，評価期間は提案アルゴリズムと同様の処理をするアルゴリズムのシミュレーションを行った．比較相手1については，練習期間のランダムな学習を100パターン実施し，各ランダム学習パターンによる事後分布のもとで算出される評価期間の行動の選択の仕方の性質を100回のシミュレーションで確認した．

また，比較相手2として，練習期間に選択可能な行動集合からリセット（a_7，a_8，a_9，a_{10}）を除いて提案アルゴリズムを適用して行動の選択の仕方を算出し，算出した行動の選択の仕方の性質を100回のシミュレーションで確認した．比較相手2はリセットの考え方を考慮していない従来研究[11]の方法をRPGに適用した場合に相当する．

シミュレーション結果を表14（本稿末尾に掲載）に示す．提案アルゴリズムと比較相手2については，評価期間に得られた総利得の100回のシミュレーションでの平均値である．比較相手1については，100パターンあるランダム学習パターンごとに100回のシミュレーション結果があるので，各ランダム学習パターンごとに評価期間に得られた総利得の100回のシミュレーションでの平均値を計算した後に，さらに100パターンのランダム学習パターンで平均値を計算した．

パターン1，パターン2ともに，利得の平均値は練習期間中のリセットを許容した提案アルゴリズムが最も高く，練習期間にランダムに行動選択する比較相手1が最も低い．小規模なシミュレーションでの2例の報告に過ぎないが，このような傾向を確認できた．

また，リセットを許容した提案アルゴリズムにおける特徴的な行動選択について1例を報告する．パターン2について算出したベイズ最適な行動選択の仕方の中で，練習期間の3期の状態$x'_3=(10,sm_3,e_1,2)$において選択すべき行動は$y'_3=a_8$である．これは，練習期間の最初の位置sm_1から右（$y'_1=a_1$），上（$y'_2=a_3$）と移動して敵e_1が出現したもとで，リセットして位置sm_4に移動しようとしている．こうすることによって，4期には上の位置sm_5に移動（$y'_4=a_3$）して位置sm_5について学習しようとしている．パターン2では練習期間が4期間なので，リセットを利用しないと位置sm_3と位置sm_5の両方について学習することができない．パターン2はリセットを許容した提案アルゴリズムにおける特徴的な行動選択を観察するための極端な例であるが，表14でパターン1と2の利得の平均値についてリセットを許容した提案アルゴリズムとリセットを考慮していない従来研究[11]をRPGに適用した場合に相当する比較相手2を比較すると，RPGの能動学習においてリセットを許容することの有効性が確認できる．リセットの有無に限らず，練習期間にランダムに行動選択するよりも，提案アルゴリズムや従来研究[11]に相当する方法によって能動的に行動選択する方が利得の平均値が高くなる傾向にあることも確認できる．

5． 考察と今後の課題

従来からMDPをRPGに適用する研究が行われている．しかし，RPGの各種確率分布を支配する真のパラメータが未知の場合に，得られる報酬を最大化したい評価期間の前に学習に専念する練習期間を設定して能動的に攻略法を学習する方法は検討されていない．

そこで，本研究では総利得（報酬）を最大化したい評価期間の前に学習に専念する練習期間を設け，真のパラメータ未知の場合に評価期間の総利得をベイズ基準のもとで最大化する練習期間および評価期間の行動選択の仕方を算出するアルゴリズムを提案した．

本研究ではMDPをRPGに適用して検討しているが，具体的な適用分野を特定せずにMDPそのものを研究対象とする従来研究[11]において，真のパラメータ未知の場合のMDPに関して本研究同様に利得を考慮せずに学習に専念する期間を設けた検討も行われている．しかし，従来研究[11]では本研究の練習期間におけるリセット（初期状態からの出直し）に相当するものは検討されていない．本研究は，適用分野を特定しないMDPの能動学習にあてはめると，練習期間の学習系列（状態と行動の履歴系列）を1本に限定せずに複数本になることも許容している点で従来研究[11]の一般化に相当する．

本研究の提案アルゴリズムでは，練習期間が長くなった際に，評価期間の行動選択の仕方が真のパラメータ既知の場合の最適な行動選択の仕方に収束することは保証していない．この点を保証するためには，従来研究[12]で提案されているような工夫が必要である．

RPGを対象とした小規模なシミュレーションであるが，リセットを許容する提案アルゴリズムの有効性，リセットの有無に限らず提案アルゴリズムなどによって能動的に学習することの有効性について，シミュレーションをとおして確認した．

本研究のシミュレーションのパターン1では従来研究[6]で報告されているマップと同じ設定にしているが，一般的なPRGと比較すると規模はとても小さい．また，本研究のRPGのモデル化も従来研究[6]と同様で，戦闘モードで選択可能な行動の種類や敵の種類などが少なく，一般的なRPGと比較するとモデルそのものの規模も小さい．シミュレーションやモデルの規模を一般的なRPGに近づけることは理論的には容易であるが，実際には計算量の面で現実的ではない．提案アルゴリズムの計算量は練習期間および評価期間の長さに対する指数オーダである．従来研究[11]で提案されている計算量軽減の工夫を採用すれば，最適性を保持したまま計算量を多項式オーダに軽減できる．しかし，現実のRPGの規模を考慮に入れると多項式オーダでも非現実的である．現実的な計算量のもとで対象となるRPGの規模を大きくするためには，具体的な適用分野を特定せずにMDPそのものの能動学習を研究対象とする従来研究[12][13][14]で検討されているような，短期間のMDPの問題を逐次的に繰り返し解くことによる近似や，状態を部分的にしか観測できない設定などのRPGへの適用を今後検討する必要がある．検討を進める中で，RPGの規模と近似精度と計算量の関係を明確にしていきたい．

上記のRPGの規模の拡大に向けた検討は直近の課題であるが，将来的な課題としては，RPGの開発現場での貢献が挙げられる．適切な規模のRPGにおける行動選択の仕方を算出できれば，算出された行動選択の仕方を使用して，実際のRPGにおけるユーザのプレイ

をシミュレートできる．開発現場では，被験者にプレイしてもらって，マップ上の隠されたアイテムやイベントなどに遭遇する割合や遭遇するまでに要する平均時間などを調査することがある．仮に今後の研究成果のアルゴリズムで算出される行動選択の仕方でユーザのプレイをシミュレートできれば，このような被験者に要するコストの一部を軽減できる．また，従来研究[6]でも指摘されているが，MDP でモデル化されたRPG を被験者にプレイしてもらい，被験者が楽しいと感じた部分を調査することにより，ユーザが楽しいと感じる要素を MDP 上のパラメータのある種の設定パターンとして把握できる可能性がある．ユーザが楽しいと感じる要素を，MDP 上のパラメータの設定によって任意に作り出すことが出来れば，開発コストの軽減や売り上げ増に寄与することが期待される．

参考文献

[1] 金子哲夫 ： マルコフ決定理論入門，槙書店，1973.

[2] Martin, L.P. : Markov Decision Processes, John Wiley & Sons, 1994.

[3] 森村英典，高橋幸雄 ： マルコフ解析，日科技連，1979.

[4] 高木幸一郎，雨宮真人 ： ロールプレイングゲーム（RPG）の戦闘におけるバランス自動調整システム開発のための基礎的考察，情報処理学会研究報告 GI, 2001(28), pp.31-38, 2001.

[5] 高木幸一郎，雨宮真人 ： ロールプレイングゲーム（RPG）のバランスとは何か：分析およびその調整に関する提案，情報処理学会研究報告 GI, 2001(58), pp. 67-74, 2001.

[6] 前田康成，後藤文太朗，升井洋志，桝井文人，鈴木正清 ： マルコフ決定過程のロールプレイングゲームへの適用，情報処理学会論文誌，Vol.53，No.6, pp.1608-1616, 2012.

[7] Berger, J.O. : Statistical Decision Theory and Bayesian Analysis, Springer-Verlag, 1980.

[8] 繁桝算男 ： ベイズ統計入門，東京大学出版会，1985.

[9] Matsushima, T., Hirasawa, S.: A Bayes coding algorithm for Markov models, TECHNICAL REPORT OF IEICE, IT95-1, pp.1-6, 1995.

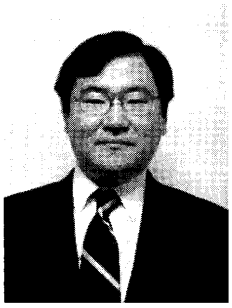
[10] 鈴木譲 ： ベイジアンネットワーク入門，培風館，2009.

[11] 前田康成，浮田善文，松嶋敏泰，平澤茂一 ： 学習期間と制御期間に分割された強化学習問題における最適アルゴリズムの提案，情報処理学会論文誌，Vol.39, No.4, pp.1116-1126, 1998.

[12] 前田康成，松嶋敏泰，平澤茂一 ： 未知パラメータを含むマルコフ決定過程に関する一考察，電子情報通信学会技術研究報告，IT95-17, pp.25-30, 1995.

[13] 木村元，Kaelbling, L.P. ： 部分観測マルコフ決定過程下での強化学習，人工知能学会誌，Vol.12，No.6, pp.822-830, 1997.

[14] 宮崎和光，荒井幸代，小林重信 ： POMDPs 環境下での決定的政策の学習，人工知能学会誌，Vol.14, No.1, pp.148-156, 1999.



前田康成（まえだやすなり）

北見工業大学准教授．

知識情報処理，自然言語処理の研究に従事．電子情報通信学会，情報処理学会等各会員．

表1　マップモードの状態遷移

条件	状態遷移確率	状態x_{t+1}
$mv(x_{t,2}, y_t) = sm_1$	1	式(3)
$mv(x_{t,2}, y_t) \neq sm_1$	$p(e_i \mid F(mv(x_{t,2}, y_t)), \theta^*)$	式(4)
$mv(x_{t,2}, y_t) \neq sm_1$	$1 - \sum_{e_i \in F} p(e_i \mid F(mv(x_{t,2}, y_t)), \theta^*)$	式(5)

表2　戦闘モードで行動a_5が選択されたときの状態遷移

状態遷移確率	状態x_{t+1}
$(1 - p(C(x_{t,3}) \mid a_5, x_{t,3}, \theta^*))(1 - p(B(x_{t,3}) \mid x_{t,3}, \theta^*))$	式(6)
$(1 - p(C(x_{t,3}) \mid a_5, x_{t,3}, \theta^*))p(B(x_{t,3}) \mid x_{t,3}, \theta^*)$	式(7), $x_{t,1} > B(x_{t,3})$の場合 式(3), $x_{t,1} \leq B(x_{t,3})$の場合
$p(C(x_{t,3}) \mid a_5, x_{t,3}, \theta^*)(1 - p(B(x_{t,3}) \mid x_{t,3}, \theta^*))$	式(8), $x_{t,4} > C(x_{t,3})$の場合 式(9), $x_{t,4} \leq C(x_{t,3})$の場合
$p(C(x_{t,3}) \mid a_5, x_{t,3}, \theta^*)p(B(x_{t,3}) \mid x_{t,3}, \theta^*)$	式(9), $x_{t,4} \leq C(x_{t,3})$の場合 式(10), $x_{t,4} > C(x_{t,3})$かつ$x_{t,1} > B(x_{t,3})$の場合 式(3), $x_{t,4} > C(x_{t,3})$かつ$x_{t,1} \leq B(x_{t,3})$の場合

表3　戦闘モードで行動a_6が選択されたときの状態遷移

状態遷移確率	状態x_{t+1}
$p(map \mid a_6, \theta^*)$	式(9)
$(1 - p(map \mid a_6, \theta^*))(1 - p(B(x_{t,3}) \mid x_{t,3}, \theta^*))$	式(6)
$(1 - p(map \mid a_6, \theta^*))p(B(x_{t,3}) \mid x_{t,3}, \theta^*)$	式(7), $x_{t,1} > B(x_{t,3})$の場合 式(3), $x_{t,1} \leq B(x_{t,3})$の場合

表4　パターン1の地形の設定

$F(sm_1)$	$F(sm_2)$	$F(sm_3)$	$F(sm_4)$	$F(sm_5)$	$F(sm_6)$	$F(sm_7)$	$F(sm_8)$	$F(sm_9)$
f_1	f_2	f_3	f_2	f_2	f_3	f_3	f_3	f_3

表5　パターン1の確率の設定（その1）

$p(e_1 \mid f_2, \theta^*)$	$p(e_2 \mid f_2, \theta^*)$	$p(e_1 \mid f_3, \theta^*)$	$p(e_2 \mid f_3, \theta^*)$
0.4	0.0	0.0	0.9

表6　パターン1の確率の設定（その2）

$p(C(e_1) \mid a_5, e_1, \theta^*)$	$p(C(e_2) \mid a_5, e_2, \theta^*)$	$p(B(e_1) \mid e_1, \theta^*)$	$p(B(e_2) \mid e_2, \theta^*)$	$p(map \mid a_6, \theta^*)$
0.9	0.8	0.5	0.8	0.8

表7　パターン1のその他の設定（その1）

M_{hp}	E	$M(e_1)$	$M(e_2)$	$G(e_1)$	$G(e_2)$
10	$\{e_1, e_2\}$	2	4	10	100

表8　パターン1のその他の設定（その2）

$C(e_1)$	$C(e_2)$	$B(e_1)$	$B(e_2)$
4	2	1	3

表9　パターン2の地形の設定

$F(sm_1)$	$F(sm_2)$	$F(sm_3)$	$F(sm_4)$	$F(sm_5)$
f_1	f_2	f_3	f_2	f_4

表10　パターン2の確率の設定（その1）

$p(e_1 \mid f_2, \theta^*)$	$p(e_2 \mid f_2, \theta^*)$	$p(e_1 \mid f_3, \theta^*)$	$p(e_2 \mid f_3, \theta^*)$	$p(e_1 \mid f_4, \theta^*)$	$p(e_2 \mid f_4, \theta^*)$
0.0	0.0	0.5	0.0	0.0	0.5

表11　パターン2の確率の設定（その2）

$p(C(e_1) \mid a_5, e_1, \theta^*)$	$p(C(e_2) \mid a_5, e_2, \theta^*)$	$p(B(e_1) \mid e_1, \theta^*)$	$p(B(e_2) \mid e_2, \theta^*)$	$p(map \mid a_6, \theta^*)$
0.9	0.9	0.6	0.6	0.6

表12　パターン2のその他の設定（その1）

M_{hp}	E	$M(e_1)$	$M(e_2)$	$G(e_1)$	$G(e_2)$
10	$\{e_1, e_2\}$	2	4	10	100

表13　パターン2のその他の設定（その2）

$C(e_1)$	$C(e_2)$	$B(e_1)$	$B(e_2)$
3	3	2	4

表14　総利得の平均

	提案アルゴリズム	比較相手1	比較相手2
パターン1	81.4	68.6	72.3
パターン2	52.0	34.0	41.0