強化学習による多足歩行ロボットの適応的歩様獲得に関する研究*

三上 貞芳^{*1}, 田野 浩明^{*2}, 嘉数 侑昇^{*1}Acquiring Adaptive Gaits For Many-Legged Robots
by Reinforcement Learning

Sadayoshi MIKAMI, Hiroaki TANO and Yukinori KAKAZU

A legged robot should be able to walk through an unknown environment and avoid unpredictable breakage. To realize this capability, the robot should learn to control its legs without being precisely informed of its internal and external environments. Since the worst condition is the situation where no environmental model is given and where only sensory input that informs the success or failure of walking is available, it is necessary to establish a learning mechanism to developing rules for walking under this situation. This paper proposes a reinforcement-learning-based method to realize this adaptive gait acquisition with minimal information. It was shown by computer simulations that the 6-and the 8-legged wall-climbing robots could successfully establish their gaits without any initial knowledge. It was also demonstrated that the simulated 8-legged robot could establish a new gait after one of its legs was broken.

Key Words: Robot, Learning, Artificial Intelligence, Legged Robot, Gait, Reinforcement Learning, Q-Learning

1. 緒 言

脚による移動は、不整地など他の移動機構では進行不能に陥るような環境において確実に移動することを必要とする応用に有効であるとされ、近年特に資源探査や補修作業用ロボットなど、未知の不整地環境を歩行するロボットの移動方式として有望視されている⁽¹⁾⁽²⁾。この場合、目的は予期せぬ故障に際しても移動を継続することを最優先させることであり、例えばロボット本体の転倒、脚への過大な負荷などの危険を考慮から除外しても、現在の状況で移動可能な歩行パターンを発見することが必要とされる。

このような適応的歩行獲得に関する研究としては、たとえば Brooks らによる包摂アーキテクチャを用いた多脚昆虫形ロボットの適応的歩行に関する研究などがあげられる⁽²⁾。しかし、これは設計者が Behavior とよばれる行動のモジュールをあらかじめ適切に用意することを前提としており、広い範囲の状況に対応できるが、その適応能力が設計者の設計スキルに大きく依

存する点が問題としてあげられている。

これに対して、ロボットがあたえられた移動目的を継続達成していることを示すセンサ(例えば前進センサ)が正しく動作していると仮定するならば、この目的達成を示すセンサ信号が得られる頻度を高くすることを目標として、試行錯誤により自動的に歩行パターンを発見するアプローチが考えられる。この場合ロボット、環境に対する先験的な知識またはモデル、これらに基づいた設計スキルなどを必要としないため、予期しない状況に対する対応能力は優れたものが期待できる。しかし、歩行パターンの組合せは膨大であるゆえ、一様乱数的な試行錯誤でこれが発見することは実用的ではない。

一つの解決策は、過去の試行の結果を反映させることにより探索の効率を改善することである。このような手法は最近になり、強化学習と呼ばれる新しい学習法として提案され、盛んに研究が進められている⁽³⁾⁽⁴⁾。強化学習法は環境の状態の観測と、行動の結果の観測の2種類のセンサ情報のみを利用して、結果の合目的度(報酬, profit)の期待値を最大化するような手法であり、環境のモデル、教師モデルを利用しない点、逐次に学習が進む点は、多足歩行ロボットの適応歩行の目的に適している。しかし従来、強化学習をこ

* 原稿受付 平成6年1月27日。

^{*1} 正員, 北海道大学工学部 (〒060 札幌市北区北13条西8丁目)。^{*2} 学生員, 北海道大学大学院。

のような歩様獲得に用いた例はない。

そこで、この論文では以上のような適応的な歩様獲得を、強化学習の一つである Q-Learning⁽⁵⁾ の導入により実現する一手法を提案する。

これにより故障などの予期しない状況に対しても、移動目的が達成されていることを示すセンサ情報が正しく入力されている限り、ロボット自身が可能な歩様を探索、発見し移動を続けられるような適応機能を多足歩行ロボットに付与することが可能となる。さらに、ロボットの機構、外界に関するモデルを必要とせずに、行動目的のセンサ情報を含めたロボットの内界、外界センサからのリアルタイムの情報のみを利用する点は、機構、歩行環境を選ばずに幅広いロボットに適用できる利点を有している。

計算機実験では、移動目的として前進を取上げ、脚の故障に対しても、それを明示的に知らされなくとも、新たな歩様を自動的に獲得できることが実際に確認できている。

2. 適応的歩様獲得問題

脚の故障なども含めて、ロボットの機構、歩行環境が、時間とともに動的に変化する場合を想定する。さらに、それら機構、環境の数理モデルが入手できない状況を仮定する。ここで扱う適応的歩様獲得問題とは、以上のような状況で、転倒などダメージを起こさずに継続的に移動目的を達成する歩様を、ロボット自らが探索、発見する問題であると定義する。本論文を通して歩様獲得とは以上の問題を指すものとする。

3. 最適行動の探索

歩様獲得にはモデルを利用せずに適切な行動を獲得する最適行動の探索が必要となる。この枠組は以下のように定式化することができる⁽³⁾。まず、次の記号を定義する。

- 時刻 t で観測された状態を $x(t)$ とする、
- 状態 $x(t)$ で可能な行動は、有限個の要素をもつ集合 $A(t)=\{a_i\}$ とする、
- t で行動 a を実行した結果、遷移した状態を $y(x, a)$ とする、
- 関数 $E(y) \in \mathfrak{R}$ を、状態 y の報酬(profit)関数と呼び、 $E(y') > E(y)$ は y' が y より適切な行動であることを意味する、
- 時刻 t における最適行動を $\hat{a}(t)$ とする。このとき最適行動 $\hat{a}(t)$ とは、式(1)を満たすような $\hat{a}(t)$ である。

$$\max_{\hat{a}(t)} V[x(t), \hat{a}(t)], \dots\dots\dots (1)$$

ここで、

$$\hat{V}[x(t), \hat{a}(t)] = \sum_{k=0}^{\infty} \gamma_k \cdot E[y(x(t+k), \hat{a}(t+k))] \dots\dots\dots (2)$$

ただし γ_k は割引率と呼ぶ 0 から 1 までの定数を示す。式(1)、(2)は、時刻 t で $\hat{a}(t)$ を選択した後、引き続き最適行動が選択され続けた場合、将来にわたる割引率付きの報酬の総和 $\hat{V}[x(t), \hat{a}(t)]$ が最大化されるように $\hat{a}(t)$ が選ばれていることを意味する。割引率は将来の報酬の考慮の度合いを与えるもので、例えば $\gamma_0=1$ かつ $\gamma_k=0, k>0$ の場合、直後の報酬のみが考慮される。

- $\hat{V}[x(t), \hat{a}(t)]$ の予測値を $V[x(t), \hat{a}(t)]$ とする、
- 次に最適行動探索の手続きをアルゴリズム式(3)により定義する。

1 : Observe $x(t)$;	}
2 : Evaluate $E[x(t)]$;	
3 : Update $V[x(t-1), a(t-1)]$ by $E(x(t))$
such that $V[x(t-1), a(t-1)] \approx \hat{V}[x(t-1), a(t-1)]$;	
4 : Select $a(t)$ according to $V[x(t), a(t)]$;	
5 : Perform $a(t)$;	
6 : Goto 1;	
- (3)

式(3)の4行は、選択される行動が最適行動になるような操作を目的とする。ここで V 値 $V[x(t), a(t)]$ は報酬の見積りの近似値ゆえ、式(1)に従って最大の V 値をもつ行動を常に選択することは、誤った行動を固定することになり好ましくない。したがって大きい V 値を与える行動を優先して選択するように、確率的揺らぎを加えて行動を選択することが望ましい。これを実現する一例として、各行動 a に対応した V 値 $V[x(t), a]$ から、ガウシアン分布に従った確率

$$Pr[x(t), a] = e^{V(x(t), a)/T} / \sum_{a'} e^{V(x(t), a')/T}, \dots\dots\dots (4)$$

を計算し、この確率に従って行動の一つを選択する機構が考えられている。ここで T は揺らぎの大きさをコントロールする係数で、温度パラメータと呼ばれる。

報酬関数と状態遷移がマルコフ的である場合に、
 $\lim_{t \rightarrow \infty} V[x(t), a(t)] = \hat{V}[x(t), a(t)], \dots\dots\dots (5)$

なるような式(3)3行の更新手続きは強化学習と呼ばれ、多くの種類が提案されている⁽³⁾。

4. 強化学習による脚の歩様獲得

以上の最適行動探索の定式化をロボットの歩様獲得に対応付けることを考える。歩行の目的を報酬関数に

設定することで、式(5)を保証する学習則である強化学習により、適応的歩様獲得問題を解決することができる。

ただし故障も含めてロボットの構造が動的に変化することを考慮しなければならない。このような場合、ロボット全体でただ一つの最適行動探索の機構を置くことは、行動探索機構の故障に対してロボット全体が機能停止する危険性のため、好ましいことではない。そこで各脚が独立したコントローラを有し、これが独立して前述の最適行動探索を行い歩様獲得の目標を達成するようなアーキテクチャを考える。

4・1 例題 以降、図1に示すような仮想的な多脚ロボットの歩様獲得を具体例にとり説明する。この例題での目的は、垂直に立つパイプ内部を、落下せずに上方に歩行移動可能な(前進する)歩様を獲得することに設定した。簡単のためロボット、パイプ内部ともに2次元性であると仮定する。図1に示すとおり、下方向に重力が作用しているとし、脚は十分なコンプライアンスを有し、滑ることなく壁面に接しているものとする。壁面が平坦であることは前提とはしていない。また簡単化のためロボットにかかるモーメントは考慮しないものとする。

これらの問題の設定、および簡単化は後述するシミュレーションの簡単化を目的としている。したがって実機における現実的制約の多くを考慮したものではないが、歩様獲得機構自身には機構に関する知識を可能な限り用いていないため、以下の定式化および結果は実機への実装への基礎的な方法論およびデータを提供

し得るものとする。実機における制約に関しては第6章で議論する。

4・2 行動集合 A 有限種類の行動 $\{a_i\}$ を脚の動作行動に対応させる。このため脚が離散的な3状態 $\{l_i\}$ を遷移するように動作するものとし、状態遷移のトリガ信号のオン・オフをそれぞれ行動 $a=1, a=0$ とする。

具体的には以下のようになる。脚 i の状態を l_i で記述すると、各脚は、遊脚状態 $l_i=0$ 、着地後支持脚状態 $l_i=1$ 、引き上げ前支持脚状態 $l_i=2$ の3状態を順に遷移する(図1)。各状態間遷移は離散時間 δt で行われる。またトリガ信号により状態 x から $\text{mod}(x+1, 3)$ への遷移が開始されるものとする。したがって各脚のコントローラは、時刻 t において脚が状態 $l(t)$ にあるときに、2種類の行動 a に従って、

$$a = \begin{cases} 1; l(t) = x \xrightarrow{\delta t} l(t+1) = \text{mod}(x+1, 3), \\ 0; l(t) = x \xrightarrow{\delta t} l(t+1) = x, \end{cases} \dots\dots\dots (6)$$

なる状態遷移を行う。

実際には状態 $l=1, 2$ で行動 $a=0$ を選択することは、歩行を妨げる誤った行動である。しかし本論では初期知識をもたないロボットから歩行の知識を獲得することが目的であることから、前述の知識は敢えて導入せず、これが試行を通して獲得されることを確認することで歩様獲得能力を検証することを目的としている。行動 a の実行を状態0に限定し、 $l=0 \rightarrow 1 \rightarrow 2$ の遷移が引き続き起きるように設定することで、より現実的な実装が可能であるが、本論ではこれ以上は議論しない。

4・3 観測状態 $x(t)$ 式(2)により行動を決定するためには、 $x(t)$ すなわち各時刻 t での脚の外界の状態を得る必要がある。ここでは他のすべての脚の状態が観測可能と仮定し、脚 i の状態を l_i 、脚数を n とし、

$$x(t) = \sum_{i=1}^n 3^{i-1} \cdot l_i, \dots\dots\dots (7)$$

により状態を与えるものとする。

4・4 報酬関数 $E(y)$ ロボットの歩行の目的は報酬関数 $E(y)$ によって具体的に与えられる。すなわち状態 y が歩行を達成している場合に $E(y)$ を高い値に、逆に転倒など好ましくない状態では $E(y)$ を低い値に設定すればよい

歩行の状態 y は、少なくとも「ロボットが転倒することなく前進した状態」を識別していなければならない

図1 2次元壁(パイプ)内を歩行前進する仮想的多足歩行ロボット

(a) $S_1=1$ for all legs (b) $S_2=1$ for leg 2, 3, 6

図2 シミュレーション環境におけるロボットの外界センサ入力の設定

い。そこで例として図1に示すロボットに対して、各脚独立に次の三つの外界センサを用意する。すなわち、

- S_1 : 胴体が壁面へ接触したことを示すセンサ
- S_2 : 脚に必要な以上の負荷が加わっていることを検出するセンサ
- S_3 : ロボット胴体の垂直方向の移動を検出するセンサ

後述するシミュレーションでは壁面は平坦であると仮定することで、 S_1 , S_2 , S_3 を以下のように{0, 1}の2値信号として簡単に定めている(図2)。

一方の壁面に2脚以上の支持脚があるとする。任意の支持脚の二つの接地点を a , b で示す。もう一方の壁面に少なくとも1脚の支持脚があるとする。この接地点を c とすれば、 a , b , c を壁面に垂直に投影したときに、 $c \leq b \leq a$ となるような脚の組合せが存在するときに、 $S_1=1$ であるとする。

S_2 は、自脚が胴体を駆動しているにもかかわらず、他脚が支持脚状態で停止している場合、またはその逆の場合に出力1を返すものとする。具体的には脚 i の状態、行動をそれぞれ x_i , a_i と記述したときに、

$$S_2 = \begin{cases} 1, & \text{if } (x_i=1) \wedge (a_i=1) \\ & \wedge (\exists j[(x_j=1 \vee x_j=2) \wedge (a_j=0)]) \\ & \text{or if } (x_i=1 \vee x_i=2) \wedge (a_i=0) \\ & \wedge (\exists j[x_j=0] \wedge (a_j=1)); \\ 0, & \text{otherwise.} \end{cases} \quad \dots\dots\dots (8)$$

とする。

移動の検出は、

$$S_3 = \begin{cases} 1, & \text{if } S_1=1 \wedge S_2=0 \wedge \exists j[(x_j=1) \wedge (a_j=1)]; \\ 0, & \text{otherwise,} \end{cases} \quad \dots\dots\dots (9)$$

すなわち歩行が成功した場合に1を返すものとする。

図3 Q-Learning

報酬関数の引き数としての状態 y は、

$$y = (S_1, S_2, S_3), \dots\dots\dots (10)$$

とする。

以上の y のもとに、各脚に対しての歩行の目標を、「転倒することなく前進し、かつ脚が過大な負荷を受けないこと」を表すように、一例として適当な重みにより次のように定義する。

$$E(y) = -0.3S_2 + 0.5S_3 \dots\dots\dots (11)$$

4・5 強化学習

ここで獲得する歩様は静的歩様であるから、故障や環境の変動を考えなければ、次の脚の状態を決定する問題はマルコフ的であり、したがって、意志決定関数 $P_t[x(t)]$ は、各時点での状態数 $|x(t)|$ の種類だけ用意すればよく、次のように書ける。

$$P_t[x(t)] = P_{x(t)} \dots\dots\dots (12)$$

ここで状態数 $|x(t)|$ は、歩様獲得問題では t によらず

$$|x(t)| = 3^n, \dots\dots\dots (13)$$

となる。ただし n はロボットの脚数を示す。したがって8脚ロボットでは全状態数は6561となる。

仮にすべての脚が確率0.5で行動 $a=1$ を選択する(すなわち脚を遷移させる)ならば、6561状態すべてに遭遇するために、約6万回の試行が必要であることがシミュレーションを通じてわかっている。この論文での歩様獲得問題の設定は、他の脚を調停する機構を用いずに、試行のみで式(12)を獲得する問題である。したがって一様乱数による探索手法を採用すると、少なくとも6万回の試行を必要とする場合が考えられる。

このことから歩様獲得問題では、

1. 広い探索空間を効率よく調べることのできるような、試行錯誤に基づく式(12)の獲得アルゴリズム、が必要であることがわかる。

さらに動的環境を考慮した場合には、

2. $x(t)$ の遷移、および対応する $E(y)$ の値に揺らぎが生じる場合にも、式(12)を獲得できること、

3. 十分に学習が行われた環境にその後変化が生じた場合でも、変化を陽に教えなくとも自動的に新たな環境への最適行動を獲得できること、
が必要な条件である。

以上の2, 3を満足するような手法の一つに、Watkinsによって提案されたQ-Learning⁽⁵⁾と呼ばれる強化学習法があげられる。そこで以下ではQ-Learningを用いたコントローラの構成を具体的に示し、かつ前項1の探索空間の問題に関して解決法を議論する。

4・6 Q-Learning Watkinsらにより提案されたQ-learningは、3章に示したような状態観測と行動、その報酬の受取りのサイクルにより、 $V(x, a)$ を $\hat{V}(x, a)$ に近づけるような V 値の更新式を与えている。以降では $V(x, a)$ を $Q(x, a)$ と表し、 Q 値と呼ぶことにする。 Q 値の更新式は次のように定式化されている(図3)。

いま状態 $x(t)=x$ で行動 $a(t)=a$ を行った結果、状態 y に遷移し、報酬関数値 $E(y)=r$ を得たものとする。この場合の Q 値を Q' とすると、 Q' は次のように書ける。

$$Q'(x, a) = r + \gamma \cdot \max_{a'} Q(y, a') \quad \dots\dots\dots (14)$$

したがって、以前の x, a での Q 値 $Q(x, a)$ は式(14)に近づくよう修正されなくてはならない。これは適当な学習係数 $\alpha \in [0, 1]$ を掛けて次のように行われる。

$$Q(x, a) \leftarrow Q(x, a) + \alpha \cdot [Q'(x, a) - Q(x, a)] \quad \dots\dots\dots (15)$$

したがって、式(12)の意志決定関数 P_x は、各行動 a に対する Q 値 $Q(x, a)$ 、 $\forall a$ を持ち、かつ式(4)による確率的行動選択により行動を選択し、式(15)により Q を更新するような動作を行う。

4・7 ニューラルネットワークを用いたQ-Learning Q-Learningでは1試行で通過した状態、行動のパスに関する Q 値が、その試行で得た報酬のみ

ではなく、遷移先の Q 値を含めた値で修正される。したがって遷移先の Q 値が V 値のよい近似値となっていれば、実際の報酬信号 r が、状態 y での真の報酬信号 $E(y)$ に揺らぎ n を伴った $E(y)+n$ で与えられたとしても、式(14)の係数 γ が大きく、 Q 値が n に対して十分大きい場合には、 n の及ぼす揺らぎに敏感に影響されることはない⁽⁵⁾。

したがってQ-Learningは一般に、よい行動の系列から逸脱せずに探索を続けることができる。すなわち4・1節に示す2の条件は満足されている。

一方、Q-Learningでは1試行で一つの Q 値のみが更新されるゆえ、未試行の状態に対しては以前の試行で直接 Q 値が更新されることはない。4・1節で議論したように、歩様獲得問題では状態空間を網羅するために膨大な探索回数を必要とする。これは長い試行期間を経ても Q 値が未知なままである状態が多く存在することを意味する。したがってこのような状態へ遷移した場合には、よい系列からの逸脱が起き得るため、学習時間が長くなる問題点が指摘できる。

そこで未試行の状態に対する Q 値に対しても、その近傍の状態からよい近似を作っておくことで、常に状態空間全体にわたって良好な Q 値を用意できることが望ましい。

適切な近傍の与え方は歩行環境、脚の構造など問題に依存するが、本論文の目的は問題依存な要因をなるべく排除することであるから、ここでは一般的な近傍への Q 値の伝播方法として、 Q 値をニューラルネットワーク上に記憶させる方法を採用する⁽³⁾。ここでは Q 値を3層バックプロパゲーション形ニューラルネットワークに記憶させた、Q-Netと呼ばれるLin⁽⁶⁾の方法を採用する(図4)。

実際には状態 $x(t)$ を2値ベクトルに表現した $\{b_i; i=0, \dots, n\}$ に対して、 n 個の入力素子、 m 個の中間層、1個の出力素子を有するBPネットワークを、行動の種類だけ用意し、入力層のみを共有させる。一つの行動 a に対応するネットワークの出力を、(通常の $[0, 1]$ の範囲をとる出力層の素子を -0.5 オフセットさせることにより) $o_a \in [-0.5, 0.5]$ として、 $o_a = Q(x, a)$ と解釈する。したがって式(15)による Q 値の更新は、次のようになる。

- a に対応するネットワークを選択する。
- 式(14)により、望ましい Q 値である $Q'(x, a)$ を計算する。
- $Q'(x, a)$ を学習教師信号として、適当な学習係数で入力 x に対してバックプロパゲーションを行い、重みの学習を1回行う。

図4 Q-NET

以上の手続きとなる。

計算機シミュレーションでは、 i 脚の状態表現に三つの入力層素子 o_i^0, o_i^1, o_i^2 を割当て、状態 l_i に対して素子 o_i^1 への入力を 1 とし、その他の素子の入力を零としている。

4.8 Q-Net による歩様獲得 歩様獲得の手続きはアルゴリズム式(16)にまとめることができる。 Q 値の引き数である状態と報酬関数の引き数の状態は、例題では別々のセンサ系統から得られている。したがって式(16)ではこれらを分けた記述となっている。また式中下線部分はロボット外部への入出力、それ以外が内部処理を表している。また δt は脚の状態遷移にかかる時間を表す。

```

1 : Sensory input  $x$ ;
2 : Update  $Q(x', a')$  by  $x, x', a', E(y)$ ;
3 :  $x' \rightarrow x$ ;  $a' \rightarrow a$ ;
4 : If  $a=1$  then Drive leg
   else Stay;
5 : Wait for  $\delta t$  time;
6 : Sensory input  $y$  for  $E(y)$ ;
7 : Goto 1;
.....(16)
    
```

5. 計算機シミュレーション

図1の仮想的ロボットを、6足および8足に設定して計算機シミュレーションを行い、基本的な歩様獲得の能力と、脚の故障を含めた動的環境への適応能力を確認する実験を行った。

表1 Q-Learning および Q-NET のパラメータ設定

Reduction ratio γ	0.01
Temperature T	0.2
Annealing schedule	$T \leftarrow T * 0.9999$
Q-NET hidden units	54(6 legs), 24(8 legs)
Learning rate for BP	5.75
Initial Q-NET weights	Random (-0.01, 0.01)

図5 6足ロボットに対して獲得された歩様

Q-Learning および Q-NET のパラメータ設定を表1に示す。各 Q-NET の初期 Q 値をランダムに設定することで、未学習状態でロボットが知識をもたない状況を設定した。また各脚の状態遷移のタイミングは共通に設定した。

図6に6足ロボットに対するシミュレーション結果を示す。図6は状態遷移の単位時間を1として計った時間軸に対して、歩行が行われた頻度をプロットしたもので、具体的には100単位時間ごとに全脚が $E_2=1$ を受取る頻度を縦軸に表示している。

シミュレーションでは約1800単位時間の後に図5の cycle 1 に示す歩様が得られた。200単位時間の後にこの歩様は崩れ、その後数回継続した歩様が獲得された後、約3800単位時間後に図5の cycle 2 に示す歩様が得られた。この歩様はその後もある程度の時間内で崩れることはなかった。

一度安定して獲得された歩様が再び崩れてしまう現

図6 6足ロボットの歩行頻度のプロット結果

図7 途中で脚を故障させた場合の8足ロボットの歩行頻度プロット結果

(a) 脚故障前の獲得歩様 (b) 第7脚故障後の獲得歩容

図8 脚故障前、故障後に獲得された歩様

象は、温度 T による探索の行動決定のランダムさが高いためと考えられる。

表1に示すようなQ-Learningの温度係数 T のアニーリングスケジュールは、学習初期で高いランダムさを与えて広く探索を行い、終期においてよりランダムさを低くすることでよい解を固定するような戦略を与えている⁽⁶⁾。したがってアニーリングスケジュールを適切に設定することでより早期に歩様を獲得できる可能性がある。しかし温度の下げ方が急な場合には十分な探索ができないまま低い探索能力にとどまり収束が逆に遅れる問題が生じるなど、適切なスケジュールは問題に深く依存する。

表1のパラメータは極めてゆるやかなスケジュールであるが、広い問題範囲で比較的よい性能を与えることが確認できている。ただし本論での実験は可能な限り先験的知識を導入しない状況を設定しているゆえ、図6および7に示すように、問題に特殊化したコントローラによらなくとも最終的に歩様が獲得できていることは、本方法の頑強性の一つを示すものと考えられる。

一方、動的な環境に対する適応能力を確認するために、8足の場合について歩様獲得後に脚が故障した場合を設定した実験を行った。図7に図6と同じ軸をとり結果をプロットしている。

図7に示すように、5500単位時間の後に、図1の左下脚(第7脚)を、第7脚のコントローラの行動出力にかかわらず遊脚状態から遷移しないように設定した。

ロボットは約800単位時間の後第7脚を利用しない歩様を発見し、1000単位時間の後崩れずに安定した歩様に収束した。図8に脚の故障以前の歩様と獲得後の歩様を示す。

各コントローラに故障を明示的に知らせてはいないにもかかわらず、故障後に自動的に新しい歩様の獲得が可能であることが明らかとなり、本手法が予測のできないさまざまな状況に対して頑健に歩行を可能とする特徴をもつことが示された。

また故障時点から歩様を獲得するまでの時間が学習

初期からの歩様獲得に比べて短いことが観察できる。これは、Q-Learningを利用することにより、歩行に寄与するような状態、行動の断片が強化されることで、新たな拘束条件での歩様獲得は、これら断片をつなぐような小さな探索行為で済むことによると考えられる。

ただしアニーリングスケジュールを固定している場合、温度が低下している時点で脚の故障が起きる状況も考えられ、この場合、新しい歩様の探索能力が低いために収束が著しく遅れることが考えられる。これに対処する方法として、例えば継続して高い報酬が得られている状況、すなわち崩れない歩様が獲得できたと判断した時点で温度を再び高くするようなヒューリスティックを導入する方法も考えられる。具体的な方法等は今後の課題として残されている。

6. 実機への適用に関する議論

以上のシミュレーション結果は、簡単な二次元のシミュレーションモデルにおいて強化学習により適応的な歩様獲得能力が期待できることを示している。そこで次の段階として実機ロボットへの実装の方法、および可能性を議論する必要がある。実機においては問題の複雑化やハードウェアの制約など、シミュレーションには含まれない技術的制約を考慮しなければならない。この点から議論はおもに次の3項目に集約される。

(1) 三次元空間の実ロボットを扱う場合には脚の自由度の増加や直進以外のさまざまな移動目的を考慮する必要があり、強化学習の行動・状態数はシミュレーションに比較して大きいものとなる。この場合組合せ爆発が生じ学習収束性が著しく悪化する恐れはないか。

(2) 実ロボットの耐久性の点からは、シミュレーション結果に見られるような多数回の試行を常に要求するのは問題がある。より高速に歩様の構築ができないのか。

(3) 適切な学習パラメータを系統的に発見することはできないか。

シミュレーションではロボットの機構依存の知識を利用していないにもかかわらず適応的な歩様獲得が可能になったことが明らかになっている。したがって実機ロボットへ実装する場合、個々の実機の機構上の知識を利用して、問題(ロボット)自身に適した前・後処理を付加することが考えられる。

そこで第1の問題に対しては、例えば前処理として強化学習の行動出力を脚へのマクロな命令(例えば着地検出まで遷移する等)とし、さらに観測する状態に

関してもマクロ化することなどにより組合せを抑える方法を一例としてあげることができる。また第2の問題に対しては、歩様の周期があらかじめわかっている場合、後処理としてこの周期で出現するパターンを検出することで一度出現した歩様の固定化、または固定ルール化、メタ知識化を行い、確率的探索期間を費やすことなく高速な適応歩様を生成するようなシステムとする方法が考えられる。

ただし第1の問題に関しては、方向転換の導入や自由度の高い脚を用いるロボットへの適用など、組合せの増大は必ずしも回避できるものではない。これらに関してはより自由度を高めたシミュレーションにより挙動追跡することが課題として残されている。第2の問題に関しては、動的環境での移動を考えると、一般には各自点で試された行動・状態対が固定すべき正しい歩様のルールであるかを判断することは難しい。前述の歩様メタ知識化の具体的方法の展開も含めて、この問題領域も今後の研究課題となる。また第3の強化学習パラメータの系統的発見法に関しては、現時点でこれに対する理論的解析の研究例はまだ見当たらない。これはニューラルネットワークの学習係数決定と同様な困難さをもつ問題であり、例えば遺伝的アルゴリズムによる組合せ探索による発見なども考えられるが、具体化には初期値集団の設定法などさらに検討の必要がある。

7. 結 論

強化学習手法である Q-Learning を利用して、予測し得ない状況下で確実に歩行しうるような、学習形の多足ロボットの脚コントロールの一手法を提案した。2次元の仮想ロボットがパイプ内を歩行する歩様獲得問題を例題として、基本的な能力を実験的に確認し、これが脚の故障などに対して頑強な歩様獲得手法であることを確認した。

文 献

- (1) 特集歩行ロボット, 日本ロボット学会誌, **11-3**(1993), 305.
- (2) Brooks, R. A., A robot that walks; emergent behaviors from a carefully evolved network, *Neural Compt.*, **1-2**(1989), 253.
- (3) Sutton, R. S., *Reinforcement Learning*, (1993), 1, Kluwer Academic Pub.
- (4) Connell, J. H. and Mahadevan, S., *Robot Learning*, (1993), 1, Kluwer Academic Pub.
- (5) Watkins, C. J. C. H. and Dayan, P., Technical note: Q-learning, *Machine Learning*, **8**(1992), 279.
- (6) Lin, L. J. and Mitchell, T. M., Reinforcement Learning with Hidden States, *From animals to animats 2*, (1993), 271, The MIT Press.
- (7) Sutton, R. S., Learning to Predict by the Method of Temporal Differences, *Machine Learning*, **3**(1988), 9.
- (8) Holland, J. H., *Adaptation in Natural and Artificial Systems*, (1992), 171, The MIT Press.