論　文

最悪ケースを考慮した最適スーパバイザの強化学習*

梶原　弘治†・山﨑　達志†

Reinforcement Learning of Optimal Supervisor based on the Worst-Case Behavior*

Kouji KAJIWARA† and Tatsushi YAMASAKI†

The supervisory control initiated by Ramadge and Wonham is a framework for logical control of discrete event systems. In the original supervisory control, the costs for occurrence and disabling of events have not been considered. Then, the optimal supervisory control based on quatitative measures has also been studied. This paper proposes a synthesis method of the optimal supervisor based on the worst-case behavior of discrete event systems. We introduce the new value functions for the assigned control patterns. The new value functions are not based on the expected total rewards, but based on the most undesirable event occurrence in the assigned control pattern. In the proposed method, the supervisor learns how to assign the control pattern based on reinforcement learning so as to maximize the value functions. We show the efficiency of the proposed method by computer simulation.

1. はじめに

離散事象システム [1](DES; Discrete Event System) に対する制御方式の一つに，Ramadge と Wonham によって提案されたスーパバイザ制御がある [2]．スーパバイザ制御では，制御対象である離散事象システムが論理的に与えられる制御仕様を満たすように，生起を許可する事象の集合 (制御パターン) を指定する．このとき，不可制御事象により制御仕様を満たせない場合には，システムの生成言語が最大となるよう，最大可制御部分言語を求めることが行われている．元々のスーパバイザ制御では，事象の生起や禁止に伴うコストや確率といった要素は考慮されていなかったが，その後，それらをも考慮した最適スーパバイザ制御についての研究も行われている [3,4]．また，Ray らは，言語測度とよばれる符号付きの実測度を導入し [5]，OS におけるソフトウェアの実行制御やロボット制御などに適用している [6,7]．

一方，機械学習の一手法である強化学習 [8] は，受け取る割引付き期待報酬の総和を最大化するべく，試行錯誤を通じて最適な行動を獲得していく手法である．様々な探索を行い学習を進めるため，環境そのものが未知の場合やコストが正確にわからない場合に対して有効である．そのため，強化学習を用いた最適スーパバイザ制御の研究もなされている [9,10]．

元々のスーパバイザ制御は保守的な制御の枠組みであり，システムで生起しうるどのような事象列に対しても制御仕様を満たすように制御パターンを指定していた．そのため，許容した制御パターンの中で最も望ましくない事象が生起した最悪ケースにおけるシステムの性能評価を行うことは，スーパバイザ設計における重要な関心事といえる．言い換えると，最悪ケースに陥った場合における性能を評価し，このときの評価値を最大にするようにシステムの振舞いを制御することが必要となる．

本論文では，対象とする離散事象システムを確率的な遷移を持った確率離散事象システムとしてモデル化し，制御パターンに含まれる事象の中で，最も望ましくない事象が生起していった場合を最悪ケースととらえ，各状態における制御パターンの与え方に対して，最悪ケースを考慮した新たな評価値を導入する．そして，強化学習を用いてこの評価値を最大にする制御パターンの与え方をスーパバイザに学習させる．制御パターンに含まれる事象すべてに対しての期待報酬という形ではなく，選択した制御パターンの中に含まれる事象の中で，最も低い

* 原稿受付　2009 年 9 月 7 日

† 摂南大学大学院　工学研究科　Graduate School of Engineering, Setsunan University; 17–8 Ikeda-nakamachi, Neyagawa, Osaka 572–8508, JAPAN

Key Words: discrete event systems, supervisory control, optimal control, reinforcement learning.

期待報酬を持つ事象に基づいた新たな評価値となっている．提案手法を用いることによって，学習を通して最悪ケースにおける報酬の期待値を最大とする制御パターンを選択するスーパバイザを獲得できる．

以下，**2.** ではスーパバイザ制御と強化学習について述べ，**3.** では導入する評価関数について述べる．**4.** では本論文で提案する学習アルゴリズムを示し，**5.** では計算機実験により提案アルゴリズムの有効性を示す．**6.** では提案アルゴリズムについての考察を述べ，最後に **7.** でまとめと今後の課題を述べる．

2. スーパバイザ制御と強化学習

2.1 スーパバイザ制御とオートマトン表現

離散事象システムの各状態において，制御パターンを作り出す制御器をスーパバイザとよぶ [2]．制御パターンとは，各状態で生起を許可する事象の集合である．このとき事象については，スーパバイザによって事象の生起を禁止できる可制御事象と，禁止できない不可制御事象とに分けられるとする．不可制御事象は，事象の生起を禁止することができないので制御パターンの中に常に含まれる．スーパバイザ制御の基本的な枠組みは，次のようになる．

(1) スーパバイザは制御パターンを選択し，制御対象である離散事象システムに提示する．
(2) 離散事象システムは，提示された制御パターンの中から事象を選択して生起させ，新たな状態に遷移する．
(3) スーパバイザは，制御対象が選択した事象の生起を観測し，再び制御パターンを提示する．

このサイクルを繰り返しながら，論理的に与えられた制御仕様を満たすように制御パターンを切り替えていく．

本論文では，対象とする離散事象システムを確率的な遷移を持った確率離散事象システム (probabilistic discrete event systems) を用いて定義する [11]．確率離散事象システム G_P のモデルとしてオートマトン表現

$$G_P = \langle X,\ \Sigma,\ P,\ x_0,\ X_m \rangle \tag{1}$$

を考える．ここで，X は状態の有限集合，Σ は事象の有限集合，$P : X \times \Sigma \times X \to [0,1]$ は状態遷移確率関数，$x_0 \in X$ は初期状態，$X_m \subseteq X$ は受理状態の有限集合を表す．また，事象については，可制御事象の集合 Σ_c と不可制御事象の集合 Σ_{uc} とに分けられるとする．すなわち，$\Sigma = \Sigma_c \cup \Sigma_{uc}$，$\Sigma_c \cap \Sigma_{uc} = \emptyset$ とする．空事象 ϵ を含み，Σ からなるすべての事象列の集合を Σ^* で表す．また，システムで実行可能な状態遷移を $(x,\sigma,x') \in X \times \Sigma \times X$ として定義しておく．(x,σ,x') は，状態 x で事象 σ が生起し状態 x' へ遷移することを表している．状態遷移確率関数 P により，ある事象が生起した時のシステムの次の状態は，決定的ではなく確率的に定められる．なお，本論文では，スーパバイザはシステムの状態遷移については既知であるが，状態遷移確率関数については未知であるとする．状態遷移確率関数 P は，以下のような性質を持つ．

$$\begin{aligned} &\forall x \in X, \forall \sigma \in \Sigma, \\ &\sum_{\sigma \in \Sigma} \sum_{x' \in X} P(x,\sigma,x') = 1 \\ &P(x,\sigma,x') = 0 \quad \text{if } (x,\sigma,x') \text{ is undefined} \end{aligned} \tag{2}$$

また，ある状態 x でスーパバイザが選択できる制御パターンの集合を $\Pi(x) \subseteq 2^{\Sigma}$ で表す．

2.2 強化学習

強化学習 [8] は，エージェントとよばれる学習者が，環境から与えられる情報を基に，各状態で評価値を最大にする行動を学習する手法である．多くの強化学習では，対象とするシステムがマルコフ決定過程 (MDP; Markov Decision Process) でなければならないが，近似的にマルコフ性が成り立つと考えて強化学習の手法を適用する場合も多い．代表的な手法の一つである方策オフ型の Q-learning について述べる．

Q-learning では，Q 値とよばれる状態行動対の評価値を更新する．状態 x で行動 a を選択し，状態 x' に遷移したときに，報酬 r を受け取ったときの Q 値は，以下のように更新される．

$$\begin{aligned} Q(x,a) \leftarrow\ & Q(x,a) \\ & + \alpha \Big[r + \gamma \max_{a'} Q(x',a') - Q(x,a) \Big] \end{aligned} \tag{3}$$

$Q(x,a)$ は，状態 x で行動 a を選択したときの期待収益 (以後に獲得する割引報酬の総和の期待値) の推定値を表す．α は学習率を表し $(0 < \alpha < 1)$，γ は報酬の割引率を表す $(0 \le \gamma \le 1)$．

3. 制御パターンに対する評価関数の導入

スーパバイザ制御問題を強化学習の枠組みでとらえると，学習者をスーパバイザ，環境をスーパバイザによって制御される離散事象システムととらえることができる．このとき，次の Bellman 最適方程式が成り立つ [9]．

$$\begin{aligned} Q^*(x,\pi) = \sum_{x' \in X} \Bigg(& \sum_{\sigma \in \pi} P_c(x,\pi,\sigma) P(x,\sigma,x') \\ & \Big[R^*(x,\pi,\sigma,x') + \gamma \max_{\pi' \in \Pi(x')} Q^*(x',\pi') \Big] \Bigg) \end{aligned} \tag{4}$$

各記号の意味は，以下の通りである．

- $Q^*(x,\pi)$: 状態 x で制御パターン $\pi \in \Pi(x)$ を選択し，以後は各状態で最大の評価値 Q^* を持つ制御パターンを選択するときの期待収益．
- $P_c(x,\pi,\sigma)$: 状態 x で制御パターン π を選択したとき，制御対象によって事象 $\sigma \in \pi$ が選択される確率．
- $R^*(x,\pi,\sigma,x')$: 状態 x で制御パターン π を選択し，制御パターン π に含まれる事象 σ が制御対象に選択され，状態 x' に遷移したときに受け取る報酬の

期待値.

- γ：報酬の割引率.

つぎに，報酬の期待値 $R^*(x,\pi,\sigma,x')$ については，次のような構造を持つとする．

$$R^*(x,\pi,\sigma,x') = R_1^*(x,\pi) + R_2^*(x,\sigma,x') \tag{5}$$

$R_1^*(x,\pi)$ は，状態 x で制御パターン π を選択したときに受け取る報酬の期待値を表す．$R_2^*(x,\sigma,x')$ は，状態 x で事象 σ を選択し，状態 x' に遷移したときに受け取る報酬の期待値を表す．直観的には，$R_1^*(x,\pi)$ は制御パターンを与えるときに，制御パターンに含まれない事象を禁止したことに対するコストを表し，$R_2^*(x,\sigma,x')$ は事象が生起したことに伴うコストや，タスクの完了など何らかの状態に到達したことに対する報酬を表すと考えられる．以上より (4) 式は，

$$\begin{aligned} Q^*(x,\pi) &= \sum_{x'\in X}\left(\sum_{\sigma\in\pi} P_c(x,\pi,\sigma)P(x,\sigma,x')\right) \\ &\quad \left[R_1^*(x,\pi) + R_2^*(x,\sigma,x') + \gamma \max_{\pi'\in\Pi(x')} Q^*(x',\pi')\right] \\ &= R_1^*(x,\pi) + \sum_{\sigma\in\pi} P_c(x,\pi,\sigma)\left(\sum_{x'\in X} P(x,\sigma,x')\right. \\ &\quad \left.\left[R_2^*(x,\sigma,x') + \gamma \max_{\pi'\in\Pi(x')} Q^*(x',\pi')\right]\right) \end{aligned} \tag{6}$$

と変形できる．

ここで，(6) 式に示される Bellman 最適方程式は，制御パターンに対する期待報酬に基づいて構成されている．しかし，期待報酬に基づき学習させるだけでは，生起確率としては大きくないが，評価の低い事象が生起する可能性が考えられる．元々のスーパバイザ制御は，保守的な制御の枠組みであり，システムで生起しうるどのような事象列に対しても与えられた制御仕様を満たすように制御パターンを決定していた．そこで，最悪ケースを考慮した制御を行うために，提示した制御パターンに含まれる事象の中で，最も生起が望ましくない事象が選択される場合を考えるとする．そのため，(6) 式を変更し，最悪ケースを考慮した新たな評価値 $U^*(x,\pi)$ を以下のように定義する [12].

$$\begin{aligned} U^*(x,\pi) &= R_1^*(x,\pi) + \sum_{\sigma\in\pi} P_c(x,\pi,\sigma) \\ &\quad \left[\min_{\sigma\in\pi}\left(\sum_{x'\in X} P(x,\sigma,x')\right.\right. \\ &\quad \left.\left.\left[R_2^*(x,\sigma,x') + \gamma \max_{\pi'\in\Pi(x')} U^*(x',\pi')\right]\right)\right] \\ &= R_1^*(x,\pi) + \min_{\sigma\in\pi} T^*(x,\sigma) \end{aligned} \tag{7}$$

ただし，

$$T^*(x,\sigma) = \sum_{x'\in X} P(x,\sigma,x')$$

$$\left[R_2^*(x,\sigma,x') + \gamma \max_{\pi'\in\Pi(x')} U^*(x',\pi')\right] \tag{8}$$

である．ここで，$U^*(x,\pi)$ は，状態 x で制御パターン π を選択したことに対する評価値を表す．$T^*(x,\sigma)$ は，状態 x で事象 σ が生起し，以後は，各状態で最大の評価値 U^* を持つ制御パターンを選択するときの期待収益を表す.

評価値 $U^*(x,\pi)$ は，スーパバイザ側が起こした行動 (制御パターン) に対しての即時の報酬の期待値 $R_1^*(x,\pi)$, 制御対象側が起こした行動 (生起事象) に対しての期待収益 $T^*(x,\sigma)$ から構成されている．言い換えると，与えた制御パターンの中に含まれる事象のうち，最も低い期待収益を持つ事象が生起したとして，$\min_{\sigma\in\pi} T^*(x,\sigma)$ と $R_1^*(x,\pi)$ の和により評価値 $U^*(x,\pi)$ を定めている．そのため，制御パターンに対する期待報酬ではなく，最悪ケースをとるように (7) 式を変更しており，(6) 式と比較して (7) 式では選択確率 $P_c(x,\pi,\sigma)$ が消えている．これにより $U^*(x,\pi)$ は，各状態 x で最悪ケースにおける事象の生起と禁止のトレードオフを考慮しての制御パターン π に対する評価値となっている.

4. 学習アルゴリズム

本論文で提案する学習アルゴリズムを Fig. 1 に示す. Fig. 1 において，一つのエピソード (episode) は初期状態 x_0 から始まる状態遷移の系列を表し，制御対象がこれ以上遷移できない状態に到達するか，一定回数の状態遷移を行った場合に終了する．後者の終了条件は，ライブロックにより一部の状態だけしか訪問できず，全状態に対して十分な学習を行えない可能性を除くためである．また，エピソードの一つのステップ (step) は，制御対象の 1 回の状態遷移と，報酬の獲得とスーパバイザによる評価値の更新から構成される．学習の目的は，制御パターンの与え方をスーパバイザに学習させることである．学習者であるスーパバイザは，ある状態 $x\in X$ において，評価値 U^* の推定値である評価値 U に基づき制御パターン $\pi\in\Pi(x)$ を選択し，制御対象に提示する．制御パターンの選択にはルーレット選択や Boltzman 選択などが考えられるが，本論文では ϵ-greedy 選択を用いた．ϵ-greedy 選択では，確率 ϵ でランダムな制御パターンを，確率 $1-\epsilon$ で評価値 U が最大となる制御パターン π を選択する．制御対象は，制御パターンの中から生起させる事象 $\sigma\in\pi$ を選択する．この選択にはスーパバイザは干渉できない．その後，スーパバイザは生起事象 σ を観測し報酬 r_1，r_2 を受け取る．ただし，r_1 は制御パターン π に対する報酬，r_2 は生起事象 σ に対する報酬である．報酬 r_1，r_2 は，それぞれ $R_1^*(x,\pi), R_2^*(x,\sigma,x')$ に基づき与えられる.

学習アルゴリズムでは，Q-learning による更新式と同様にして学習させている．(7) 式より，評価値 U^* は，

1. Initialize $U(x,\pi)$, $T(x,\sigma)$ and $R_1(x,\pi)$ at each state.
2. Repeat (for each episode):
  (a) $x \leftarrow$ *initial state* x_0.
  (b) Repeat for each step of episode.
    i. Select a control pattern $\pi \in \Pi(x)$ based on the U values by the supervisor.
    ii. Observe the occurrence of event $\sigma \in \pi$ and state transition (x,σ,x') in the G_P.
    iii. Acquire rewards r_1 and r_2.
    iv. Update $T(x,\sigma)$ and $R_1(x,\pi)$:

$$R_1(x,\pi) \leftarrow R_1(x,\pi)+\alpha[r_1-R_1(x,\pi)]$$
$$T(x,\sigma) \leftarrow T(x,\sigma)+\beta[r_2 + \gamma \max_{\pi' \in \Pi(x')} U(x',\pi')-T(x,\sigma)]$$

    v. Update $U(x,\pi)$:

$$\text{For all } \pi' \in \Pi(x) \text{ s.t. } \pi' \cap \pi \neq \emptyset$$
$$U(x,\pi') \leftarrow R_1(x,\pi')+\min_{\sigma \in \pi'} T(x,\sigma)$$

    vi. $x \leftarrow x'$.

Fig. 1 The proposed algorithm

スーパバイザ側が起こした行動 (制御パターン π) に対しての即時の報酬の期待値 R_1^* と，制御対象側が起こした行動 (生起事象 σ) に対しての期待収益 T^* を用いて求めることができる．そこで，R_1^* と T^* の推定値をそれぞれ R_1 と T で表すとし，観測した情報から R_1 と T を更新する．

$$R_1(x,\pi) \leftarrow R_1(x,\pi)+\alpha[r_1-R_1(x,\pi)] \quad (9)$$
$$T(x,\sigma) \leftarrow T(x,\sigma)+\beta[r_2 + \gamma \max_{\pi' \in \Pi(x')} U(x',\pi')-T(x,\sigma)] \quad (10)$$

ここで，α，β は学習率を表す．また，すべての事象を禁止する制御パターンが選択された場合，事象 ϵ が生起したとし報酬を受けとり，再度制御パターンを選択する．つぎに，(7) 式に基づき評価値 U を更新するが，R_1 と T は複数の U 値に影響を与えるので，R_1 と T を用いて，実際に選択した制御パターンのみではなく，π に含まれる事象を含む全制御パターンに対し同時に評価値 U の更新を行うことができる．すなわち，

$$\text{For all } \pi' \in \Pi(x) \text{ s.t. } \pi' \cap \pi \neq \emptyset$$
$$U(x,\pi') \leftarrow R_1(x,\pi')+\min_{\sigma \in \pi'} T(x,\sigma) \quad (11)$$

として複数の評価値 U の同時更新による学習効率の向上が期待できる．

5. 計算機実験

2 種類の例題に対して，計算機実験を行った．なお，スーパバイザは，システムの状態遷移 (x,σ,x') については観測できるが，状態遷移確率関数 P，報酬の期待値 R_1^*，R_2^* については未知であるとする．

5.1 例題 1

最初の例題として，文献 [3] で取り上げられた問題を考える．この問題は，決定性のオートマトンでモデル化されており，システムの状態遷移は確率的でなく決定的に定まる．この問題の状態遷移図を Fig. 2 に示す．Fig. 2 で各事象に数字を記しているが，前者は事象の生起に対するコストを表し，後者は禁止に対するコストを表している．なお，事象 e は，不可制御事象であるため禁止することができない．

学習の設定として，制御パターンの与え方に対する報酬 r_1 は，制御パターンに含まれない事象の禁止に対するコストの総和に -1 を乗じたものと設定し，事象の生起に対する報酬 r_2 は，生起した事象のコストに -1 を乗じたものと設定した．評価値 U の初期値はすべて 0 とし，学習率 α，β はいずれも 0.1 を用いた．また，文献 [3] の条件と合わせるために，報酬の割引率 γ=1 と設定した．制御パターンの選択には，$\epsilon = 0.1$ の ϵ-greedy 選択を用いた．1 エピソードは，初期状態から始まり，30 ステップ経過することで終了とした．各状態で最悪ケースにおける評価値が最大となる制御パターンを Table 1 に示す．ここで，状態 5 における制御パターン $\pi = \{\epsilon\}$ はすべての事象を禁止し，その場にとどまる制御パターンを表している．また，制御対象が制御パターンの中から各事象を選択する際の確率は，等確率に設定した．Fig. 3 は，横軸にエピソードを，縦軸にスーパバイザが学習した最

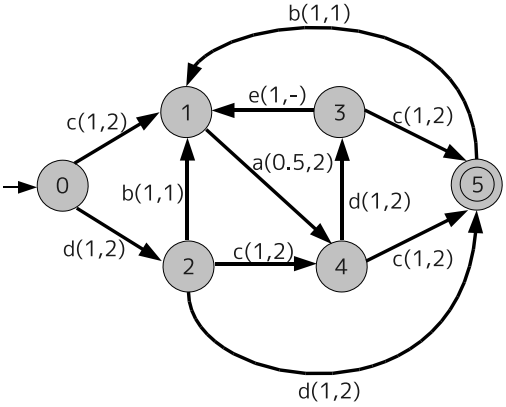

Fig. 2 The state transition diagram of the example 1

Table 1 The optimal control patterns of the example 1

State	Optimal control pattern
0	$\{c,d\}$
1	$\{a\}$
2	$\{d\}$
3	$\{c,e\}$
4	$\{c\}$
5	$\{\epsilon\}$

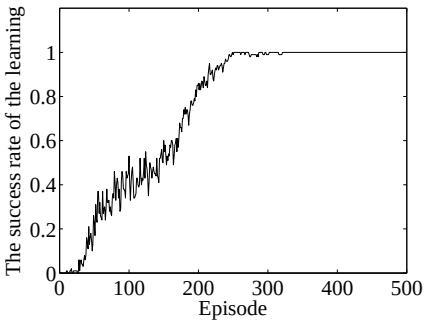

Fig. 3 Learning result of the example 1

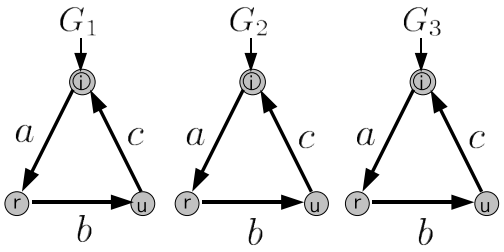

Fig. 4 The state transition diagram of three computers

悪ケースにおける評価値Uを最大にする制御パターンと，Table 1 に示した制御パターンとがすべての状態で一致した割合を示している．この割合は，各パラメータを固定にして 100 回の実験を行った結果の平均である．学習が進むにつれて最適な制御パターンを学習していっているのがわかる．

5.2 例題 2

次の例題として，文献 [2] で取り上げられた非同期に動作する複数のコンピュータが 1 台のプリンタへアクセスする問題を考える．文献 [2] では，2 台のコンピュータを合成オートマトンでモデル化していた．本論文ではコンピュータが 3 台の場合を考える．各コンピュータ G_1, G_2, G_3 の状態遷移図を Fig. 4 に示す．Fig. 4 において，各状態はプリンタに対するそれぞれのコンピュータの状態を表しており，状態 i は “IDLE”(空き状態)，状態 r は “REQUEST”(待ち状態)，状態 u は “USE”(使用中) を表している．3 台のコンピュータの振舞いを表現するために，合成オートマトン $G_1||G_2||G_3$ を生成し制御対象とした．生成した合成オートマトン $G_1||G_2||G_3$ の状態遷移図を Fig. 5 に示す．Fig. 5 において，各状態はそれぞれ順に G_1, G_2, G_3 の状態を表している．たとえば，初期状態 “i,i,i” はそれぞれのコンピュータが “IDLE” の状態にあることを意味する．また，システムの状態遷移は決定的ではなく確率的に起こるとする．たとえば，初期状態 “i,i,i” では事象 a が生起するが，状態 “r,i,i”，“i,r,i”，“i,i,r” のいずれに遷移するかは確率的に定まるとする．学習の目的は，事象の生起や禁止にいくらかのコストがかかる場合に，各状態での最悪ケースにおける評価値を最大にする制御パターンをスーパバイザに学習させることである．

学習の設定として，制御パターンに対する報酬 r_1 は，制御パターンに含まれない事象を禁止したことによるコストの総和に設定した．ここで，各事象の禁止コストは，-1 から -10 の範囲でランダムに設定した．また，生起事象に対する報酬 r_2 は，-1 から -10 の範囲でランダムに設定した．状態遷移確率関数 P は，(2) 式の条件に収まるように，0 から 1 の範囲でランダムに設定した．評価値 U の初期値はすべて 0 とし，学習率 α，β はいずれも 0.1 を用いた．報酬の割引率 γ=0.9 とし，制御パターンの選択には，$\epsilon = 0.1$ の ϵ-greedy 選択を用いた．1 エピソードは，初期状態から始まり 30 ステップ経過することで終了するとした．各状態で最悪ケースにおける評価値が最大となる制御パターンを Table 2 に示す．なお，この制御パターンにより初期状態から遷移しなくなる状態は除いている．また，制御対象が制御パターンの中から各事象を選択する際の確率は，等確率に設定した．

Table 2 The optimal control patterns of the example 2

State	Optimal control pattern
“i,i,i”	$\{a\}$
“i,i,r”	$\{b\}$
“i,i,u”	$\{c\}$
“i,r,i”	$\{b\}$
“i,u,i”	$\{c\}$
“r,i,i”	$\{b\}$
“u,i,i”	$\{c\}$

実環境では，たとえば，事象の生起は観測できても，その生起確率まではわかっていない場合や，センサの乱れやモデル化できていない要素などにより受け取る報酬にはぶれがあるという状況が考えられる．そこで，報酬 r_1，r_2 についてそれぞれ観測ノイズがあるとして，真値を平均とし，分散 2.0 とした正規分布にしたがって与えるとした．Fig. 6 は，横軸にエピソードを，縦軸にスーパバイザが学習した最悪ケースにおける評価値Uを最大にする制御パターンと，Table 2 に示した制御パターンとがすべての状態で一致した割合を示している．この割合は，各パラメータを固定にして 100 回の実験を行った結果の平均である．Fig. 6 より，最悪ケースにおける評価値を最大にする制御パターンを学習していっていることがわかる．一方，評価値を同時更新した場合と，選択した制御パターンに対する評価値のみを更新した場合とを比較すると，同時更新の方が学習の立ち上がりが早くなっているのがわかる．

6. 考察

本論文では，最悪ケースを考慮するために新たな評価関数を導入した．制御パターンにより生起しうる事象す

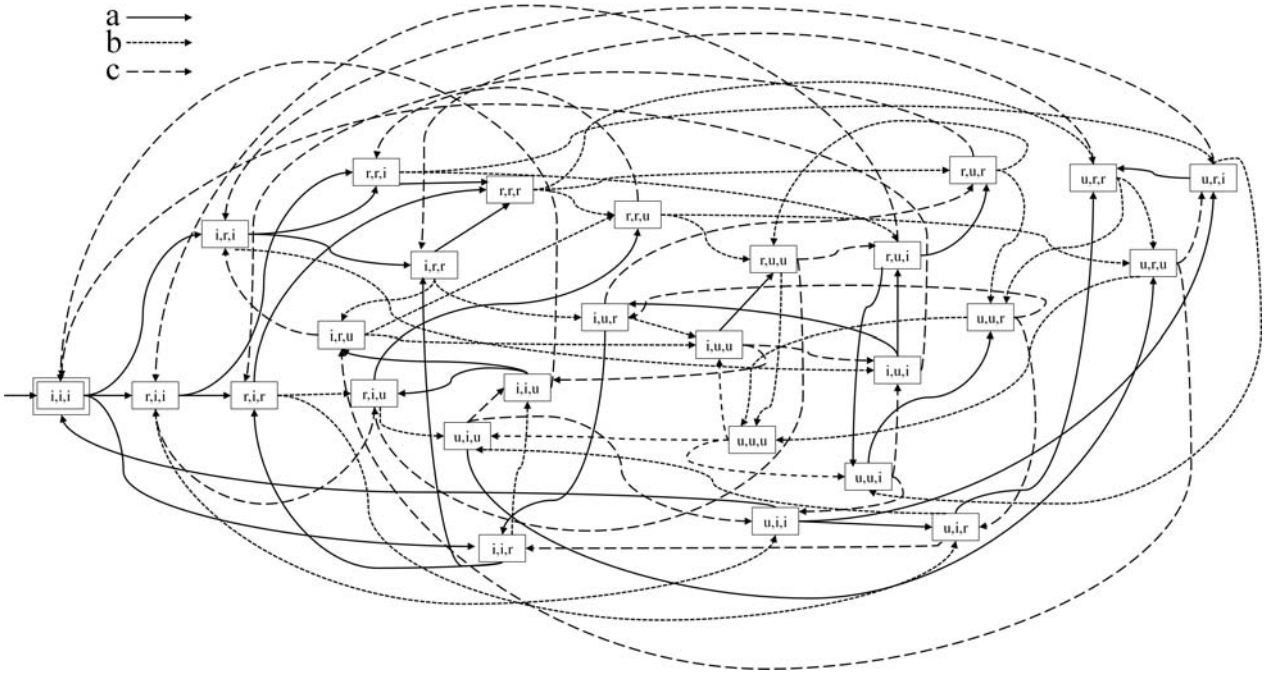

Fig. 5 The state transition diagram of automaton $G_1||G_2||G_3$

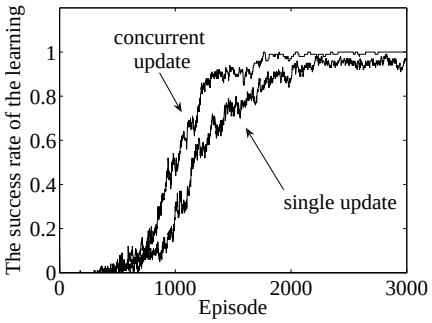

Fig. 6 Learning result under the rewards with the noises

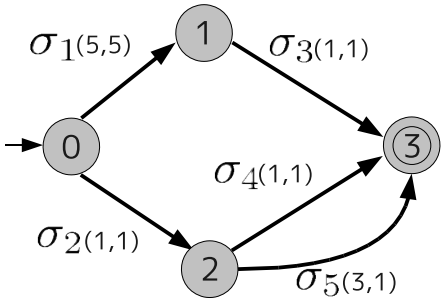

Fig. 7 The state transition diagram of system G_4

べてに対する期待報酬という形ではなく，選択した制御パターンの中に含まれる事象の中で，最も低い期待報酬を持つ事象に基づいた評価値となっている．スーパバイザ制御の一つの特徴として，システムが可制御となる最大の生成言語である最大可制御部分言語がある．しかし，これを求めるには，制御対象や制御仕様を形式言語やオートマトンで厳密に表現しなければならなかった．提案手法では，強化学習を用いているため事象の生起や禁止の評価を報酬といった形で与えることで，報酬を基にして制御仕様の詳細を求めることができる．ただし，提示した制御パターンの中に含まれる事象の中で，最も低い期待報酬を持つ事象の生起確率が小さいときに，学習が遅くなる場合が考えられる．

本論文において学習されるスーパバイザについては，文献 [3] で述べられているものと同様の性質が存在する．例として，Fig. 7 のシステム G_4 を考える．Fig. 7 で各事象に数字を記しているが，前者は生起事象に対するコスト，後者は禁止に対するコストを表している．学習の設定を前節 **5.1** と同様に行い提案手法で学習させた場合，評価値 U を最大とする制御パターンは “ 状態 2 では事象 σ_5 を禁止し，他の状態ではすべての事象の生起を許容する” となった．このときの評価値 U は，$U(0,\{\sigma_1,\sigma_2\})=-6$, $U(1,\{\sigma_3\})=-1$, $U(2,\{\sigma_4\})=-2$ となる．また，このときのスーパバイザを S_1 とすると，スーパバイザ S_1 によって制御される制御対象 G_4 の受理言語は $L_m(G_4/S_1)=\{\sigma_1\sigma_3,\ \sigma_2\sigma_4\}$ となる．

一方，状態 2 で σ_4 と σ_5 の両方の事象を許容した場合でも，状態 0 での評価値 U は変化しない．これは，事象 σ_1 の生起による報酬が -5 であり，状態 2 で事象 σ_5 の生起を許容した場合でも，σ_1 の生起が最悪ケースと考えられ，状態 0 での評価値 U に影響を与えないからである．このときの評価値 U は，$U(0,\{\sigma_1,\sigma_2\})=-6$, $U(1,\{\sigma_3\})=-1$, $U(2,\{\sigma_4,\sigma_5\})=-3$ となる．また，このときのスーパバイザを S_2 とすると，受理言語は $L_m(G_4/S_2)=\{\sigma_1\sigma_3,\ \sigma_2\sigma_4,\ \sigma_2\sigma_5\}$ となり，$L_m(G_4/S_1)$

より集合として大きくなる．両者を比較すると評価値としては同じであり，生成言語の大きさという点から，後者の方が望ましいと考えることもできる．一方，提案手法は，動的計画法に基づく強化学習を用いているため，各状態を初期状態として構成した部分問題の解についても最適となる制御パターンを学習する手法となっており，この点で優れているといえる．

7. おわりに

本論文では，離散事象システムの最適スーパバイザ制御問題について，制御パターンに対して最悪ケースに基づく新たな評価関数を導入した．制御対象を確率離散事象システムとしてモデル化し，スーパバイザが用いる評価値に，最も望ましくないシステムの振舞いを考慮することで，最悪ケースにおける評価値を最大化する制御パターンを学習することができる．提案手法では，スーパバイザの設計に強化学習を用いているため，コスト情報や詳細な仕様の記述が事前にはわからない場合や，観測ノイズにより正確に得られない場合においても柔軟に対応できる．

今後の課題としては，提案手法における評価値を言語測度 [5] として導入できるかを検討することが考えられる．また，考察で述べたスーパバイザの性質 [3] に関し，生成言語の点で最大となるスーパバイザを求められるアルゴリズムの検討が必要である．

参 考 文 献

[1] C. G. Cassandas and S. Lafortune: *Introduction to Discrete Event Systems*, Kluwer Academic Publishers (1999)

[2] P. J. Ramadge and W. M. Wonham: Supervisory control of a class of discrete event processes; *SIAM J. Control Optim.*, Vol. 25, No. 1, pp.206-230 (1987)

[3] R. Sengupta and S. Lafortune: An optimal control theory for discrete event systems; *SIAM J. Control Optim.*, Vol. 36, No. 2, pp.488-541 (1998)

[4] H. Marchand, O. Boivineau and S. Lafortune: On optimal control of a class of partially observed discrete event systems; *Automatica*, Vol. 38, pp.1935-1943 (2002)

[5] X. Wang and A. Ray: A language measure for performance evaluation of discrete-event supervisory control systems; *Applied Mathematical Modelling*, Vol. 28, pp.817-833 (2004)

[6] V. Phoha, A. Nadgar, A. Ray, S. Phoha and V. Jain: Supervisory control of software systems; *IEEE Transactions on Computers*, Vol. 53, No. 9, pp.1187-1199 (2004)

[7] X. Wang, A. Ray, P. Lee and J. Fu: Optimal control of robot behaviour using language measure; *International Journal of Vehicle Autonomous Systems*, Vol. 2, Nos. 3/4, pp.147-167 (2004)

[8] R. S. Sutton, 三上, 階川（共訳）: 強化学習, 森北出版 (2000)

[9] 山崎, 潮: 強化学習を用いた離散事象システムのスーパバイザ制御；システム制御情報学会論文誌, Vol. 16, No. 3, pp. 118-124 (2003)

[10] 谷口, 山崎, 潮: 言語測度に基づいた最適スーパバイザの強化学習；システム制御情報学会論文誌, Vol. 18, No. 12, pp. 433-439 (2005)

[11] V. K. Gang, R. Kumar and S. I.Marcus: A probabilistic language formalism for stochastic discrete-event systems; *IEEE Transactions on Automatic Control*, Vol. 44, No. 2, pp.280-293 (1999)

[12] 梶原, 山崎: 最悪ケースを考慮した最適スーパバイザの強化学習；電子情報通信学会技術研究報告, Vol. 108, No. 415, pp. 45-50 (2009)

著 者 略 歴

梶原（かじわら） 弘治（こうじ）

1983 年 7 月 2 日生．2009 年 3 月摂南大学大学院工学研究科機械・システム工学専攻博士前期課程修了．現在，同大学院創生工学専攻博士後期課程在学中．スーパバイザ制御の研究に従事．電子情報通信学会学生会員．

山崎（やまさき） 達志（たつし） (正会員)

1975 年 1 月 14 日生．1999 年 3 月大阪大学大学院基礎工学研究科システム人間系専攻博士前期課程修了．2002 年 11 月より関西学院大学理工学部契約助手．2003 年 3 月大阪大学大学院基礎工学研究科システム人間系専攻博士後期課程修了．2006 年 4 月より摂南大学工学部講師となり現在に至る．計算生態モデル，強化学習，離散事象システムなどの研究に従事．博士 (工学)．電子情報通信学会，計測自動制御学会，IEEE などの会員．