AI化建築へ：マルチタスク強化学習による住居屋根制御

＊田中 文英，山村 雅幸
東京工業大学 大学院総合理工学研究科 知能システム科学専攻
E-mail: boom@es.dis.titech.ac.jp（連絡先：田中）

1　研究の背景と構想概要

本研究の目的は，筆者の提案に基づく人工知能（AI）分野における強化学習手法を，具体的な応用として将来の建築設計に適用することにある。

従来，強化学習分野における研究は理論面が中心であり，現実的な場面に適用されたものは比較的少ない。この理由の一つは，これまでの強化学習手法は学習に莫大な試行回数を要すること，そして二つ目は，その性能評価を現実的な有限学習回数（時間）にて行う手段が存在しなかったことによる。本研究では，これらの問題点に解決策を与えることをまず目指す。

一方で，応用対象として考えている建築，特に最も身近な対象である住居に関する現状を述べる。昨今における多くの著書等を持ち出すまでもなく，現在この分野は大きな転機を迎えていると言われている。20世紀後半の住居は，その室内気候形成に関して，それよりも昔に行われていたように自然換気等を活かす（パッシブ）よりも空調等による強制的な環境制御（アクティブ）を中心に据えていた。しかし，今世紀に入り，持続可能な社会や自然との共生といったテーマが際立ってくると共に，住居の室内気候形成はパッシブとアクティブの概念を融合し，これまで以上に一層の快適性とエネルギー効率とを両立すべきとの意見が強まってきた。ここで元来，エージェント（意志決定に基づく学習対象）と環境（エージェントから見た外界）との間の適応現象を幅広く扱えるように設計された強化学習枠組は，上に述べた現実問題適用に際してのギャップを埋めることができさえすれば，住居のコンポーネントをエージェントとみなしその住環境への適応という場面に有効性を発揮できるものと思われる。例えば，屋根の角度や厚みというものは室内気候形成を考える上で重要なパラメータであることが知られている。日本古来の民家には，太陽の位置が高い夏にはその光を遮り暑さを和らげ，逆に位置が低い冬にはそれがうまく家の奥まで射し込み空気を暖めるよう高度に調整されたものがある。屋根に関しては近年他にも色々な知見が出てきており，これら複雑な調整を自律的に行うような機能をシステムとして実現できれば，様々な場所の様々な住環境に対し個別に適応した住まい，ひいては無駄を省きエネルギー効率の良い住居実現に繋がることが期待できる。本研究の二つ目の目標である応用は，以上のような背景を持つ。

本研究では，将来的な実アプリケーション（ハード）作成を見据え，その前段階としてソフトウェアによるシステム開発を行う。計画は上で記した二目標を軸に展開される。まず，以下に説明するポイントを踏まえた強化学習手法の開発を行う。同時に，対象を住居屋根制御とした上で仮想環境シミュレータを構築し，これを用いて開発した手法の評価，そして最終的にアプリ実現性の検証を行う。併せて各方面からの評価・フィードバックを得ることにより，ハード作成に向けた重要な足掛かりと成ることを目指す。

先に，対象とする問題について説明する。上に述べた背景を基として，住居の屋根を制御することにより住む人間にとって快適な気候（室温・湿度など）を形成することを目指す。制御する具体的部位に関しては，近年の建築環境工学等の分野で得られた知見を元に様々なもの（現時点で最も有望な部位：屋根角度・厚み・屋根裏の容積など）を試す。ここではハード的な実現性にも十分気を配る。目標とする理想気候データは同分野で示されているものを用い，これら部位パラメータを試行錯誤的に制御することによりどの程度理想値に近づけたかを単位時間ごと測定し，評価（報酬）に用いる。ここで，住居立地条件や建物内状況が異なれば制御解は千差万別である。対象とする現象は要因の複雑さのみならず，評価の遅れ（制御入力がすぐ結果に影響するとは限らない）等もあって，理想とする目標データがあったとしても解析的手法によりそれを求めようとするのは事実上困難である。このような性質を持つ本問題に対し，有効と思われる手法のポイントを以下に述べる。

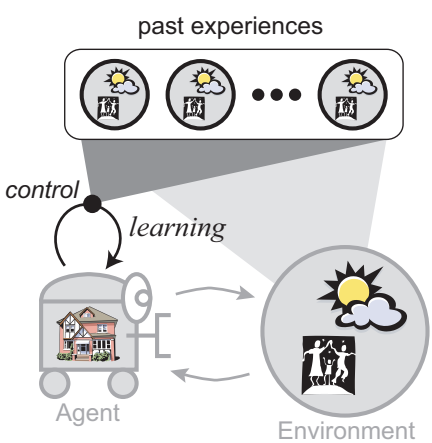
1.1　インタラクションを幅広く扱える枠組に基づくこと

適応能力を考える上で，インタラクションは本質的な概念である。我々生物は，変動していく環境下において，知覚（観測）と行動とを繰り返すことにより様々な情報を収集しながら日々生きている。これこそ適応の意味する所であり，その基本単位がインタラクションである。ここでは，このインタラクションを幅広く扱える汎用枠組を基にしていることが望ましい。強化学習はその代表であり，試行錯誤を通じての行動学習を幅広くモデル化できる。例えば，マウスロボットの様な原始的センサ・アクチュエータを備えたものに，地図を与えずゴールに辿り着いた時の報酬情報のみから，自律的に迷路の最短パスを学習させることができる。ここでは，エージェントであるロボットが迷路問題を解くという過程を通して，環境への適応現象を論じている訳である。強化学習は非常に汎用性の高い枠組であり，更に遅れのある評価（報酬）や環境の不確実性を幅広く扱えることが知られている。これは本研究で対象とする問題に向いている為，ここでも利用することにする。

1.2　適切なタスク設定を行うこと：マルチタスク強化学習へ

学習には例題が必要である。強化学習は一般的な教師あり学習とは異なるが，エージェントの入出力組としてそれを捉えれば，同様のことが言える。例題の数は学習効率に大きく影響する。この数はタスク設定によって決まるので，これをうまく定める必要がある。本研究の対象問題では，タスクを決める主要素に時間があり，これをなるべく小さくすることが肝心である。そこで最小単位として一日を選ぶ。これは周期的に最も基本的な単位であり，また理想データもこの単位で与えられることが多いからである。本研究では一日の現象を元にひとつのタスク（エージェント行動・環境レスポンス［評価報酬］組の集合）を設定し，その繋がり（マルチタスク）として年月に渡る適応性を論じていくことにする。

この設定下における有効な強化学習として，本研究では以下の方法を用いることにする。ここでは一日を基本単位として日々独立した学習を行う訳であるが，毎日の学習結果を丸々捨ててしまわずに，一部を維持していくことを考える。日々の学習結果は，行動指標として用いるある評価値テーブルとして得られるのであるが，ここでその各要素の統計量（平均・分散）をタスク終了後に計算・保存することにする。すると，全タスク間にある関わりが存在すれば，その統計量を用いて次の日以降の学習をより効率良く行えることがわかっている。しかし，常識的に考えて，タスク間の関わりが余りにも小さい場合は，タスク集団から抽出した統計量を用いることの意味も薄れてくるはずである。実際，提案手法に関しても，その効果には環境を表現する状態数とエージェントの行動数から決まる上限があることが分かっており，万が一タスク集団の性質がこれをも上回る程ランダム性の強いものであった場合の対処も考えておく必要がある。本研究の実験では，最初はカリフォルニア地域のような一年を通して比較的変動の小さな気候想定から始め，徐々にその変動度合を上げていく。ここで追従困難となるレベルを実験により明確にする。もしも日本の気候のように，一年を通して比較的変動が大きいもののその中で四季として分類ができるような場合は，四季の移り変わり時期の始めに過去同時期のデータ（他の住居で得られたものでも構わないのがポイント）をファーストチョイスとして採用し，そこから徐々に自らの現状に適応するよう利用する方法が考えられる。現在，各家庭において急速にネットワークインフラが進みつつあるが，これを利用して他の住居で得られた学習結果を利用するというアイディアは，より学習挙動に柔軟性を与え得るものとして期待が持てる。



本研究における最初のゴールは，以上のポイントを踏まえた強化学習手法を提案し，既存ベンチマーク問題や解析法により測定可能な基本性能評価（ごくシンプルな問題下での学習収束性等）を行うことである。

次に，数学的に無限学習回数後の収束性を拠り所としている強化学習研究は，これまでは理論面での議論が中心であり，有限学習回数での性能評価が為されることは殆どなかった。しかし，現実的な場面ではあるレベルで学習を打ち切らねばならない。その際にもし，N 学習回数後に（無限回後における）最適値の

x %までの性能が保証される，というような指標があれば，この打ち切りに大きく役立つものと思われる。二番目のゴールは，この指標を求めることである。ここで問題の大きさは環境を記述する状態変数とエージェントの行動変数の要素数積で測り，計算量は多数の実験からオーダーレベルで測定し，これらデータ組から指標を作成する。

そして最後のゴールは，提案した手法や評価方法を元に，仮想環境シミュレータ（VRML 等を用い視覚性に優れたもの）上で，モデルの有意性に十分気を配りながら制御部位や条件を色々と試し，蓄積した知見を元に本物アプリケーションの実現性を議論することである。

2　マルチタスク強化学習：イントロダクション

動物の学習能力を主に計算機上で実現することを目指す機械学習は，人工知能（AI）の中でも最も主要な分野の一つとして，盛んに研究が行われてきている。より具体的に，機械学習は様々なアプローチに分類することが可能であるが，概してこれまでに行われてきた研究は，単一のタスクを如何にして学習するか？といった立場で為されてきたものが多い。それに対して近年，複数のタスクに跨った学習を考えようとする立場が注目を浴び始めてきている [10, 11]。ヒューマノイド型ロボットを始め，周辺分野では高度なハードウェアの開発が劇的に進みつつある。これに応じて，ソフトウェア側でもより人間に近い高度な性能を要求されることは，自然な動向であると言えよう。例えば，我々の生活を振り返ってみると，日々似通った問題解決（学習）を繰り返し行っている場面が多々あることに気づく。ここで，その場面に二度目以降遭遇した際には，過去の学習経験を用いて現在面する問題（タスク）の学習をより効率の良いものとすることが普通である。これが複数タスクに跨った学習であり，単一タスク学習の研究では扱われてこなかった視点である。このような，複数タスクを扱う機械学習の研究は，その意義や重要性といった要素は理解できるものの，未だ系統立った研究が行われにくい状況にある。我々はその理由として，対象とする問題の定式化が明確に為されていない点を重視した。そこで，我々は，強化学習枠組の元で複数タスクの学習を定式化し，具体的な実装手法の一例を提案することにする。

強化学習とは，試行錯誤を通じて外界に適応するエージェントの汎用的な学習枠組である [2, 9]。エージェントの例としては，上に挙げたロボット等がその代表例であり，知的な行動獲得における中核技術の一つとして期待されている。基本的にここでは，動物の躾のように多数回のトライアルと評価（報酬）を手掛かりとして徐々に所望の行動戦略を学習させていくということを行う。多数回のトライアル（試行錯誤的行動）は外界の情報を得る上で本質的な操作であるが，同時にこの量が余りにも多いと実問題適用に際する足枷にも成り得る。本研究では，この強化学習にマルチタスク性を与え，過去タスク学習経験を利用することにより新たな学習に要する試行回数の削減を図る。これは，実応用に向けて有用性の面でも意義がある。

3　強化学習の基本概念

強化学習は，試行錯誤に基づく行動学習を幅広く扱うことのできる学習枠組である。元々は「パブロフの犬」の様な現象を説明する為に始められた研究分野であるが，近年の計算機性能の大幅な向上に伴い，工学的な方法論まで盛んに研究が行われてきている。特に，1990 年代に入り，ダイナミック・プログラミングやモンテカルロ法といった周辺分野と融合することにより，理論面からの理解が大きく進んだ。本章では，その基本概念について説明する。

外界と離散時間スケールでインタラクションし続ける学習対象を考え，これをエージェントと呼ぶ。エージェントは各時間ステップ毎に外界の状態 $s_t \in \mathcal{S}$ を観測し，行動 $a_t \in \mathcal{A}$ を選択・実行する。これを受けて外界は状態遷移し，次状態 s_{t+1} と報酬値 r_{t+1} とをエージェントに返す。通常，外界は 4 項組 $< \mathcal{S}, \mathcal{A}, \mathcal{P}, \mathcal{R} >$ により定義される有限マルコフ決定過程（以下 MDP）で記述される。ここで $\mathcal{P}$ は状態遷移確率 $P^a_{ss'} = \Pr\{s_{t+1} = s' \mid s_t = s, a_t = a\}$，$\mathcal{R}$ は期待報酬 $R^a_{ss'} = E\{r_{t+1} \mid s_t = s, a_t = a, s_{t+1} = s'\}$ のそれぞれ集合である。エージェントの目標は，各状態において下式で定義される期待獲得報酬値（利得）を最大化する，政策 $\pi(s, a)$ を学習することである。これは一般に $\mathcal{S} \times \mathcal{A}$ 上の確率密度関数で与えられる。

$$V^\pi(s) = E\left\{ r_t + \gamma r_{t+1} + \gamma^2 r_{t+2} \cdots \middle| s_t = s, \pi \right\} \tag{1}$$

ここで γ は割引率である。$V^\pi(s)$ は政策 π の元で状態 s の評価値（value）を表していて，この V^π を value function と呼ぶ。そして更に，これに基づき最適 value function V^* を下式で定義する。

$$V^*(s) = \max_\pi V^\pi(s) \tag{2}$$

このようにして，未来の報酬まで見積もった関数を学習の指標として用いることにより，報酬の遅れといった通常の教師あり学習では対処が困難な問題を扱うことができるようになる。MDP で記述可能な不確実性を含んだ意志決定問題を扱えること，そしてこの遅れのある報酬への対応力が強化学習の特徴である。

強化学習では，ダイナミック・プログラミングの手法と異なり，予め外界の情報（上で言う $\mathcal{P}$ や $\mathcal{R}$）は未知である。その代わりに，試行錯誤を通じて value function を推定していく。推定法には，Sutton の TD 学習法 [7] をベースとして様々なものが提案されており，その代表的アルゴリズムが Q-learning[12] である。Q-learning では，評価値として，全状態 s と全行動 a の組に対しそれぞれ $Q(s,a)$ という値を考え，上と同様にこの下で利得を考える。行動についても加味する以外は上の $V(s)$ と同様であるが，用語としてはこれを区別し，$Q(s,a)$ を action value，$V(s)$ を state value と呼ぶ。これらを総称して value と呼ぶことも多い。この Q 値を導入することにより，ある条件の下で政策 π によらず，十分な学習の後に最適 value function Q^* を得られることが証明されている [12]。ここで，ある状態において常に最大 Q 値をとる行動を選択する政策が，最適政策 π^* である。Q-learning は，上述した政策によらない点などから広く用いられている。

4　マルチタスク強化学習の定式化

例として以下の場面を想像してみる。あるロボット（エージェント）が，ユーザから毎日ひとつずつタスクを与えられるものとする。各々タスクについては，ロボットの入出力ハードウェアを理解しているユーザが，そのハード資源内でその日中に対処（学習）可能な程度のものを用意するものとする。そしてロボットはその日の内に，与えられたタスクを解く。次の日，再び新たなタスクがユーザから提示される。ロボットは再びその日の内にタスクを解く…。このプロセスがロボットの一生期間（lifetime）続くものとする。さて，この設定下でロボットは，過去のタスク学習経験を何らかの形で維持し続け，現在面しているタスクの学習に活かすことができるものとする。このような強化学習は，どのようにしたら表現可能であろうか？

知的なエージェントを考える上で，こうした問題，つまり複数のタスクに跨る学習を目指すのは自然な発想である。強化学習を含めて従来多くの機械学習研究は，単一のタスクを扱うものがほとんどであったが，近年ではそれを超え，複数タスクを扱う機械学習の重要性が提案され始めている [10, 11]。文献 [11] は，このテーマに関する殆ど唯一のサーベイである。その中でも紹介されている文献 [6] は，MDP で記述された三つのシンプルな問題を別個に Q-learning で解いて学習結果をモジュール化して保存し，これを切り替えて用いることによりそれらが結合した問題を解くということを行っている。これらの手法は，思想的には我々のイメージする所に近いものの，タスク間の関連性が明確に議論されておらず，拡張性が低い。我々は，マルチタスク性というテーマに関してより広範囲に適用が可能で系統立った研究を行っていく為には，定式化の部分からより一般的に問題を考えていく必要があると考えた。そこで，ここでは前章にて説明した強化学習の基本枠組の上に，マルチタスク学習（以下 MTRL）の一般的な定式化を行う。引き続き次章において，ここで有効な具体的手法の一例に触れることにする。

4.1　準備１：タスクと環境の定義

本研究では，ある一つのタスクを，MDP とその継続時間 τ により定義する。つまり，一つのタスクとは五項組 $< \mathcal{S}, \mathcal{A}, \mathcal{P}, \mathcal{R}, \tau >$ を定めることにより一意に決まるものとする。

続いて，このタスク上の分布を考えて，これを広く環境と呼ぶことにする。言い換えると，タスクとは環境からの独立なサンプルであり，環境とはタスクの母集団（タスクインスタンスに対するクラス）であるとする。これらの定義は通常の強化学習の場合と異なることがあるので注意を要する。

以上は最も一般的な定義であり，ここからより具体的な派生系を色々と考えることができる。そこで，以下本論文内においては，MDP ダイナミクスを構成する $\mathcal{P}$ と $\mathcal{R}$ の全要素に関しそれぞれ正規分布を仮定し，その集合 $\mathcal{DP}$ と $\mathcal{DR}$ を含む 4 項組 $< \mathcal{S}, \mathcal{A}, \mathcal{DP}, \mathcal{DR} >$ により，環境を定義するものとする。

4.2 準備２：生涯獲得報酬の導入

前章にて説明した強化学習の基本枠組において，エージェントはタスク内で利得と呼ばれる評価指標の最大化を目指すことを述べた。我々のテーマである複数タスク設定下では，これに加え，エージェントの lifetime を通じての総獲得報酬を考える必要がある。これをフォーマルに表すと，

$$TR = \sum_{i=1}^{N} \int_{t_{i-1}}^{t_{i-1}+\tau_i} r_t \, dt \quad \left(t_i = \sum_{j=1}^{i} \tau_j, \; t_0 = 0 \right) \tag{3}$$

ここで N は総タスク数を表し，この式で与えられる TR を生涯獲得報酬と呼ぶ。さて，実はこの式は，時間スケールの概念に関して暗に重要な問題を提示している。前章で説明した基本枠組において利得は無限時間下にて定義されているのに対し，上の複数タスクに跨る時間（lifetime）概念は有限性を仮定している。つまりここで浮かび上がってきた問題とは，有限時間下での強化学習とその性能評価に対する必要性である。これは大きなテーマであると言えるが，本論文では解を与えず，タスク内の時間スケールが各タスクの継続時間 τ と比較して十分細かいものと仮定して以下話を進めることにする。

4.3 問題の定式化

以上の議論を元に，MTRL を定式化する。ある一生期間において N 個のタスクを順番に与えられる強化学習エージェント（図 1 参照）を考える。ここでタスクとは $< \mathcal{S}, \mathcal{A}, \mathcal{P}, \mathcal{R}, \tau >$ により定義され，タスク上の分布を定めた環境クラスから，τ 時間毎独立にサンプルされエージェントに提示されるものとする。本エージェントの目的は，各タスク内にて局所的に利得を最大化することに加えて，生涯獲得報酬 (3) 式をも最大化することである。この目的達成の為に，エージェントは過去のタスク学習経験を何らかの形で維持し続け，現在面しているタスク学習にそれを活かそうとする。

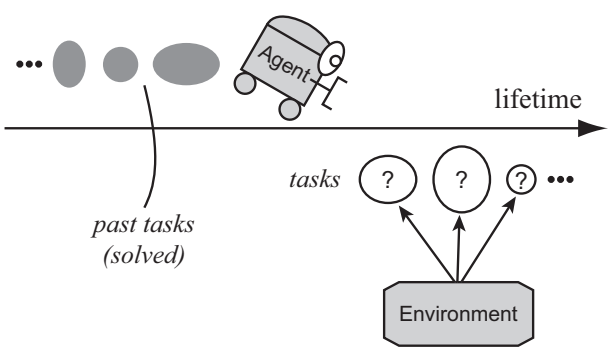


図 1: マルチタスク強化学習を行うエージェント

5 MTRL エージェントの実装方法

4.3 節にて記した様に，MTRL におけるエージェントの目標は，生涯獲得報酬 (3) 式の最大化である。ここで (3) 式の値を直接計算することは一般に困難であるが，本論文の問題設定では各タスクは独立してサンプルされる為，この目標は，全てのタスク内においてそれ以前のタスク学習で得られた経験を何らかの形で用いることにより τ 期間における総獲得報酬の最大化を目指すことと見なすことが可能である。この目標に対するアプローチとして，以下まずは維持する学習経験の形態について述べ，続いてそれをどのようにして現在のタスク学習に活かしていくかを述べる。

5.1 タスクを通じて維持する学習経験

強化学習エージェントにおいて，その学習結果として得られるものの中で最も一般的なデータは，推定した value テーブルである。本論文にて扱う MTRL では，各タスクを記述する MDP ダイナミクスの間に，

4.1 節で説明した環境により定義される関わりが存在する。その為，各タスク学習ごと最終的に得られる value テーブルに関しても，何らかの関わりが存在するものと思われる。そこで，この量を value に関する統計量から抽出し，これを学習経験（知識）として複数のタスクを通じ維持していくことにする。

統計量として最も一般的な量は，平均値と標準偏差（分散値）である。そこで，例えば value として action value（Q 値）を考えるならば，全ての状態・行動組について

$$\overline{Q}(s,a) = \frac{1}{n}\sum_{i=1}^{n} Q(s,a)_i \tag{4}$$

$$\sigma_{Q(s,a)} = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(Q(s,a)_i - \overline{Q}(s,a))^2} \tag{5}$$

（i はタスク番号，n は過去に解いた総タスク数）を各タスク終了後に計算・更新し，これを維持していくことにする。さて，これらの統計量の意味についてであるが，これは総じて環境クラスの特徴量を意味している。そしてまた，次に新たなタスクが提示され，$\overline{Q}(s,a)$ 値（テーブル）をそのタスク学習における初期値として用いた場合，その値の信頼度が $\sigma_{Q(s,a)}$ で与えられているとみることもできる。

5.2 Value 統計量を利用したモデルベース強化学習

本章冒頭部に述べた目標を達成する為に，エージェントはなるべく少ない行動ステップ数でより良い value function を推定する必要がある。一般に強化学習では，実ロボットを用いるなどして一回の行動が高価な場合，モデルベースと呼ばれるアプローチを採用する場合が多い [8, 9]。これは，通常のオンライン的強化学習（real experience）に加えて，同時にそこで得られた経験からタスク毎外界のモデルをエージェント内に構築し，そこでの仮想的強化学習（simulated experience）をも行って，単一の value function 推定を加速しようとする試みである。最も基本的なモデルベース強化学習（Dyna-Q アルゴリズム [8]）の具体的手順を図 2 に示す。ここでは，通常の Q-learning に加え，モデル（単純な値テーブルにより表現）の更新，そして X 回のモデル内 Q-learning をシリアルに行っている。

```
Initialize Q(s,a) and Model(s,a) for all s,a
Do forever:
  Normal Q-learning
  Model(s,a) ← s',r
  Repeat X times:
    s ←random previously observed state
    a ←random action previously taken in s
    s',r ← Model(s,a)
    Q(s,a) ← Q(s,a) + α [r + γ max_a' Q(s',a') − Q(s,a)]
```

図 2: Dyna-Q アルゴリズム

ここでは，このモデルベース強化学習を出発点として，以下に説明する二方法で過去経験（それまでのタスク学習から得られた value 統計量）を組み入れた手法を提案する。

〔Value 平均値の利用〕強化学習において，学習を始めるに際し value の初期値をどのように初期化するかは，その後の学習性能に大きく影響するにも関わらず，これまで余り知見は得られていない。結果として通常，全ての値を 0.0 と設定する場合が多い。それに対し MTRL の問題設定下では，常に初期値を全タスクまでに得られた value テーブルの平均値と定めることが可能である。当然の事ながら，この効果はタスク間を通じて MDP ダイナミクスの変動が少ないほど高いことが推察される。

$$Q_0(s,a) = \overline{Q}(s,a),\ for\ all\ s,a \tag{6}$$

〔Value 標準偏差の利用〕まず，下式で定義される $I(s,a)$ 値を準備する。これは $Q(s,a)$ 値の信頼度幅（interval）を表す指標であり，値が小さい程，対応する Q 値が真値に近い確率が高いことを意味している。κ_1 はその初期値を決める際のパラメータである。I 値は，該当する行動が選択される（i は更新番号）度に κ_2 ずつ減少させる。これは，Q-learning の value 推定性能が単調上昇する点から導入した定性的なパラメータである。本論文では κ_1, κ_2 共に実験的にその値を定めることにする。

$$I_0(s,a) = \kappa_1 \cdot \sigma_{Q(s,a)} \tag{7}$$

$$I_i(s,a) = I_{i-1}(s,a) - \kappa_2 \tag{8}$$

続いて，この $I(s,a)$ を用いて，全ての Q 値に関し以下の二つの量を定義する。

$$Q^{+\sigma}(s,a) = Q(s,a) + I(s,a) \tag{9}$$

$$Q^{-\sigma}(s,a) = Q(s,a) - I(s,a) \tag{10}$$

さて，モデルベース強化学習で重要なのは，simulated experience における value の更新を，なるべく効率の良いものとすることである。エージェントが一回の実行動を取る間に行うモデル内仮想行動の数は一定値 X として定められている為，言い換えると一仮想行動当りの value 更新（バックアップ）を，なるべく正しく，なるべく多く 行う必要がある。ここで後者（なるべく多く）のポイントに関しては，次式で定義される優先度に応じて更新の順番付けを行う方法が提案されている [3, 4]。

$$p \leftarrow \left| r + \gamma \max_{a'} Q(s',a') - Q(s,a) \right| \tag{11}$$

前者（なるべく正しく）のポイントに対してこれまでに有効な手法は提案されていないが，MTRL では value 標準偏差の利用を考えることができる。ここでは 5.1 節にて述べた通り，計算した value 平均値を初期値として用いた際に value 標準偏差をその信頼度を表すものと見ることができる。通常の Q 値更新式では，次状態における最大 Q 値を参照しながら更新を行うが，この推定は多くの試行回数を通じて徐々に行われるため，当然の事ながら学習の途中ではその参照する Q 値が正しい値から大きく外れている場合も多々ありうる。ここで，上で述べた信頼度（value 標準偏差）を用いてそのミスバックアップ（間違った Q 値更新）を減らすことができれば，試行回数の削減に繋がることが期待できる。このアイディアは，(11) 式の p 値計算において，$Q(s',a')$ の部分を $Q^{-\sigma}(s',a')$ により置き換えることで実現できる。そして計算された p 値を用いて X 回のモデル内 Q-learning を行う。ここでは，エージェントの実行動で遭遇した場面全てで p 値を計算し，その高い順に (s,a) 組を格納したキューを用意する。毎回のモデル内 Q-learning フェーズでは，キューの上から順に X 個を取り出し，それら (s,a) に関して仮想行動を行い，Q 値テーブルを更新する。（この操作は文献 [3, 4] と同様）

ここでは更に，上で value 標準偏差の大きい部分は小さい部分よりも新たな探索が必要であると考えて，通常の Q-learning（図 2 の上から三行目）における探索戦略に I 値を加味した行動選択を導入する。通常の Q-learning では，その行動選択に際し，下式で表される確率密度関数を用いたルーレット選択により行われることが多い [9, 12]。（T は温度パラメータ）

$$\pi(s,a) = \frac{e^{Q(s,a)/T}}{\sum_{all\ a} e^{Q(s,a)/T}} \tag{12}$$

ここで $Q(s,a)$ 値の代わりに $Q^{+\sigma}(s,a)$ 値を用いることにより，偏差の度合を反映させた行動選択が可能となる。Q 値の推定幅を用いるアイディアは，Kaelbling の Interval Estimation Method[1] と同様であるが，この手法が単一タスク学習内での多数行動を通した幅推定に基づくのに対し，我々の用いる I 値は複数タスクを通じての value 統計量から得られたものであるという違いがある。因みに他の関連研究として，モデル内で信頼性の低い部分を実問題では反って危険部位と見なし逆に探索を抑えるというアイディアも提案されている [5]。

参考文献

[1] Kaelbling, L.P.: "Learning in Embedded Systems", MIT Press (1993)

[2] Kaelbling, L.P., Littman, M.L. and Moore, A.W.: "Reinforcement Learning: A Survey", *Journal of Artificial Intelligence Research*, Vol.4 pp.237–285 (1996)

[3] Moore, A.W. and Atkeson, C.G.: "Prioritized Sweeping: Reinforcement Learning with Less Data and Less Real Time", *Machine Learning*, Vol.13 pp.103–130 (1993)

[4] Peng, J. and Williams, R.J.: "Efficient Learning and Planning Within the Dyna Framework", *Adaptive Behavior*, Vol.1 No.4 pp.437–454 (1993)

[5] Schneider, J.G.: "Exploiting Model Uncertainty Estimates for Safe Dynamic Control Learning", *Advances in Neural Information Processing Systems 9*, pp.1047–1053 (1997)

[6] Singh, S.P.: "Transfer of Learning by Composing Solutions of Elemental Sequential Tasks", *Machine Learning*, Vol.8 pp.323–339 (1992)

[7] Sutton, R.S.: "Learning to Predict by the Method of Temporal Differences", *Machine Learning*, Vol.3 pp.9–44 (1988)

[8] Sutton, R.S.: "Integrated Architectures for Learning, Planning, and Reacting Based on Approximating Dynamic Programming", *Proceedings of the 7th International Conference on Machine Learning (ICML'90)*, pp.216–224 (1990)

[9] Sutton, R.S. and Barto, A.G.: "Reinforcement Learning: An Introduction", MIT Press (1998)

[10] Tanaka, F. and Yamamura, M.: "An Approach to Lifelong Reinforcement Learning through Multiple Environments", *Proceedings of the 6th European Workshop on Learning Robots (EWLR-6)*, pp.93–99 (1997)

[11] Thrun, S. and Pratt, L.(eds.): "LEARNING TO LEARN", Kluwer Academic Publishers (1998)

[12] Watkins, C.J.C.H. and Dayan, P.: "Q-learning", *Machine Learning*, Vol.8 pp.279–292 (1992)

[13] 田中 文英: 未来型住宅のための適応エージェント技術，建築雑誌，Vol.117 No.1488 pp.85 (2002)

[14] 田中 文英，山村 雅幸: MDP 集合の分布上におけるマルチタスク強化学習（電気学会論文誌に投稿中）