

モデル誤差を利用したモデルベース強化学習

Model-based Reinforcement Learning with Model Error

但馬慶行†

Yoshiyuki TAJIMA

鬼沢武久†

Takehisa ONISAWA

1. はじめに

近年、ヒューマノイド・ロボット[1]、環境知能[2]、携帯情報端末[3]などのシステムの研究が盛んに行われている。これらのシステムは、ユーザと対話し協調するために、作りこみではなく、ユーザや環境の変化に適応していく自律的な学習能力や適応能力が必要である[4]。自律的な学習を実現する方法として、強化学習[5]が目目されている[6]。しかしながら、強化学習は、学習に多くの時間が必要なため、実用的な問題への適用には、学習の効率化を図る必要がある。その一つの方法としてモデルベース強化学習[5][7]が挙げられる。モデルベース強化学習では、通常の強化学習によるものに加え、環境の内部モデル（以下、モデル）を構築し、このモデルによる予測を利用することによって学習を加速させている。しかし、モデルに間違いがあると、学習に影響を与えるが、この影響について考慮している研究はない。

そこで本論文では、まず、モデル誤差が学習にどのように影響を与えるかを考察する。そして、モデル誤差に基づいて、モデルによる学習を適切に制御するモデルベース強化学習のアルゴリズム（Model Error based Forward Planning Reinforcement Learning: ME-FPRL）を提案し、その有効性を示す。

本論文は、第2章で、モデルベース強化学習におけるモデル誤差の問題とその対策を議論する。そして第3章で、アルゴリズムの説明を行う。第4章にて、掃除ロボットのシミュレーションにより評価を行う。最後に、第5章でまとめを行う。

2. 強化学習とモデルベース強化学習

2.1 強化学習とモデルベース強化学習の枠組み

強化学習は、意思決定し行動するシステム（以後エージェントと呼ぶ）が、環境との相互作用から定められた目標を達成する振る舞いを自律的に学習するための方法である。その枠組みを図1左に示す。環境はエージェントが相互作用する対象であり、エージェントの目標は、環境から与えられる報酬と呼ばれる信号を時間経過の中で最大化することとして定義される。

モデルベース強化学習では、エージェントがモデルを構築し、学習時に利用することによって学習の効率化を行う。すなわち、環境との相互作用からの学習（以下、直接的な学習）とモデルとの相互作用からの学習（以下、間接的な学習）を同時に行う。

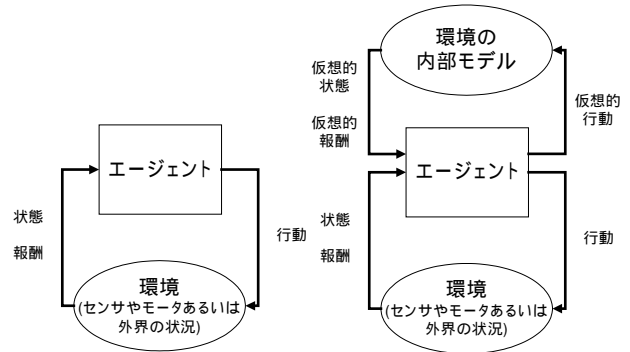


図1 強化学習(左)とモデルベース強化学習の枠組み(右)

2.2 モデルベース強化学習の問題点

エージェントは、直接的な学習として各試行のステップごとに、式(1)を用いて、ある時刻における状態 s での行動 a の行動価値 Q を学習する。なお、 s' 、 a' は次の時刻での状態とその状態に選ばれる行動である。また、 $\gamma (0 \leq \gamma \leq 1)$ は割引率、 $\alpha_d (0 \leq \alpha_d \leq 1)$ は学習率、 r はその状態での報酬を示している。

$$Q(s, a) = Q(s, a) + \alpha_d [r + \gamma Q(s', a') - Q(s, a)] \quad (1)$$

一方、1ステップの遷移軌跡に基づくモデルベース強化学習[7][8]の場合、エージェントは、式(2)のように、間接的に学習する。なお、 \hat{s}' 、 \hat{a}' は、モデルによって予測された次の時刻での状態とその状態に選ばれる行動であり、 $\alpha_i (0 \leq \alpha_i \leq 1)$ は間接的な学習での学習率である。

$$Q(s, a) = Q(s, a) + \alpha_i [r + \gamma Q(\hat{s}', \hat{a}') - Q(s, a)] \quad (2)$$

式(1)と式(2)のTD (Temporally Difference) 誤差（右辺第二項）[5]を $\Delta_d = \alpha_d [r + \gamma Q(s', a') - Q(s, a)]$ および、 $\Delta_i = \alpha_i [r + \gamma Q(\hat{s}', \hat{a}') - Q(s, a)]$ とする。また、 $s' = \hat{s}'$ となるために必要なステップ数を状態非類似度 (state dissimilarity: $sds(s', \hat{s}')$) として定義する。

ある時刻 t での状態の価値を表す累積報酬[5]は式(3)のように表される。目標にのみ報酬が与えられる場合、1ステップ目標から遠のくと価値は γ 倍に割引される。

$$Q_t(s, a) = \sum_{k=0}^{\infty} \gamma^k r_{t+k+1} \quad (3)$$

ここで、 $Q(s', a')$ に対して $Q(\hat{s}', \hat{a}')$ は $sds(s', \hat{s}')$ ステップ離れている。そのため、式(3)より、最も小さい場合、価値は割引率の $sds(s', \hat{s}')$ ステップ分小さくなり、価値は $\gamma^{sds(s', \hat{s}')}$ 倍となる。逆に最も大きい場合、価値は $\gamma^{-sds(s', \hat{s}')}$ 倍となる。また、状態が違ってても価値が等しい場

† 筑波大学大学院 システム情報工学研究科
知能機能システム専攻

合があることから、価値の間には式(4)の関係が得られる。ここで、 k は $-sds(s', \hat{s}') \leq k \leq sds(s', \hat{s}')$ を満たす。

$$Q(\hat{s}', \hat{a}') = \gamma^k Q(s', a') \quad (4)$$

全体の学習量 $\Delta_d + \Delta_i$ は、

$$\begin{aligned} \Delta_d + \Delta_i &= \alpha_d [r + \gamma Q(s', a') - Q(s, a)] + \alpha_i [r + \gamma Q(\hat{s}', \hat{a}') - Q(s, a)] \\ &= \alpha_d \left\{ [r + \gamma Q(s', a') - Q(s, a)] + \frac{\alpha_i}{\alpha_d} [r + \gamma Q(\hat{s}', \hat{a}') - Q(s, a)] \right\} \\ &= (\alpha_d + \alpha_i)r + \alpha_d \left[\left(1 + \frac{\alpha_i}{\alpha_d} \gamma^k\right) \gamma Q(s', a') - \left(1 + \frac{\alpha_i}{\alpha_d}\right) Q(s, a) \right] \end{aligned}$$

である。 k が負の方向に大きくなると $0 \leq \gamma \leq 1$ であるため、学習量は指数関数的に影響を受ける。すなわち、正しい価値を大きく破壊する可能性がある。たとえば、価値が収束状態の近傍で、 $\Delta_d > 0$ とした場合の影響を考察する。 $Q(s, a)$ が $Q(s', a')$ に対してのみ影響を受ける場合を考えると $\gamma Q(s', a') = Q(s, a)$ である。したがって、

$$\Delta_d + \Delta_i = (\alpha_d + \alpha_i)r + \alpha_i Q(s, a)(\gamma^k - 1)$$

となり、モデル誤差がない場合では $(\alpha_d + \alpha_i)r$ であるべき学習量が k に応じて増減してしまい、正しい学習が行われない。

2.3 モデル誤差への対策

モデル誤差へのひとつの対策として、定数であった α_i を $\alpha_i = \alpha_d h \gamma^{|k|}$ ($0 \leq h \leq 1$) とすると、

$$\begin{aligned} \Delta_d + \Delta_i &= (\alpha_d + \alpha_i)r + \alpha_d \left[\left(1 + \frac{\alpha_i}{\alpha_d} \gamma^k\right) \gamma Q(s', a') - \left(1 + \frac{\alpha_i}{\alpha_d}\right) Q(s, a) \right] \\ &= (\alpha_d + \alpha_i)r + \alpha_d \left[(1 + h \gamma^{|k|} \gamma^k) \gamma Q(s', a') - (1 + h \gamma^{|k|}) Q(s, a) \right] \end{aligned}$$

(1) $k \geq 0$ のとき

$$\begin{aligned} \Delta_d + \Delta_i &= (\alpha_d + \alpha_i)r + \alpha_d \left[(1 + h \gamma^k \gamma^k) \gamma Q(s', a') - (1 + h \gamma^k) Q(s, a) \right] \end{aligned}$$

となる。たとえば、 $\gamma Q(s', a') = Q(s, a)$ の場合、

$$\begin{aligned} \Delta_d + \Delta_i &= (\alpha_d + \alpha_i)r + \alpha_d \left[(1 + h \gamma^k \gamma^k) Q(s, a) - (1 + h \gamma^k) Q(s, a) \right] \\ &= (\alpha_d + \alpha_i)r + \alpha_d Q(s, a) h \gamma^k (\gamma^k - 1) \end{aligned}$$

である。つまり、 k が大きくなるにつれ、 $(\alpha_d + \alpha_i)r$ に近づき、モデル誤差の影響が小さくなる。

(2) $k < 0$ のとき

$$\begin{aligned} \Delta_d + \Delta_i &= (\alpha_d + \alpha_i)r + \alpha_d \left[(1 + h \gamma^{-k} \gamma^k) \gamma Q(s', a') - (1 + h \gamma^{-k}) Q(s, a) \right] \\ &= (\alpha_d + \alpha_i)r + \alpha_d \left[(1 + h) \gamma Q(s', a') - (1 + h \gamma^{-k}) Q(s, a) \right] \end{aligned}$$

となる。たとえば $\gamma Q(s', a') = Q(s, a)$ の場合、

$$\begin{aligned} \Delta_d + \Delta_i &= (\alpha_d + \alpha_i)r + \alpha_d \left[(1 + h) Q(s, a) - (1 + h \gamma^{-k}) Q(s, a) \right] \\ &= (\alpha_d + \alpha_i)r + \alpha_d Q(s, a) h (1 - \gamma^{-k}) \end{aligned}$$

である。つまり、 k が小さくなるにつれ、 $(\alpha_d + \alpha_i)r + \alpha_d Q(s, a) h$ に近づく。したがって、モデル誤差による影響は、最大でも $\alpha_d Q(s, a) h$ となる。

以上から、 $\alpha_i = \alpha_d h \gamma^{|k|}$ の関係が成り立つ場合には、モデルに誤差を生じていても影響を受けにくい学習を行うことができる。ところが、 $\alpha_i = \alpha_d h \gamma^{|k|}$ における k は学習時に正確に決定することはできない。一方、 $-sds(s', \hat{s}') \leq k \leq sds(s', \hat{s}')$ であることから、 $|k| \leq sds(s', \hat{s}')$ である。ゆえに、 $h \gamma^{|k|} \geq h \gamma^{sds(s', \hat{s}')}$ となり、 $h \gamma^{sds(s', \hat{s}')}$ としても k の影響を抑えられる。したがって、状態非類似度に基づいて、間接学習での「安全」なメタパラメータを設定することができる。ただし、この本論文での「安全」とは、指数関数的な間違いを防ぐことを意味している。

3. ME-FPRL

2.3 節のモデル誤差とパラメータの関係を踏まえた学習アルゴリズム (Model Error based Forward Planning Reinforcement Learning: ME-FPRL) を提案する。ME-FPRL のアルゴリズムを図 2 に示す。ME-FPRL は、遷移軌跡に基づく価値関数の更新を行うモデルベース強化学習に、状態非類似度に基づいた学習量の制御機構を導入したアルゴリズムである。エージェントは、直接学習、環境のモデルを学習すると同時に、その環境のモデルを使って次の時刻の状態を予測する。そして、予測した状態と実際に観測される状態との差から、その時点での状態非類似度を計算する。各時刻で計算された状態非類似度を使って、一般に、ある状態である行動を選択するときの状態非類似度を予測した値 $ModelAcy(s, a)$ ($0 \leq ModelAcy \leq 1$) を式(5)により推定する。式(5)において、 ρ は最近の状態非類似度をどの程度重視するかを定めるパラメータである。 ρ が 1 に近づくとき最近の状態非類似度を重視する。

$$\begin{aligned} ModelAcy(s, a) &= (1 - \rho) \cdot ModelAcy(s, a) + \rho \cdot \gamma^{sds(s', \hat{s}')} \quad (5) \end{aligned}$$

次に、間接学習において、状態を予測することにモデル精度を掛け合わせた $ComAcy$ ($0 \leq ComAcy \leq 1$) を計算する。 $ComAcy$ は予測を開始した状態から離れるにつれて小さくなる。この $ComAcy$ を間接学習でのステップサ

イズパラメータに掛け合わせることで、間接的な学習における適切な学習量にする。なお、ME-FPRL では、 N_l ステップ先までの予測を N_p 回行う。更新は、スタックを用いて N_l ステップ先の予測状態から現在の状態へ順番に行う。 ε -greedy 方策は、 ε の確率でランダムに行動を選択し、それ以外では価値の高い行動を選択する。

(0) 各変数を初期化、 $a \leftarrow \varepsilon$ -greedy(s, Q)
各試行に対して繰り返し：

(1) s 現在の状態(非終端)

(2) 行動 a を取り、結果の状態 s' と報酬 r を観測

(3) $a' \leftarrow \varepsilon$ -greedy(s', Q)

(4) $Q(s, a)$

$$Q(s, a) + \alpha[r + \gamma Q(s', a') - Q(s, a)]$$

(5) $SModel$ 、 $RModel$ から状態 s' と報酬 r を予測

(6) $ModelAcy(s, a)$

$$(1 - \rho) \cdot ModelAcy(s, a) + \rho \cdot h^{\gamma sds(s', \hat{s})}$$

(7) N_p 繰り返し：

(7-1) $s_p \leftarrow s, a_p \leftarrow a, ComAcy \leftarrow 1$

(7-2) N_l 繰り返し：

(7-2-1) $\hat{r} \leftarrow RModel, \hat{s}' \leftarrow SModel(\hat{s}, \hat{a})$

(7-2-2) $\hat{a}' \leftarrow \varepsilon$ -greedy(\hat{s}, Q)

(7-2-3) $ComAcy \leftarrow ComAcy \cdot ModelAcy(\hat{s}, \hat{a})$

(7-2-4) スタックに $(\hat{s}, \hat{a}, \hat{r}, \hat{s}', \hat{a}', ComAcy)$ を積む

(7-2-5) $\hat{s} \leftarrow \hat{s}', \hat{a} \leftarrow \hat{a}'$

(7-3) スタックがなくなるまで繰り返し：

(7-3-1) $(\hat{s}, \hat{a}, \hat{r}, \hat{s}', \hat{a}', ComAcy)$ を取り出す

(7-3-2) $Q(\hat{s}, \hat{a}) \leftarrow Q(\hat{s}, \hat{a}) +$

$$ComAcy \cdot \frac{\alpha}{N_p} [\hat{r} + \gamma Q(\hat{s}', \hat{a}') - Q(\hat{s}, \hat{a})]$$

(8) $s \leftarrow s', a \leftarrow a', \hat{s}' \leftarrow SModel(s', a')$

図2 ME-FPRL のアルゴリズム

4. 実験

4.1 効率の評価

本論文における効率化とは、設定された問題をいかに少ないステップで達成するかである。さらに、単に未学習の状態から学習を進めるだけでなく、ある程度学習が進んでから、環境が変化した場合も評価する。

4.2 掃除ロボットシミュレーション

ME-FPRL の検証のために掃除ロボットのシミュレーション実験を行う。掃除ロボットは、図3左で示されるような部屋を残らず掃除することが目標である。ここで、図中の黒色で示される場所は障害物を意味する。掃除ロボットが観測できるのは、自分の位置、上下左右の障害物、その場所の掃除状況、上下左右の方向の掃除状況である。この観測される情報をもとに、掃除ロボットは上下左右の移動

および掃除の5つの行動を取り目標を達成する。報酬は、未掃除状態から掃除済み状態にすることができれば10、直前の位置に戻ると-3、掃除済み状態に位置すると-2、それ以外-1とする。実験では、部屋をすべて掃除するまでを1試行として、その試行を500回達成させた後、部屋を図3右で示すように変え、さらに500回達成するまで行う。

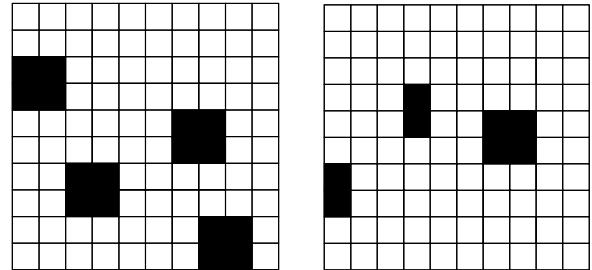


図3 変化前(左)、変化後(右)の部屋

4.3 学習エージェント

(1)直接的な学習のみ (2)ME-FPRL (3)通常モデルベース強化学習のエージェントで行う。各エージェントのパラメータは、 $\alpha=0.1, \gamma=0.95, \varepsilon=0.05, \rho=0.3, N_p=16, N_l=10, l=0.9$ としている。また、ME-FPRL のモデル誤差に対する適応能力を測るために、誤差の大きいものと小さいものの2つのモデルを用意する。誤差の大きいモデルは、部屋のモデルを実際に行動する際に観測しているセンサの状態どおりに更新するように、誤差の小さいモデルは、間接的な学習での行動も含め行動するたびにモデルに更新するようにモデルを構築している。したがって、前者のモデルは、間接的な学習で、エージェントが部屋を掃除したことを記憶できず、未掃除領域を掃除することに関する予測は成功しない。一方、後者のモデルは、過去に同じ状況で掃除することを経験していれば、掃除することによって起こる部屋の変化も予測できる。

4.4 実験結果と考察

図4に誤差の大きいモデルにおける学習の推移、図5に誤差の小さいモデルにおける学習の推移、表1に全試行の平均と定常状態の安定性を示す。ここで、定常状態は101回目から500回目および601回目から1000回目の試行としている。

誤差の大きいモデルにおいて、モデルベース強化学習は、定常状態で500ステップを越える回数が直接学習のみの場合31回であるのに対して、370回となり不安定になる。全試行のステップ数の平均についても直接学習のみに比べ2倍程度となり悪化している。一方、ME-FPRL は、直接学習のみとほとんど同程度となっており安定している。すなわち、提案手法はモデルの誤差の影響を受けていない。したがって、モデル誤差により間接的な学習を制御することは効果的であったといえる。

一方、予測精度の高い誤差の小さいモデルを用いた学習では、正確なモデルが構築されるため直接学習のみに比べ、モデルベースの両手法は効率的に学習できている。さらに、モデルによる先読みすることで、定常状態において500ステップを越えることはほとんどなく安定化がなされている。ただし、ME-FPRL は誤差がないことを徐々に検証するため、学習初期ではモデルベース強化学習に比べ速くはない。

表 1 全試行の平均と定常状態の安定性

	全試行のステップ数の平均	定常状態で 500 ステップを越える回数
直接学習のみ	373.9	31
ME-FPRL (誤差大)	377.2	31
モデルベース強化学習 (誤差大)	743.6	370
ME-FPRL (誤差小)	295.7	0
モデルベース強化学習 (誤差小)	276.6	1

5. おわりに

本論文では、これまでのモデルベース強化学習での学習の危険性を指摘し、改善策としてモデルの誤差に応じて適切に学習を制御するモデルベース強化学習のアルゴリズムを提案した。また、掃除ロボットシミュレーションにて提案したアルゴリズムの有効性を確かめた。

今後の課題としては、予測による先読みとその信頼性に基づいて、エージェント同士が協調的にコミュニケーションすることで、学習の加速を図る手法を開発することが挙げられる。

参考文献

[1] 五十棲 隆勝: ヒューマノイド・ロボットの開発, 川田 技報, Vol. 22, 2003

- [2] 塩見 昌裕, 神田 崇行, Daniel Eaton, 石黒 浩: コビキタスセンサネットワークと連動したコミュニケーションロボットによる科学館での展示案内, インタラクション 2005, pp.127-134, 2005
- [3] 安随晋太郎, 福田聡, 濱崎雅弘, 大向一輝, 武田英明, 山口高平: オントロジーに基づく携帯情報端末用レコメンデーションシステムの構築, 第 19 回人工知能学会全国大会, 2D2-03, 2005
- [4] 高田 敏弘: コビキタスサービスと実空間コンピューティング, 人工知能学会誌, Vol. 19, No. 4, pp. 454-461, 2004
- [5] Richard S. Sutton, Andrew G. Barto: Reinforcement Learning: An Introduction, MIT Press, Cambridge, MA, 1998
- [6] 山口智浩: 人工知能分野における強化学習研究の広がり, 人工知能学会全国大会論文集, Vol. JSAI02, pp.89-90, 2002
- [7] Richard S. Sutton: Dyna, an Integrated Architecture for Learning, Planning, and Reacting, Appeared in Working Notes of the 1991 AAAI Spring Symposium, pp.151-155, 1991
- [8] Kuvayev, L., Richard S. Sutton: Model-based reinforcement learning with an approximate, learned model. Proceedings of the Ninth Yale Workshop on Adaptive and Learning Systems, pp. 101-105, Yale University, New Haven, CT, 1996

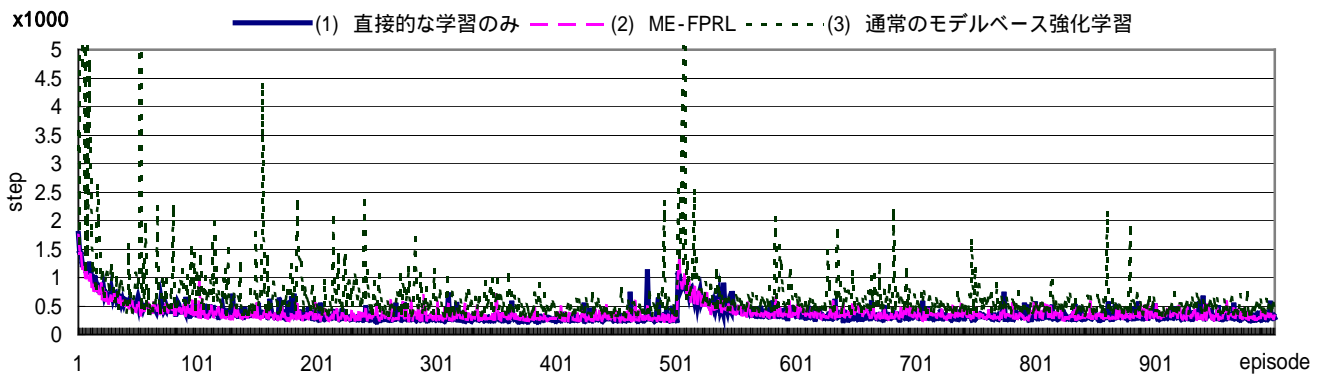


図 4 誤差の大きいモデルにおける学習の推移

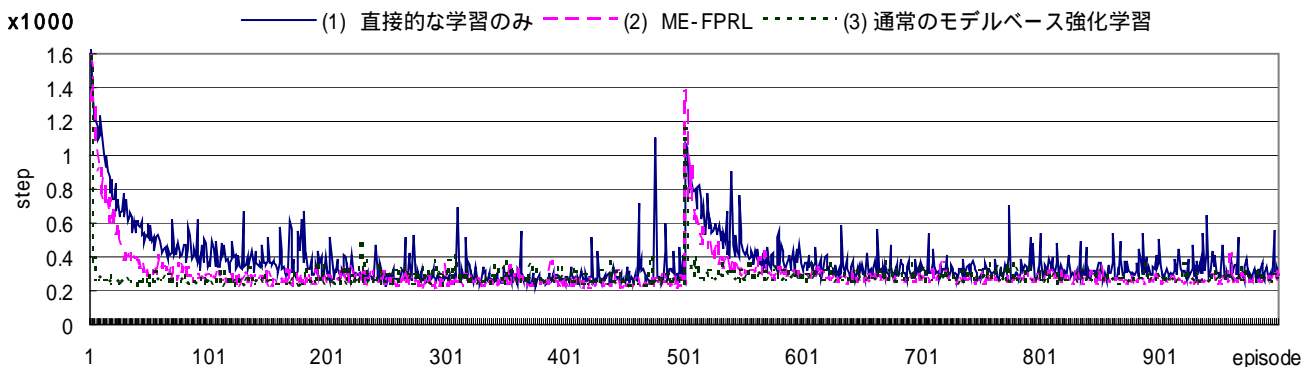


図 5 誤差の小さいモデルにおける学習の推移