Parallel reinforcement learning for weighted multi-criteria model with adaptive margin

Kazuyuki Hiraoka · Manabu Yoshida ·
Taketoshi Mishima

Received: 10 April 2008 / Revised: 16 September 2008 / Accepted: 16 September 2008 / Published online: 7 October 2008
© Springer Science+Business Media B.V. 2008

Abstract Reinforcement learning (RL) for a linear family of tasks is described in this paper. The key of our discussion is nonlinearity of the optimal solution even if the task family is linear; we cannot obtain the optimal policy using a naive approach. Although an algorithm exists for calculating the equivalent result to Q-learning for each task simultaneously, it presents the problem of explosion of set sizes. We therefore introduce adaptive margins to overcome this difficulty.

Keywords Reinforcement learning · Multi-criteria · Convex hull · Minkowski sum

Introduction

Much attention has recently been given to reinforcement learning (RL) and related subjects in engineering and biological fields (Sutton and Barto 1998; Tsukada et al. 2007; Freeman 2007). The purpose of this paper is to explain RL for a linear family of tasks. Such learning is useful in various scenes including time-varying environments, multi-criteria problems, and inverse RL (Natarajan and Tadepalli 2005, Hiraoka and Mishima 2006). The family is defined as a weighted sum of several criteria. This family is linear in the sense that the reward is linear with respect to weight parameters. For instance, criteria of network routing include end-to-end delay, loss of packets, and the power level associated with a node (Natarajan and

Tadepalli 2005). Selecting appropriate weights beforehand is difficult in practice and we are compelled to use trial and error. In addition, appropriate weights might change someday. Parallel RL for all possible weight values is desirable in such cases.

The key of our discussion is nonlinearity of the optimal solution; it is not linear but actually piecewise linear, which implies that we cannot obtain the best policy using the following naive approach.

1. Find the value function for each criterion.
2. Calculate their weighted sum to obtain the total value function.
3. Construct a policy based on the total value function.

A typical example is presented in the Experiments section.

Piecewise linearity of the optimal solution has been pointed out independently in Kaneko et al. (2005) and Natarajan and Tadepalli (2005). The latter is intended at fast adaptation under time-varying environments. The former is our previous report; we have tried to obtain optimal solutions for various weight values all together, and we have developed an algorithm that gives an exactly equivalent solution to Q-learning for each weight value. However, it presents difficulty with the explosion of set size. This difficulty is not a problem of the algorithm but an intrinsic characteristic of Q-learning for the weighted criterion model.

We introduced a simple approximation with a ‘margin’ into the decision of convexity first (Hiraoka and Mishima 2006). Then we improved it so that we obtain an interval estimation, and we can monitor the approximation effects (Yoshida et al. 2007). In this paper, we propose adaptive adjustment of margins.

In a margin-based approach, we must manage large sets of vectors in the first stage of learning. The peak of the set

K. Hiraoka (✉) · M. Yoshida · T. Mishima
Graduate School of Science and Engineering,
Saitama University, 255 Shimo-Okubo, Sakura-ku,
Saitama-shi, Japan
e-mail: hira@ics.saitama-u.ac.jp

size tends to be large if we set a small margin to obtain an accurate final result. The proposed method reduces the worry of this trade-off. By changing margins appropriately through learning steps, we can enjoy a small set size in the first stage with large margins and an accurate result in the final stage with small margins.

Below, the weighted criterion model is defined first, and parallel RL for it is described. Then the difficulty of set size is pointed out and margins are introduced. Adaptive adjustment of margins is also proposed there. Its behavior is verified experimentally; a conclusion is presented finally.

Weighted criterion model

An “orthodox” RL setting is assumed for states and actions as follows.

- The time step is discrete ($t = 0, 1, 2, 3, \dots$).
- The state set S and the action set A are finite and known.
- The state transition rule $\mathcal{P}$ is unknown.
- The state s_t is observable.
- The task is a Markov decision process (MDP).

The reward r_{t+1} is given as a weighted sum of partial rewards $r_{t+1}^1, \dots, r_{t+1}^M$:

$$r_{t+1}(\boldsymbol{\beta}) = \sum_{i=1}^M \beta_i r_{t+1}^i = \boldsymbol{\beta} \cdot \mathbf{r}_{t+1}, \quad (1)$$

weight vector $\boldsymbol{\beta} \equiv (\beta_1, \dots, \beta_M) \in \mathbb{R}^M$,

reward vector $\mathbf{r}_{t+1} \equiv (r_{t+1}^1, \dots, r_{t+1}^M) \in \mathbb{R}^M$.

We assume that the partial rewards $r_{t+1}^1, \dots, r_{t+1}^M$ are also observable, but that their reward rules $\mathcal{R}(1), \dots, \mathcal{R}(M)$ are unknown. Multi-criteria RL problems of this type have been introduced independently in Kaneko et al. (2004) and Natarajan and Tadepalli (2005).

We hope to find the optimal policy $\pi_{\boldsymbol{\beta}}^*$ for each weight $\boldsymbol{\beta}$ that maximizes the expected cumulative reward with a given discount factor $0 < \gamma < 1$,

$$\pi_{\boldsymbol{\beta}}^* = \operatorname{argmax}_{\pi} E^{\pi} \left[\sum_{\tau=0}^{\infty} \gamma^{\tau} r_{\tau+1}(\boldsymbol{\beta}) \right],$$

where $E^{\pi}[\cdot]$ denotes the expectation under a policy π . To be exact, π^* is defined as a policy that attains

$$Q_{\boldsymbol{\beta}}^{\pi_{\boldsymbol{\beta}}^*}(s, a; \gamma) = Q_{\boldsymbol{\beta}}^*(s, a; \gamma) \equiv \max_{\pi} Q_{\boldsymbol{\beta}}^{\pi}(s, a; \gamma)$$

for all state-action pairs (s, a) , where the action-value function $Q_{\boldsymbol{\beta}}^{\pi}$ is defined as

$$Q_{\boldsymbol{\beta}}^{\pi}(s, a; \gamma) \equiv E^{\pi} \left[\sum_{\tau=0}^{\infty} \gamma^{\tau} r_{\tau+1}(\boldsymbol{\beta}) \mid s_0 = s, a_0 = a \right].$$

MDP is well known to have a deterministic policy $\pi_{\boldsymbol{\beta}}^*$ that satisfies the above condition; such $\pi_{\boldsymbol{\beta}}^*$ is obtained from the optimal value function (Sutton and Barto 1998),

$$\pi_{\boldsymbol{\beta}}^* : S \rightarrow A : s \mapsto \operatorname{argmax}_{a \in A} Q_{\boldsymbol{\beta}}^*(s, a; \gamma).$$

Consequently, we concentrate on estimation of $Q_{\boldsymbol{\beta}}^*$. Note that $Q_{\boldsymbol{\beta}}^*$ is nonlinear with respect to $\boldsymbol{\beta}$. A typical example is presented in the section of experiments. Basic properties of the action-value function Q are described briefly in the remainder of this section (Kaneko et al. 2005; Natarajan and Tadepalli 2005; Hiraoka and Mishima 2006). The discount factor γ is fixed through this paper, and it is omitted below.

Proposition 1 $Q_{\boldsymbol{\beta}}^{\pi}(s, a)$ is linear with respect to $\boldsymbol{\beta}$ for a fixed policy π .

Proof Neither $\mathcal{P}$ nor π depend on $\boldsymbol{\beta}$ from assumptions. Hence, the joint distribution of $(s_0, a_0), (s_1, a_1), \dots$ is independent of $\boldsymbol{\beta}$. It implies linearity. $\square$

Definition 1 If $f : \mathbb{R}^M \rightarrow \mathbb{R}$ can be written as $f(\boldsymbol{\beta}) = \max_{\mathbf{q} \in \Omega} (\mathbf{q} \cdot \boldsymbol{\beta})$ with a nonempty finite set $\Omega \subset \mathbb{R}^M$, we call f Finite-Max-Linear (FML) and write it as $f = \text{FML}_{\Omega}$.

It is trivial that f is convex and piecewise-linear if f is FML.

Proposition 2 The optimal action-value function is FML as a function of the weight $\boldsymbol{\beta}$. Namely, there exists a nonempty finite set $\Omega^*(s, a) \subset \mathbb{R}^M$ for each state-action pair (s, a) , and $Q_{\boldsymbol{\beta}}^*$ is written as

$$Q_{\boldsymbol{\beta}}^*(s, a) = \max_{\mathbf{q} \in \Omega^*(s, a)} \mathbf{q} \cdot \boldsymbol{\beta}. \quad (2)$$

Proof We have assumed MDP. Then $Q_{\boldsymbol{\beta}}^*$ can be written as $Q_{\boldsymbol{\beta}}^*(s, a) = \max_{\pi \in \Pi} Q_{\boldsymbol{\beta}}^{\pi}(s, a)$ for the set Π of all deterministic policies. Π is finite, and $Q_{\boldsymbol{\beta}}^{\pi}$ is linear with respect to $\boldsymbol{\beta}$ from Proposition 1. Hence, $Q_{\boldsymbol{\beta}}^*$ is FML. $\square$

Proposition 3 Assume that an estimated action-value function $Q_{\boldsymbol{\beta}}$ is FML as a function of the weight $\boldsymbol{\beta}$. If we apply Q -learning, the updated

$$Q_{\boldsymbol{\beta}}^{\text{new}}(s_t, a_t) = (1 - \alpha) Q_{\boldsymbol{\beta}}(s_t, a_t) + \alpha \left\{ \boldsymbol{\beta} \cdot \mathbf{r}_{t+1} + \gamma \max_{a \in A} Q_{\boldsymbol{\beta}}(s_{t+1}, a) \right\} \quad (3)$$

is still FML as a function of $\boldsymbol{\beta}$, where $\alpha > 0$ is the learning rate.

Proof There exists a nonempty finite set $\Omega(s, a) \subset \mathbb{R}^M$ such that $Q_{\boldsymbol{\beta}}(s, a) = \max_{\mathbf{q} \in \Omega(s, a)} (\mathbf{q} \cdot \boldsymbol{\beta})$ for each (s, a) . Then (3) implies $Q_{\boldsymbol{\beta}}^{\text{new}}(s_t, a_t) = \max_{\tilde{\mathbf{q}} \in \tilde{\Omega}} \tilde{\mathbf{q}} \cdot \boldsymbol{\beta}$, where

$$\tilde{\Omega} \equiv \{ (1 - \alpha)\mathbf{q} + \alpha(\mathbf{r}_{t+1} + \gamma \mathbf{q}') \mid \mathbf{q} \in \Omega(s_t, a_t), \mathbf{q}' \in \Omega(s_{t+1}, a), a \in A \}, \quad (4)$$

because $\max_x f(x) + \max_y g(y) = \max_{x,y} (f(x) + g(y))$ holds in general. The set $\tilde{\Omega}$ is finite; Q_{β}^{new} is FML. $\square$

These propositions imply that (1) the true Q_{β}^* is FML, and (2) its estimation Q_{β} is also FML, as long as the initial estimation is FML.

Parallel Q-learning for all weights

A parallel Q-learning method for the weighted criterion model has been proposed in Hiraoka and Mishima (2006). The estimation Q_{β} for all $\beta \in \mathbb{R}^M$ are updated all together in parallel Q-learning.

In this method, $Q_{\beta}(s, a)$ for each (s, a) is treated in an FML expression:

$$Q_{\beta}(s, a) = \max_{q \in \Omega(s, a)} q \cdot \beta = \text{FML}_{\Omega(s, a)}(\beta) \quad (5)$$

with a certain set $\Omega(s, a) \subset \mathbb{R}^M$. We store and update $\Omega(s, a)$ instead of $Q_{\beta}(s, a)$ on the basis of Propositions 2 and 3. Although a naive update rule has been suggested in the proof of Proposition 3, it is extremely redundant and inefficient. We require several definitions to describe a better algorithm.

Definition 2 An element $c \in \Omega$ is redundant if

$$\text{FML}_{\Omega - \{c\}} = \text{FML}_{\Omega}.$$

Definition 3 We use $\Omega^{\dagger}$ to represent non-redundant elements in Ω .

Note that $\text{FML}_{\Omega^{\dagger}} = \text{FML}_{\Omega}$ (Natarajan and Tadepalli 2005).

Definition 4 We define the following operations:

$$\begin{aligned} c\Omega &\equiv \{cq | q \in \Omega\}, c + \Omega \equiv \{c + q | q \in \Omega\}, \\ \Omega \sqcup \Omega' &\equiv (\Omega \cup \Omega')^{\dagger}, \bigsqcup_{k=1}^K \Omega_k \equiv \left(\bigcup_{k=1}^K \Omega_k \right)^{\dagger}, \\ \Omega \oplus \Omega' &\equiv \{q + q' | q \in \Omega, q' \in \Omega'\}, \\ \Omega \boxplus \Omega' &\equiv (\Omega \oplus \Omega')^{\dagger}. \end{aligned} \quad (6)$$

With these operations, the updating rule of Ω is described as the following (Hiraoka and Mishima 2006).

$$\Omega^{\text{new}}(s_t, a_t) = (1 - \alpha)\Omega(s_t, a_t) \boxplus \alpha \left(r_{t+1} + \gamma \bigsqcup_{a \in A} \Omega(s_{t+1}, a) \right). \quad (7)$$

The initial value of Ω at $t = 0$ is $\Omega(s, a) = \{o\} \subset \mathbb{R}^M$ for all $(s, a) \in S \times A$. It corresponds to a constant initial function $Q_{\beta}(s, a) = 0$.

Proposition 4 Suppose that (5) holds for all states $s \in S$ and actions $a \in A$. Then $Q_{\beta}^{\text{new}}(s_t, a_t)$ in (3) is equal to

$\text{FML}_{\Omega^{\text{new}}(s_t, a_t)}(\beta)$ for (7). The proposed parallel Q-learning is therefore equivalent to Q-learning for each β .

$$\begin{array}{ccc} & \text{update} & \\ \text{FML} & \begin{array}{c} \{Q_{\beta}(s, a)\} \\ \updownarrow \\ \{\Omega(s, a)\} \end{array} & \begin{array}{c} \rightarrow Q_{\beta}^{\text{new}}(s_t, a_t) \\ \updownarrow \\ \Omega^{\text{new}}(s_t, a_t) \\ \text{update} \end{array} \\ & & \text{FML} \end{array}$$

Proof We have introduced a set $\tilde{\Omega}$ in (4) to prove Proposition 3. With the above operations, (4) is written as

$$\tilde{\Omega} = (1 - \alpha)\Omega(s_t, a_t) \oplus \alpha \left(r_{t+1} + \bigsqcup_{a \in A} \Omega(s_{t+1}, a) \right).$$

Then $(\tilde{\Omega})^{\dagger} = \Omega^{\text{new}}(s_t, a_t)$ is obtained and

$$\text{FML}_{\Omega^{\text{new}}(s_t, a_t)}(\beta) = \text{FML}_{\tilde{\Omega}(s_t, a_t)}(\beta) = Q_{\beta}^{\text{new}}(s_t, a_t)$$

is implied. $\square$

In fact, $\Omega^{\dagger}$ is well known to be equal to the vertices in the convex hull of Ω (Hiraoka and Mishima 2006). Efficient algorithms of the convex hull have been developed in computational geometry (Preparata and Shamos 1985). Using them, we can calculate the merged set $(\Omega \sqcup \Omega') = (\Omega \cup \Omega')^{\dagger}$. The sum set $(\Omega \boxplus \Omega')$ has also been studied as Minkowski sum algorithms (Bekker and Roerdink 2001; Fukuda 2004; Fogel and Halperin 2006). Its calculation is particularly easy for two-dimensional convex polygons (Fig. 1).

Before closing the present section, we note an FML version of the Bellman equation in our notation. Theoretically, we can use successive iteration of this equation to find the optimal policy when we know $\mathcal{P}$ and $\mathcal{R}$, although we must take care of numerical error in practice.

Proposition 5 The FML expression $Q_{\beta}^* = \text{FML}_{\Omega^*}(\beta)$ satisfies

$$\Omega^*(s, a)^{\dagger} = r_s^a + \gamma \bigsqcup_{a' \in A} \bigsqcup_{s' \in S} \mathcal{P}_{ss'}^a \Omega^*(s', a'), \quad (8)$$

Fig. 1 Calculation of $\Omega \boxplus \Omega'$ in (6) for two-dimensional convex polygons. Vertices of polygons correspond to Ω , Ω' and $\Omega \boxplus \Omega'$

where

$$\begin{aligned} \mathbf{r}_s^a &= (\mathcal{R}_s^a(1), \dots, \mathcal{R}_s^a(M)), \\ \mathcal{R}_s^a(i) &= E[r_{t+1}^i | s_t = s, a_t = a], \\ \mathcal{P}_{ss'}^a &= P(s_{t+1} = s' | s_t = s, a_t = a), \\ \boxplus_{s' \in \{s_1, \dots, s_k\}} X_{s'} &= X_{s_1} \boxplus X_{s_2} \boxplus \dots \boxplus X_{s_k}. \end{aligned}$$

In particular, the next equation holds if the state transition is deterministic:

$$\Omega^*(s, a)^\dagger = \mathbf{r}_s^a + \gamma \bigsqcup_{a' \in A} \Omega^*(s', a'),$$

where s' is the next state for the action a at the current state s .

Proof Substituting (2) and $\mathcal{R}_{s,\beta}^a \equiv E[r_{t+1}(\beta) | s_t = s, a_t = a] = \mathbf{r}_s^a \cdot \beta$ into the Bellman equation $Q_\beta^*(s, a) = \mathcal{R}_{s,\beta}^a + \gamma \sum_{s' \in S} \mathcal{P}_{ss'}^a \max_{a' \in A} Q_\beta^*(s', a')$, we obtain

$$\begin{aligned} \max_{q \in \Omega^*(s,a)} q \cdot \beta &= \max_{q' \in \Omega^*(s,a)} q' \cdot \beta, \\ \Omega^*(s, a) &= \bigcup_{a' \in A} \left\{ \mathbf{r}_s^a + \gamma \sum_{s' \in S} \mathcal{P}_{ss'}^a q'_{s'} \mid q'_{s'} \in \Omega^*(s', a') \right\} \end{aligned}$$

in the same way as (4). Consequently, Ω^* is equal to Ω' , except for redundancy. $\square$

Interval operations

Under regularity conditions, Q-learning has been shown to converge to Q^* (Jaakkola et al. 1994). That result implies pointwise convergence of parallel Q-learning to Q_β^* for each β because of Proposition 3. From Proposition 2, $Q_\beta^*(s, a)$ is expressed with a finite $\Omega^*(s, a)$.

However, as presented in Fig. 1, the number of elements in the set $\Omega(s, a)$ increases monotonically; it never ‘converges’ to $\Omega^*(s, a)$. This is not a paradox: the following assertions can be true simultaneously (Fig. 2).

1. Vertices of convex polygons $P_1, P_2, \dots$ increase monotonically.
2. P_t converges to a convex polygon P^* in the sense that the volume of the difference $P_t \triangle P^* = (P_t \cup P^*) - (P_t \cap P^*)$ converges to 0.
- 2'. The function $\text{FML}_{P_t}(\cdot)$ converges pointwise to $\text{FML}_{P^*}(\cdot)$.

Fig. 2 Illustration of convergence of parallel Q-learning. Though the ‘shapes’ of polygons converge, the numbers of their vertices increase infinitely

In short, pointwise convergence of a piecewise-linear function does not imply convergence of the number of pieces.

We can interpret the mechanism of this trouble as follows. The updating rule (7) is divisible into three procedures:

- Shrink the previously estimated polygon slightly.
- Obtain an instantaneously estimated polygon and shrink it.
- Merge them using the Minkowski sum $\boxplus$.

Then we notice that iterations of these procedures yield many small edges. It is not a problem of the algorithm because of Proposition 4. It is an intrinsic characteristic of pointwise Q-learning of the weighted criterion model for each weight β .

We attempted a simple approximation with a small ‘margin’ at first (Hiraoka and Mishima 2006) to overcome this difficulty. Then we introduced interval operations to monitor the approximation error (Yoshida et al. 2007). A pair of sets $\Omega^L(s, a)$ and $\Omega^U(s, a)$ is updated instead of the original $\Omega(s, a)$ so that

$$\text{CH } \Omega^L(s, a) \subset \text{CH } \Omega(s, a) \subset \text{CH } \Omega^U(s, a)$$

holds, where $\text{CH } Z$ represents the convex hull of Z . This relation implies lower and upper bounds $Q_\beta^L(s, a) \leq Q_\beta(s, a) \leq Q_\beta^U(s, a)$, where $Q_\beta^X(s, a) = \text{FML}_{\Omega^X(s,a)}(\beta)$ for $X = L, U$. It is guaranteed that the effect of the approximation can be ignored when the difference between Q^L and Q^U is sufficiently small. The updating rules of Ω^L and Ω^U are the same as those of Ω , except for the following approximations after every calculation of $\sqcup$ and $\boxplus$.

Lower approximation for Ω^L : A vertex is removed if the change of the measure (area for $M = 2$, volume for $M = 3$) of $\text{CH } \Omega^L(s, a)$ is smaller than a threshold $\epsilon^L/2$ (Fig. 3 left).

Upper approximation for Ω^U : A facet (edge for $M = 2$, face for $M = 3$) is removed if the change of the measure of $\text{CH } \Omega^U(s, a)$ is smaller than a threshold $\epsilon^U/2$ (Fig. 3 right).

In this paper, we propose an automatic adjustment of the margins ϵ^L, ϵ^U . The procedures described below are

Fig. 3 Lower approximation (left) and upper approximation (right) to simplify polygons

performed at every step t after the updating of Ω^L , Ω^U . The symbol X represents L or U here. $\xi_s, \xi_w \geq 1$ and $\theta_Q, \theta_\Omega \geq 0$ are constants.

1. Check the changes of set sizes and interval width compared to those of the previous ones. Namely, check these values:

$$\Delta_\Omega^X = |\Omega^{X\text{new}}(s_t, a_t)| - |\Omega^X(s_t, a_t)|,$$

$$\Delta_Q = \left(Q_\beta^{U\text{new}}(s_t, a_t) - Q_\beta^{L\text{new}}(s_t, a_t) \right) - \left(Q_\beta^U(s_t, a_t) - Q_\beta^L(s_t, a_t) \right),$$

where $|Z|$ is the number of elements in Z , and $\bar{\beta}$ is selected beforehand.

2. Increase of the set size suggests a need of thinning, whereas the increase of the interval width suggests the need for more accurate calculation. Modify margins as

$$\epsilon^{X\text{new}} = \begin{cases} \tilde{\epsilon}^X(\Delta_Q \leq \theta_Q) \\ \tilde{\epsilon}^X / \xi_w(\Delta_Q > \theta_Q) \end{cases},$$

$$\tilde{\epsilon}^X = \begin{cases} \epsilon^X & (\Delta_\Omega^X \leq \theta_\Omega) \\ \xi_s \epsilon^X & (\Delta_\Omega^X > \theta_\Omega) \end{cases}.$$

To avoid underflow, we set $\epsilon^{X\text{new}} = \epsilon_{\min}$ if $\epsilon^{X\text{new}}$ is smaller than a constant $\epsilon_{\min}$.

On numerical errors

We must alleviate a pitfall, numerical instability, in implementation of the proposed algorithm. As explained earlier, iterations of shrinkage procedures yield very small edges in updating by (7). They cause degeneration in numerical calculations; the obtained new vertices can be severely wrong in the upper approximation. Individual checks of such cases are onerous and oversights often worry us.

It is nonsense to try to satisfy $Q_\beta(s, a) \leq Q_\beta^U(s, a)$ exactly here because the input (the previously calculated Ω^U) itself contains numerical errors already. Each point in Fig. 3 should be viewed as a vague disk; we must accept errors that are comparable with these disks. This acceptance simplifies the process.

Before upper approximations, we apply lower approximations to Ω^U with a small fixed margin ϵ_L^l that corresponds to the above discussion. They remove overly small edges that might cause trouble otherwise.

Experiments with a basic task of weighted criterion

We have verified behaviors of the proposed method through experiments with a basic task. Although this task is simple, it illustrates a typical nonlinearity of the weighted criterion model: five kinds of optimal policies appear depending on the weight.

We set

$$S = \{S, G, A, B, X, Y\},$$

$$A = \{\text{Up}, \text{Down}, \text{Left}, \text{Right}\},$$

$s_0 = S$, and $\gamma = 0.8$ (Fig. 4) (Hiraoka and Mishima 2006). Each action causes a deterministic state transition to the corresponding direction except at G , where the agent is moved to S irrespective of its action. Rewards 1, 4b, b are offered respectively at $s_t = G, X, Y$. The state is unchanged and a negative reward (-1) is added further if a_t is an action to ‘outside wall’ at $s_t \neq G$.

It is a weighted criterion model of $M = 2$, because it can be written as the form $r_{t+1} = \beta \cdot r_{t+1}$ for $r_{t+1} = (r_{t+1}^1, r_{t+1}^2)$ and $\beta = (b, 1)$. The optimal policy changes depending on the weight b . Hence, the optimal value function is non-linear with respect to b (Table 1). Note that the second pattern ($S \rightarrow A \rightarrow Y$) in Table 1 cannot appear on the naive approach in the introduction.

Fig. 4 Task for experiments. It has 6 states (S, G, A, B, X, Y) and 4 actions (Up, Down, Left, Right). Numbers in parentheses are reward values. Five kinds of optimal policies appear depending on the weight parameter b

Table 1 Optimal state-value functions and optimal policies for the task in Fig. 4 at the initial state S

Range of weight	Optimal $V_b^*(S)$	Optimal state transition
$b < -16/25$	0	$S \rightarrow A \rightarrow S \rightarrow \dots$
$-16/25 \leq b < -225/1796$	$(2000b + 1280)/2101$	$S \rightarrow A \rightarrow Y \rightarrow B \rightarrow G \rightarrow S \rightarrow \dots$
$-225/1796 \leq b < 15/47$	$(400b + 80)/61$	$S \rightarrow X \rightarrow G \rightarrow S \rightarrow \dots$
$15/47 \leq b < 3/4$	$32b/3$	$S \rightarrow X \rightarrow Y \rightarrow X \rightarrow \dots$
$3/4 \leq b$	$16b - 4$	$S \rightarrow X \rightarrow X \rightarrow \dots$

Fig. 5 Experimental results of proposed algorithm. Upper: transition of margins ϵ^L and ϵ^U . Lower: transition of total set sizes $\sum_{s,a} |\Omega^L(s,a)|$, $\sum_{s,a} |\Omega^U(s,a)|$, and mean interval width $Q^U_\beta(s,a) -$

$Q^L_\beta(s,a)$ for all (s,a) at $\beta = (-0.4, 1) \neq \bar{\beta}$. The proposed algorithm realizes reduced set sizes and a small interval width simultaneously compared with Fig. 6

Fig. 6 Trade-off between set sizes and interval width in the conventional fixed-margin algorithm ($\epsilon^U = \epsilon^L = 10^{-4}$ and $\epsilon^U = \epsilon^L = 10^{-8}$). Left: set size. Right: interval width

The proposed algorithm is applied to this task with parameters $\alpha = 0.7$, $(\zeta_s, \zeta_w) = (1.3, 1.01)$, $(\theta_Q, \theta_\Omega) = (0.2)$, $\beta = (0.2, 1)$, $\epsilon_{\min} = \epsilon^L_L = 10^{-14}$. The initial margin $\epsilon^L = \epsilon^U$ at $t = 0$ is a 10^{-1} , 10^{-4} , 10^{-7} , 10^{-10} , 10^{-13} . The agent selects the estimated optimal action $a_{t+1} = \arg\max_{a \in A} Q^L_\beta(s_t, a)$ with probability 0.8, and it takes a random action otherwise. On this task, we can replace convex hulls with upper convex hulls in our algorithm because β is restricted to the upper half plane (Hiraoka and Mishima 2006). We add imaginary vertical edges to both ends in Fig. 3 for upper approximation. We also assume $|b| \leq 10 \equiv b_{\max}$ and we safely remove the edges on both ends in Fig. 3 if the absolute value of their slope is greater than $b_{\max}$ for lower approximation.

Averages of 1000 trials are shown in Fig. 5 for $t = 0, 500, 1000, 1500, \dots$. The proposed algorithm is robust to a wide range of initial margins. It realizes reduced set sizes and a small interval width simultaneously; these

requirements are a trade-off in the conventional fixed-margin algorithm (Yoshida et al. 2007) (Fig. 6). The proposed algorithm still works for inappropriate factor parameters ζ_s and ζ_w , although they cause an inappropriate balance between the above requirements (Figs. 7 and 8). It can fail if the parameters are further inappropriate; we have observed $\epsilon^U \approx 10^{13}$ at $t = 300$ when we set $\epsilon^L = \epsilon^U = 0.1$ at $t = 0$, $\zeta_s = 5.0$, $\zeta_w = 1.005$, and $\theta_\Omega = 1$. Another problem is slow convergence of the interval width $Q^U - Q^L$ compared to that of the fixed-margin algorithm.

Conclusion

A parallel RL method with adaptive margins is proposed for the weighted criterion model. Its behaviors are verified experimentally with a basic task. Adaptive margins realize reduced set sizes and accurate results.

Fig. 7 Results for inappropriate factor parameters $\zeta_s = 1.1$, $\zeta_w = 1.03$ in the proposed algorithm. (Upper) The margins ϵ^X decrease too fast. (Lower) They cause too large set sizes $|\Omega^X|$ and small interval width $Q^U - Q^L$

Fig. 8 Results for inappropriate factor parameters $\zeta_s = 5.0$, $\zeta_w = 1.005$ in the proposed algorithm. (Upper) The margins ϵ^X decrease too slow. (Lower) They cause small set sizes $|\Omega^X|$ and too large interval width $Q^U - Q^L$

A problem of the adaptive margins is parameter tuning. Although it is robust for initial margins, inappropriate factor parameters can cause an inappropriate balance between the above criteria. Another problem is slow convergence of the interval between upper and lower estimations. These points must be studied further.

References

- Bekker H, Roerdink JBTM (2001) An efficient algorithm to calculate the Minkowski sum of convex 3D polyhedra. LNCS 2073 part I. Springer Verlag, NY, USA
- Freeman WJ (2007) Definitions of state variables and state space for brain-computer interface Part I. Multiple hierarchical levels of brain function. Cogn Neurodyn 1:3–14

- Fogel E, Halperin D (2006) Exact and efficient construction of Minkowski sums of convex polyhedra with applications. In: Proc ALENEX, pp 3–15
- Fukuda K (2004) From the zonotope construction to the Minkowski addition of convex polytopes. *J Symbolic Comput* 38:1261–1272
- Hiraoka K, Mishima T (2006) Parallel reinforcement learning for tasks with weighted sum of partial rewards. *Brain Neural Netw, Jpn Neural Netw Soc* 13:137–145 (in Japanese)
- Jaakkola T, Jordan MI, Singh SP (1994) On the convergence of stochastic iterative dynamic programming algorithms. *Neural Comput* 6:1185–1201
- Kaneko Y, Hiraoka K, Mishima T (2004) Reinforcement learning for context dependent reward model. In: Proc IEICE society conference (in Japanese)
- Kaneko N, Hiraoka K, Mishima T (2005) Parallel reinforcement learning for linear family of infinite tasks. In: Proc IEICE society conference (in Japanese)
- Natarajan S, Tadepalli P (2005) Dynamic preferences in multi-criteria reinforcement learning. In: Proc int conf on machine learning, pp 601–608
- Preparata FP, Shamos MI (1985) Computational geometry. Springer-Verlag, NY, USA
- Sutton RS, Barto AG (1998) Reinforcement learning. The MIT Press, Massachusetts
- Tsukada M, Yamazaki Y, Kojima H (2007) Interaction between the Spatiotemporal Learning Rule (STLR) and Hebb type (HEBB) in single pyramidal cells in the hippocampal CA1 Area. *Cogn Neurodyn* 1:157–167
- Yoshida M, Hiraoka K, Mishima T (2007) Multi-criteria reinforcement learning based on interval estimation of value functions. In: Proc FIT (in Japanese)