情報処理学会論文誌 Vol. 50 No. 3 1170–1180 (Mar. 2009)

推薦論文

強化学習を用いたモジュール型多脚ロボットにおける適応的移動法獲得

新堀航大[†1] 兵頭和幸[†1]
砂山享祐[†2] 三上貞芳[†3]

探査作業のような未知環境の下でロボットを用いるような研究が進められているが，ロボットのパーツの破損によって移動不可となる可能性などについては，まだ研究の余地が残されている．本論文では，ロボットが破損した場合でも，回収が困難な場合には破損部以外の利用可能なアクチュエータを用いることで移動法を再獲得するようなシステムを想定し，想定外の状況にもある程度適応できるような移動法獲得を，強化学習を用いて実現する．提案する手法では，脚形状から車輪形状などの想定外の形状へのアクチュエータモジュールの換装もある程度可能なシステムを前提とする．このような前提では新たな移動手順を広く探査することになるが，ロボットの移動機能を迅速に回復するためには，なるべく有用な行動を速く探査し利用することに重点を置く必要がある．このため，本研究では強化学習手法に対して，時間的信頼性に基づいた「行動価値の成長」と呼ぶ再探索手法を導入する．3D 物理シミュレータによる 6 脚移動ロボットの実験により，提案する方法が比較的高速に良い候補となる移動法を獲得できていることが示されている．

Acquisition of Adaptive Movement Strategy by Reinforcement Learning in Modular Multiple-leg Mobile Robot

Kodai Shimbori,[†1] Kazuyuki Hyodo,[†1]
Kyosuke Sunayama[†2] and Sadayoshi Mikami[†3]

Currently, intensive studies are carried out to make robots that can work under an unknown environment. It is however not so much investigated for methods to realize recovery from situations where a part of a robot has been broken. This study is to propose a configuration of a mobile robot system that is able to achieve a new movement under the situation where some of its actuators are broken and replaced by alternative ones, which may not be the same configuration as the original ones. In particular, the proposed method is designed to be able to deal with replacement of a leg-type actuator to a wheel-type actuator, which may not be considered in design-time. The proposed method is based on a Reinforcement Learning and is modified so that it can achieve rapid conversion over a wide search space. To this end, a "growth of action-value" method is proposed, which enables effective exploration of an action space based on temporal reliability of each action-value. A series of 3D simulation-based experiments are conducted, where the proposed method shows rapid conversion to a good candidate of movement patterns.

1. はじめに

本研究の目的は，ロボットが破損などによって移動不可となることを回避するシステムの開発である．

近年，未知環境での探査作業にロボットを用いる研究がさかんになされている．このような環境では，必要時に故障で動かないという状況にも遭遇しうる．ロボットを災害現場のような緊急を要する場面で用いるには，軽度の破損程度で移動不可となってはならず，修理に時間がかかってはならない．本研究では移動ロボットの移動不可を迅速に回復するシステムの開発を行うものである．未知環境での探査作業では人間が立ち入ることが困難な遠隔地でロボットを用いることが想定される．そのため，回収が困難である遠隔地でロボットが移動不可となってはならない．この問題を解決するために，破損部を補った移動法を自動的に獲得させることによって，探査作業を続行させることを可能とし，回収も容易にできることを目的とする．回収をした場合には，破損脚を代替脚あるいは車輪へ交換した後，新たにその構成の下で適切に前進できるような新しい移動法を自動的に獲得させることを目的とする．これらの移動法獲得手法には，教師なし学習の 1 つである強化学習手法を利用する．強化学

†1 公立はこだて未来大学大学院
Graduate School of Future University-Hakodate
†2 東芝ソリューション株式会社
Toshiba Solutions Corpolation
†3 公立はこだて未来大学
Future University-Hakodate
本論文の内容は 2007 年 9 月の北海道支部主催シンポジウムにて報告され，同支部長により情報処理学会論文誌ジャーナルへの掲載が推薦された論文である．

習により自律的に獲得させることの利点として，確率的な行動ルールを得ることができるため，不確実性の高い環境での暫定的な動作ルールの獲得に適している．

本研究では，第 1 に，ロボット破損時に残された可動部を利用して移動法を再獲得する自己修復システムの開発を目指す．第 2 に，移動アクチュエータのモジュール化を行い破損部の交換の簡単化，また，他形状モジュールの取り付けを可能とし，その際の自動チューニングシステムの開発を行う．本論文では，自己修復システム・移動アクチュエータのモジュール化・自動チューニングシステムを提案し，物理シミュレータ上で 6 脚のモジュール型移動ロボットモデルを対象として具体的な適応的移動法獲得の手順を説明し，実験を通して有効性を示す．

2.　移動不可を回避するシステム要素の提案

ここでは移動不可からの回復を実現するための要素である自己修復システム，アクチュエータのモジュール化，自動チューニングシステムについて必要性と可能性を述べる．

2.1　自己修復システム

移動法が完全にプログラムされたロボットでは，一部に破損を負うとプログラムどおりの動作ができず，移動不可になる可能性が高い．しかし，残存機能だけで自律的に移動法が再獲得可能ならば，正常状態より移動効率は劣化しても移動不可状態より利用性が高い．そのため，移動不可からの回復という目標に対し自己修復システムが必要となる．これはロボット自体に学習機構を持たせることで実現する．

2.2　アクチュエータのモジュール化

移動不可から回復させるために，破損部を代替パーツで補う方法を考える．ただし，緊急を要する場面ではパーツ交換に時間がかかってはならない．そのため，交換を容易にするためにアクチュエータのモジュール化を想定する．この実現に関しては様々な方式が考えられるが，たとえば一般的な構成としては，**図 1** に示すように，アクチュエータに制御システム（子）を実装し，ロボットの中枢となる制御システム（親）と共通の規格で通信を行うことで全体を制御する方式などが考えられる．本論文ではアクチュエータの構造について制約の少ない状況を考える．たとえば，6 脚ロボットで 1 脚が破損した場合に，モジュール部が同一であれば，脚ではなく車輪であってもよい．これは，パーツ（アクチュエータ）の流用性を高める．この場合，様々な形状のアクチュエータの換装について事前に動作パターンの設計を用意することは不可能なので，自動チューニングにより，オンラインで獲得させる方法が必要になる．

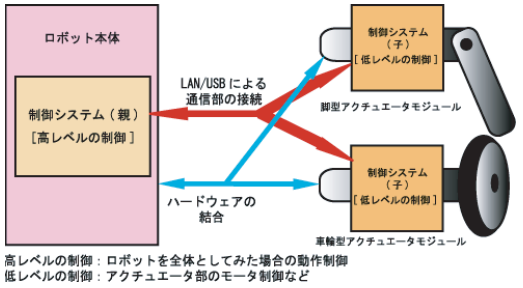

図 1　モジュール化における制御モデル
Fig. 1　Module control model.

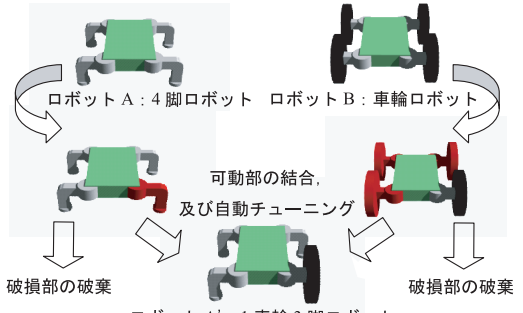

図 2　可動部を集めることによる修復
Fig. 2　Fixing a robot by using live parts from other robots.

2.3　自動チューニングシステム

状況に応じた脚の自動チューニングが有効に働く例を示す．**図 2** の例では，一部破損の脚ロボット A と，大部分破損の車輪ロボット B があった場合に，ロボット A では残りの 3 脚で自己修復可能と推測できるが，ロボット B に関しては，破損が著しく自己修復不可能である．ロボット A が自己修復を行った場合，3 脚による動歩行が必要となり静止時の安定性は得られない．一方，ロボット B は可動なアクチュエータが 1 つ存在している．そこで，ロボット B の可動アクチュエータを，ロボット A に移植することで 1 輪 3 脚のロボット A’ ができる．これにより，4 点接地が実現でき，自動チューニングを適用することで静歩行による安定性が保証できる．この例は，モジュール化と自動チューニングシステムの実現により，故障からの復帰における稼働率の向上が期待できる例を表している．

3. 対象ロボット

本論文では，移動法再獲得の学習手法を中心に説明する．本論文では以下に示す一般的な 6 脚の移動ロボットを例にとる．脚モジュールは 1 脚の自由度を**図 3** に示すような 3 自由度とし，すべて回転関節で構成する．ロボットのサイズや取り付け角度などに関して**図 4** に示す．点線は各関節の最大可動範囲を示す．脚の動作パターンは固定的な 3 パターンであり，**図 5** と**表 1** に示すようなパターン 0（図 5-(a)），パターン 1（図 5-(b)），パターン 2（図 5-(c)）の動作である．ただし，後ろ脚（ℓ_5，ℓ_6）に関しては図 4 に示すように取り付け角度が 45 [deg] 変更されているものとし，左右対称で取り付けられている．脚の動作パターンの動作順序はパターン 0 からパターン 1，パターン 1 からパターン 2，パターン 2 からパターン 0 の 1 方向のみであり，**図 6** に示すように，3 脚を 1 つのグループとした 2 グループを同期対象とし，各グループ交互に動作を行う．本論文では脚モジュールの破損の際に，車輪モジュールへ換装する場合も対象とするが，その際の車輪モジュールは受動車輪であるものとする．受動車輪を用いた場合には脚の動作パターンの遷移の影響は受けない．

4. シミュレーションによる具体例

提案システムの学習手法を検証するため，実装対象とする脚・車輪換装可能なモジュール型移動ロボットモデルのシミュレータを作成して用いた．画面の例を**図 7** に示す．シミュレータ画面ではロボットモデルの動作を 3D で確認することができ，コンソールでは各時間のそれぞれの脚の動作パターン，移動距離，角度誤差，報酬値，座標を確認することができる．本シミュレータはオープンソースの物理計算エンジンライブラリである Open Dynamics Engine（ODE）を用いて，WindowsXP を搭載した汎用 PC（Pentium 4 HT CPU 3.2 GHz，メモリ 1 GB，グラフィックカード RADEON9800PRO）で実装を行った．開発環境は Visual Studio 2005 Professional で C 言語を用いている．本研究のシミュレータでは，3 章の対象ロボットの仕様をもとにロボットモデルを再現している．ただし，車輪モジュールに関して

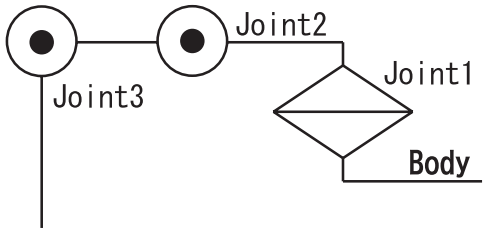

図 3　関節構成図
Fig. 3　Leg joint configuration diagram.

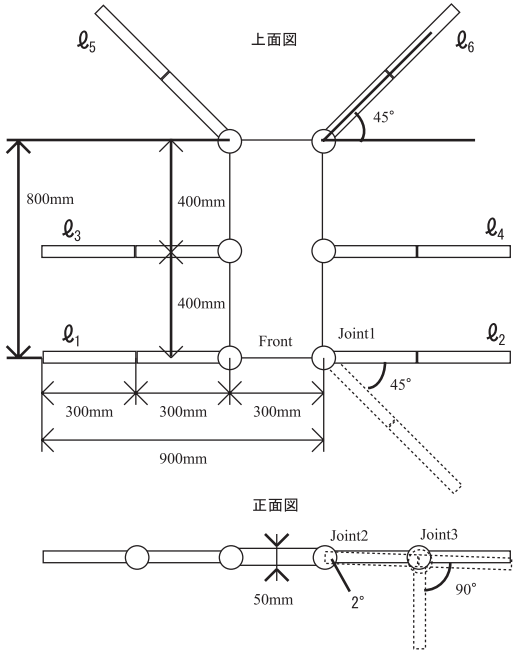

図 4　ロボットのサイズと可能動作角
Fig. 4　Robot size and the range of movement.

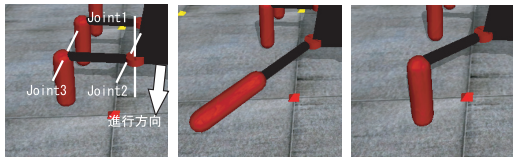

図 5　脚の動作パターン
Fig. 5　Leg's motion pattern.

表 1　各パターンの関節角度
Table 1　Joint angle for each pattern.

	パターン 0	パターン 1	パターン 2
Joint1	0 [deg]	45 [deg]	45 [deg]
Joint2	0 [deg]	0 [deg]	2 [deg]
Joint3	90 [deg]	0 [deg]	90 [deg]

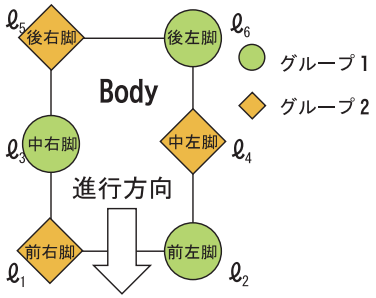

図 6　同期グループ
Fig. 6　Synchronizing leg group.

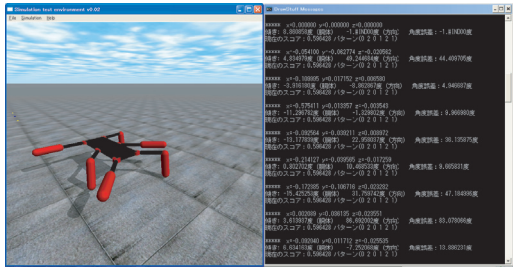

図 7　シミュレータ画面
Fig. 7　Simulator screenshot.

は，本来であれば脚先に車輪を実装するべきであるが，シミュレータ実装の簡単化のため**図 8** のように，進行方向へ車輪が回転するように車軸を直接胴体へ固定する実装方法とし，隣り合う脚との衝突は検出しないものとする．

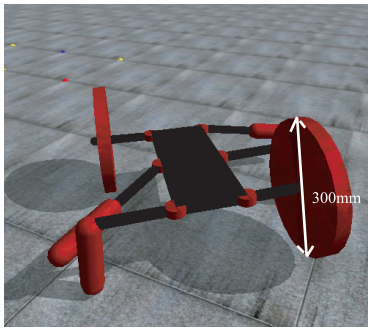

図 8　車輪の例
Fig. 8　Example of a wheel module.

5. 学習による移動法獲得

以上の前提のもとで，本システムで用いる移動法獲得の手法と，それに用いる強化学習手法について述べる．具体的には，強化学習の設計と強化学習の収束速度を高めるための行動価値の成長，およびシステムの流れについて述べる．

5.1 移動法の獲得手法

本研究では，破損時や，脚の再構成を行った時点から，有効な移動法を獲得させるための手法として，強化学習を適用することとした．提案手法では，強化学習の段階を 2 段階に分けて適用する．第 1 段階ではランダムに行動選択し（行動については後述），動作の評価に基づく報酬を得て価値関数の更新を行う．第 1 段階の学習は一定のステップ数で完了させ，第 2 段階を開始する．第 2 段階では次に示す行動選択の規則に従うものとし，試行することで報酬を得て価値関数の更新を行う．

(1)　行動選択

提案するシステムでの行動の定義は，アクチュエータの動作開始パターンの組合せの 1 つをさすものとする．つまり，それぞれのアクチュエータに対して，試行開始状態での脚の動作パターンを決定することに相当する．具体的には，行動を識別するインデックスを a とし，脚 n の動作パターンの値を P_n として式 (1)，式 (2) で定義する．

$$P_n \in \{0, 1, 2\} \tag{1}$$

表 2　ステップとサイクルの一例
Table 2　Example of step and cycle.

	cycle					
	step1	step2	step3	step4	step5	step6
ℓ_1	pat0	-	pat1	-	pat2	-
ℓ_4	pat0	-	pat1	-	pat2	-
ℓ_5	pat0	-	pat1	-	pat2	-
ℓ_2	-	pat1	-	pat2	-	pat0
ℓ_3	-	pat1	-	pat2	-	pat0
ℓ_6	-	pat1	-	pat2	-	pat0

pat:pattern　group1 (ℓ_1, ℓ_4, ℓ_5)　group1 (ℓ_2, ℓ_3, ℓ_6)

$$a = \sum_{n=1}^{6} 3^{n-1} P_n \tag{2}$$

行動選択の規則は，ϵ-greedy 手法[1] を拡張したものを用いる．ϵ-greedy では通常，行動価値の最も高い行動の選択を行い，次に ϵ の確率でランダム性の高い方法で行動選択を行う．しかし，本論文では確率 ϵ の場合には，通常の ϵ-greedy とは異なり，最も行動価値の高い行動に対して，少量の変更を加えたものと同等の類似行動を選択することで行動決定とする．少量の変更とは，脚 6 本のうち，ランダムに選択された 1〜3 脚に対してのみ脚の行動パターンの開始状態をランダムに変更することである．これに関しては，行動価値の高い行動から少量の変更を加えることによって，探索を行動価値の高い行動の近傍に限定し探索の爆発を防ぐことを意図している．

(2)　**試行**

試行は選択された行動を 2 サイクル動作させることを 1 試行とする．1 サイクルの定義としては，シミュレータで用いるロボットモデルの場合，脚の動作パターンの状態を 1 つ進めることを 1 ステップとすると，脚の動作パターンが 3 状態であるのに対して同期グループが 2 つ存在するため，6 ステップでスタート状態に戻る．この一連の動作を 1 サイクルとする．行動を $< P_1, \cdots, P_6 >=< 0, 1, 1, 0, 0, 1 >$ とした場合の一例を**表 2** に示す．試行からは移動距離，試行前と試行後のロボットの向きの角度誤差を得ることができる．

(3)　**報酬**

本研究では 1 試行での移動距離が大きいことと，直進できることを価値の高い移動法とする．そのため，**図 9** に示すような移動距離を報酬の基準とし，直進性を反映させるために試行開始時と試行終了時のロボットの向きの角度誤差をペナルティとして与えたものを行動に対する報酬とする．報酬式は試行を t とし，R を行動 a の報酬値，D を $t-1$ から t での移動距離，E を $t-1$ から t での角度誤差，δ（$0 < \delta$）をペナルティ係数として式 (3) に示す．

$$R_t(a) = D(t, t-1) - \delta\,|E(t, t-1)| \tag{3}$$

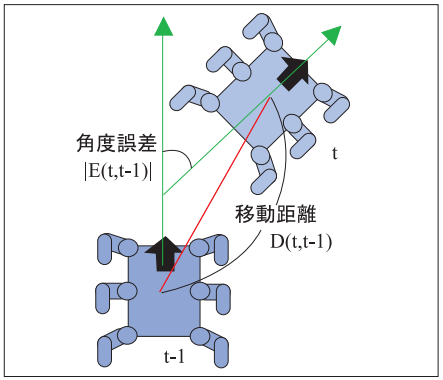

図 9　移動距離と角度誤差
Fig. 9　Travel distance and angular displacement.

(4)　**行動価値関数の更新**

試行ごとに得られた報酬をもとに行動価値が更新される．更新式は Q を行動価値とし，割引係数 μ（$0 < \mu < 1$）を用いて式 (4) に示す．これは通常の強化学習の更新式（行動価値について）と同一である．

$$Q(a) = \mu Q(a) + (1-\mu) R_t(a) \tag{4}$$

5.2　行動価値の成長

5.1 節で述べた学習手法のみでは，脚・車輪混在状態で，かつ障害状況が刻々と変化する状況など，本研究の対象とするような幅広い可能パターンの探索・収束を必要とする状況への適用には不十分な点が残されている．本学習手法は ϵ-greedy 手法を基準としているため，ロボットが破損状態から正常状態へ回復するといった試行途中でパフォーマンスの上昇が起きる場合に，適切な移動法が獲得できないという問題が生じる可能性がある．破損状態と正常状態ではそれぞれ最適な移動法は異なるため，先に低パフォーマンスである破損状態で学習の収束が行われてしまうと，正常状態に回復した場合であっても同一の行動価値関数を用いているため，破損状態で獲得した移動法を選択し続け過学習を引き起こしてしまう．よっ

て，より適切な移動法に回復することができなくなってしまう．

そこで過学習による偏りを修正するために，時間的信頼性を考慮した行動価値の成長を提案する．一般的には，行動価値の更新時刻が古いものに対して，その行動価値の信頼性は低いと想定できる．行動価値の成長は更新時間の古さの度合いに応じて行動価値に直接関与し価値の上昇を行うものである．これを行うことにより，行動価値が低いものであっても，信頼性が低くなるにつれて価値の上昇が起きるため，最終的に価値が最大となり試行される．よって例のようなケースであってもより適切な解を順次見つけることが可能となる．行動価値の成長を用いることで，一時的に不適切な行動が試行されてしまう場合もあるが，試行により再度行動価値の更新で大きく価値の降下が起きるため，行動価値の成長によって再度価値が最大になるには時間がかかる．それに対し，最適なものではないが，ほどよい移動法に関しては，試行によって最適解付近の行動価値に降下するため，次に価値が最大になるまでの時間は短くなる．そのため相対的に行動価値の成長による試行頻度は自動的に適切なものに改善されていくことが可能となる．行動価値の成長を用いた場合と用いなかった場合の例を正常状態から破損状態，そして正常状態へ遷移した場合を例に，縦軸を各行動の行動価値，横軸を試行回数とし**図 10** に示す．各グラフの線はそれぞれの行動に対する価値の遷移を表している．図 10 から分かるように，行動価値の成長を用いない場合では最も価値の高いもので行動をし続けるため移動法の回復が行われないが，行動価値の成長を用いた場合にはすべての価値が信頼性に基づき上昇していくため，より適切な解を見つけることが可能となる．また，低い順位から成長によって持ち上げられた行動であっても，より適切な解であるならばすぐに実行される行動として切り替えることができ，また，逆に実行されている行動価値が下がった場合に，次に良いと知識として蓄積されている行動への切替えも容易となる．行動価値の成長式は ζ（$0 < \zeta < 1$）を成長係数として式 (5) に示す．

$$Q(a) = Q(a) + \zeta \frac{Time}{LastUpdateTime(a)} \tag{5}$$

$Time$ は現在の試行時刻をさし，$LastUpdateTime(a)$ は行動 a の試行による価値の最終更新時刻をさす．式 (5) は 1 試行ごとにすべての行動 a に対して適用する．

5.3　システムフロー

ここでは 5.1 節と 5.2 節で述べた手法を組み込んだ全体のシステムの流れについて**図 11** を用いて説明する．第 1 段階では，ランダムに行動を選択して試行し，報酬を得ることによって行動価値の更新を行う．この一連の動作を一定時間繰り返した後，第 2 段階へ遷移する．第 2 段階の学習では行動価値関数の中から最も価値の高い行動を選択し ϵ の確率で

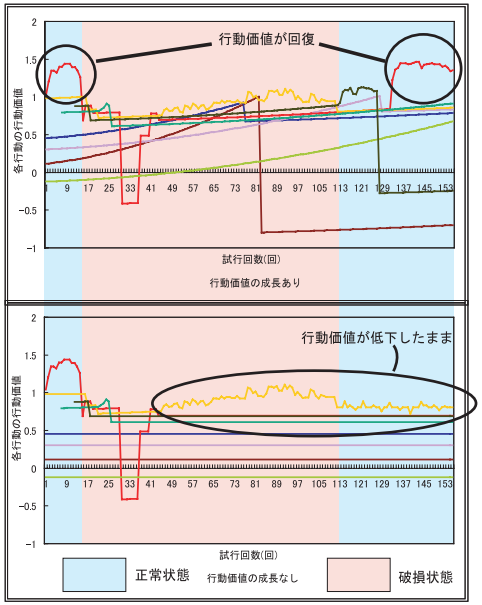

図 10　行動価値の成長による効果
Fig. 10　An effect of the growth of action-value.

a と b に分岐する．a の場合には選択した行動をそのまま試行し，b の場合には 5.1 節 (1) で述べた類似な行動を試行する．試行結果から報酬を得て行動価値を更新した後，行動価値全体に対して行動価値の成長を行う．この一連の動作を繰り返す．

5.4　本学習手法と TD(λ) 法の比較

本システムのような，実環境で稼動しつつ学習を行うことが想定されるシステムでは，多くの試行を繰り返し，データを収集して学習する手法は適当ではない．そのため，少ない試行で，ある程度適切な解に収束可能な学習手法が必要となる．その解決策である適格度トレースを用いた手法の 1 つである TD(λ) 法[1] との比較を行う．

TD(λ) 法は TD 法に適格度トレースを取り入れた手法であり，計算量が多くなるが，少ない試行数で収束可能な収束速度の速い学習手法である．TD(λ) 法では訪問した各状態に対して，将来の報酬と状態を見通した結果から各状態へトレース減衰パラメータ λ の値に従いバックアップを行う．これによって速い速度で目先の価値にとらわれない最適解への収

束が可能とされている．しかし，本学習手法との違いは対象とするロボットの変化への適応速度である．TD(λ) 法では，ロボット自体の変化によって最適解が変動するような場合に，完全に価値が切り替わるまでは連続して不安定な行動をとり続けてしまう可能性があり，行動の切替わりの速度が遅い．連続した不安定な行動は本研究で想定するような即時に安定を求められるシステムでは適切ではない．

また，このシステムでは greedy な方策を利用することで，ロボットの変化によって現在の行動が不適（不安定）となる場合でも，次に良い行動（安定した行動）とされるものを選択させるようにしている．これに対して行動価値の成長を導入すると，価値の高い安定した行動を，順次試行してゆくような仕組みが導入されるため，探査を行いつつも，安定した行動をつねに確保しておくことが可能である．SoftMax 方策[1)] といった，行動価値の高い行動を優先的に選択する手法が知られているが，これらは行動選択の確率的な要因が強いため，本論文で扱うように，試行に十分な回数をさけないような対象には，適用が難しい．

以上から，本研究で想定するシステムに対して，本学習手法は適切であると考えられる．ただし，行動価値の成長速度は成長式 (5) に依存しているため，時間的信頼性の低いものに関しては過剰に優先して試行されてしまうという可能性もあり，成長式についてはより検討の余地が残されている．

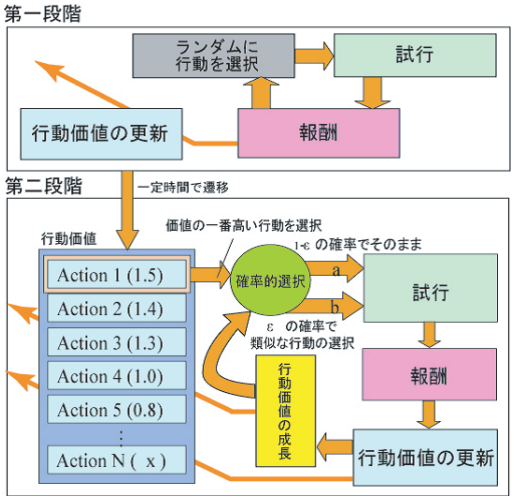

図 11　システムフロー図
Fig. 11　System flow.

6. 評価実験と考察

自己修復システムと自動チューニングシステムの性能を，シミュレータ上で検証することを目的として実験を行った．提案するシステムなどのオンライン適応システムにおいては，目標とする動作に対し，学習手法での収束時間が人間の手による修復やチューニングよりも長い時間がかかることは好ましくないため，各種条件下での収束性の目安を示すことが必要となる．また，実際に獲得した移動法の精度も評価の対象となる．本研究では 3 つの問題を設定し実験を行う．また，本実験で学習に用いた各パラメータ値は，**表 3** に示すとおりに設定した．これらパラメータはいくつかの値の幅で実験を繰り返した際，最も良好な動作を行うものを選んでいる．しかし，このパラメータ値が特異的に良好な性能を与えているわけでなく，比較的広い範囲で同様な挙動が観測できている．

6.1 実験 1：正常状態での移動法獲得

まず，実験 1 として各アクチュエータが正常な状態での移動法獲得実験を行う．この実験の目的は，これ以降の実験で用いるための基本学習データの取得と，パフォーマンスを比較するための基準を得るためである．正常状態で得られたパフォーマンスを基本パフォーマンスとし，破損状態から自己修復システムを用いてどれだけパフォーマンスの得られる移動方法に修正できたか，また，他形状モジュール混在時の自動チューニングシステムでどれだけパフォーマンスを得られたかの指標とする．ロボットの正常状態の様子を**図 12** に示す．

この実験の結果として，縦軸の第 1 軸を報酬値，第 2 軸を角度誤差とし，横軸を試行回数とした学習遷移のグラフを**図 13** に示す．この結果から分かることは，学習の第 2 段階である 25 試行を経過し，50 試行目から報酬の値が約 1.5 付近に収束傾向があることが分かる．これは学習の収束を意味する．また，同様に 50 試行目以降の角度誤差は数度に収束していることが分かる．しかし，これらが収束傾向にある中で，ときどき突発的な値の変動が見ら

表 3　学習のパラメータ値
Table 3　Learning parameters.

	設定値
ϵ	$1 \rightarrow 0$
δ	0.01
μ	0.2
ζ	0.0005
学習第 1 段階の試行数	25 [試行]

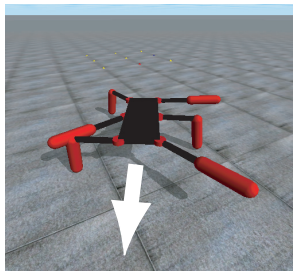

図 12　実験 1 のロボットの様子（正常状態）
Fig. 12　Experiment 1: State of robot (Normal state).

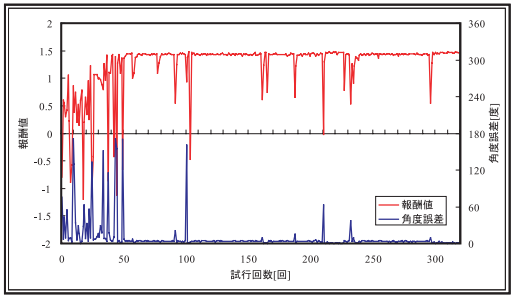

図 13　実験 1：報酬と角度誤差の遷移
Fig. 13　Experiment 1: Transition of reward value and angular displacement.

れるのは，行動価値の成長の特性によるものである．以上で得られた結果より正常状態での基本パフォーマンスの値を 1.5 とし，以降の実験でのパフォーマンスの比較対象の基準とする．

6.2　実験 2：破損状態から復旧した際の自己修復システムを用いた移動法回復

次に，実験 2 として破損状態から何らかの要因（衝撃など）によって復旧した際の移動法回復実験を行う．この実験の目的は，1 度破損した状態から破損部が何らかの要因によって復旧した場合に，自律的に破損前のパフォーマンスと同等，または同じ移動法を獲得できるかどうかを検証することである．実験設定としては，破損状態で 1 度移動法を獲得した学習データを初期状態の知識として与える．対象ロボットは初期状態では右前足を動作不能とし，1 度収束した後（今回は 130 試行後）に右前足を正常状態へ回復させるというシナリオ

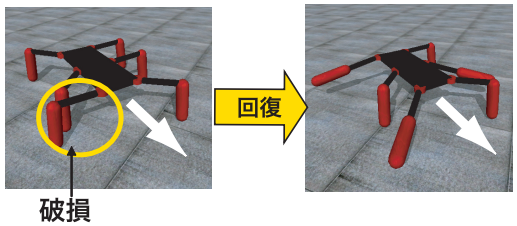

図 14　実験 2 のロボットの様子（破損から回復シナリオ）
Fig. 14　Experiment 2: State of robot (Recovery from damage).

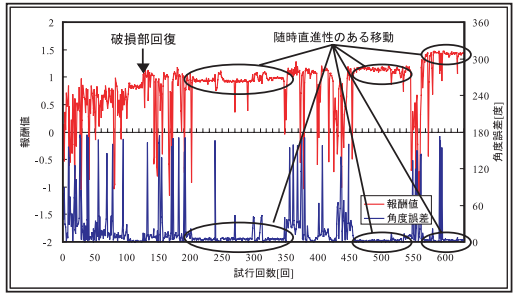

図 15　実験 2：報酬と角度誤差の遷移
Fig. 15　Experiment 2: Transition of reward value and angular displacement.

で実験を行う．実験シナリオのイメージを**図 14** に示す．

この実験の結果について，他の実験同様に報酬値と角度誤差について**図 15** に示した．この結果から分かることは，正常状態から破損，もしくは他形状モジュールで換装した状態より，収束に時間がかかるということである．つまり，パフォーマンスの低下する状態への収束は速いが，パフォーマンスが戻る状態への収束は遅いということが分かる．しかし，本実験では，報酬値を見ると約 100 から 125 試行，約 200 から 340 試行，約 440 から 520 試行，約 800 試行以降でそれぞれ突発的な変動値は出ているが一時的に安定状態にある．その際，角度誤差も，変動値が出ているとき以外は小さい値に収束している．つまり，良いパフォーマンスまで回復するには時間がかかるが，順次直進精度の高い移動法を獲得しつつ学習が行われているといえる．以上より，本実験では本学習手法において，破損から回復した際の移動法再獲得が可能であり，自己修復可能であることを示すことができた．パフォーマ

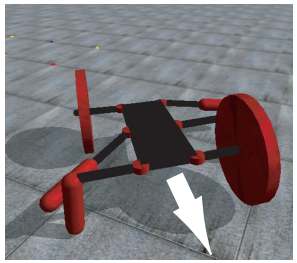

図 16　実験 3 のロボットの様子（車輪混在状態）
Fig. 16　Experiment 3: State of robot (Wheels replaced).

ンスの面でも，最終的な学習収束時には基本パフォーマンスと同等の値まで回復できている．また，この移動法の獲得では本学習手法の特性上，行動価値の成長という手法を導入しなければ，基本パフォーマンスと同等の値まで回復するという保証はできず，最初の安定状態（100 から 125 試行）のまま収束してしまう可能性がある．ゆえに，本実験において行動価値の成長を用いることで正常状態のパフォーマンスまで回復させられることを実証し，有用性を示すことができた．

6.3　実験 3：車輪混在状態での自動チューニング

最後に，実験 3 として車輪混在状態での自動チューニングを用いた移動法獲得実験を行う（実験 3A）．この実験の目的は，対象ロボットが破損した場合にアクチュエータを脚モジュールから車輪モジュールへ換装を行った場合に移動法が獲得できるかを確認し，また，基本パフォーマンスと比較してどれくらいのパフォーマンスの得られる移動法が獲得できるかを検証することである．実験設定としては実験 1 から得られた学習データを初期状態の知識として与え，対角上の脚を車輪モジュールへと換装を行った状態で学習を始める．この車輪混在状態のロボットの様子を**図 16** に示す．また実験 3A では，車輪混在というハードウェアへの大きな変更があるため，同一の状態のロボットに対し，「実験 1 から得られた学習データを初期状態の知識として与えない」場合の実験結果（実験 3B）を示し，比較検討を行う．

この実験 3A の結果について，他の実験同様に報酬値と角度誤差について**図 17** に示した．この結果から分かることは，報酬値では 50 試行目付近で安定状態が見られるが，角度誤差では約 10 [deg] の誤差が見られ直進性があるとはいえない．しかし 110 試行以降からは報酬値が約 1.25 付近に収束するとともに角度誤差でもほぼ誤差はないに等しい程度となり，良

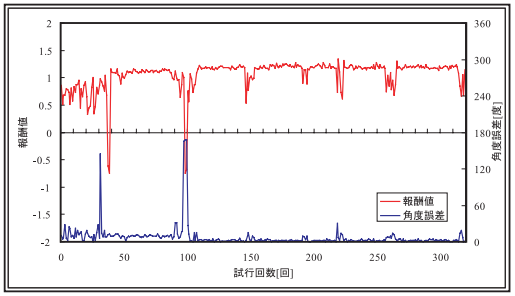

図 17　実験 3A：報酬と角度誤差の遷移
Fig. 17　Experiment 3A: Transition of reward value and angular displacement.

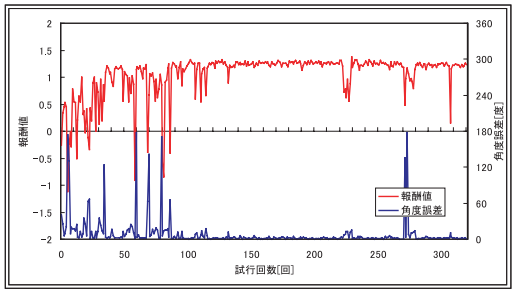

図 18　実験 3B：報酬と角度誤差の遷移
Fig. 18　Experiment 3B: Transition of reward value and angular displacement.

い移動法が獲得できている．また，突発的な変動値が発生しているのは 6.1 節の実験同様，本学習手法の特性によるものである．以上より，基本パフォーマンスと比較した場合でも，約 83% 程度のパフォーマンスを得ることができており，直進性と移動速度のある移動法を自律的に獲得できているといえる．また，車輪モジュールを混在させた場合であっても，本学習手法を用いることで自動チューニングが実現できているといえる．

次に，実験 3B の結果を**図 18** に示した．この結果と比較していえることは，収束までにかかる速度が実験 A では約 110 試行，実験 B では約 115 試行とほぼ誤差程度の差であり，同等スピードでの学習収束が得られているといえる．どちらの条件であっても特別優性な傾向は見られないが，逆にどちらの場合を用いても劣ることはなく，比較的速い収束速度を得ることが可能である．

表 4　各実験の収束試行回数と復旧率
Table 4　Number of trials for convergence, and recovery rate.

	試行回数 [回]	復旧率 [%]
実験 1	約 50	100
実験 2（破損まで）	約 100	53
実験 2（破損→正常）	約 450	100
実験 3A	約 110	83
実験 3B	約 115	83

7. 結　　論

本研究では，移動ロボットの破損時のリカバーや修復に対して，自己修復システム，アクチュエータのモジュール化，自動チューニングシステムの提案を行い，自己修復システムと自動チューニングシステムに関しては強化学習と行動価値の成長を用いることで実装を行った．これらを，シミュレータ実験を行うことにより，自己修復システムとモジュール換装時の自動チューニングシステムの有用性について示した．また，これらのシステムの学習収束までにかかる試行数は**表 4** に示すような回数で収束可能であり，その際の復旧率（正常状態でのパフォーマンスと各実験の収束時のパフォーマンスとの比較）について示した．これにより少ない試行回数で学習収束が可能であること示し，変化への適応力を示すことができた．本論文では，行動価値の成長という手法を提案し，これにより，過学習を抑制し，短期的には有用な解を利用させつつ，より適切な解を継続して積極的に獲得させるような調整項目を導入した．ただし，現段階では副作用により一時的な変動値が出るなどの問題点がまだ残されており，この方法が有効な場合の条件や理論解析について，今後の課題として明らかにしてゆく必要がある．また，今回の実験では，特定の学習パラメータで行っているが他の学習パラメータを設定した際の収束の変化や，TD(λ) との比較などについての詳細を検証する必要がある．本研究ではシミュレーションによる実験であったため，今後はシステムの実用化を目指すために実ロボットでの実験・評価を行う必要がある．学習手法の面では直進を対象とした移動法獲得のみの検証しか行っていないため，その他多種の行動に対しても適応できるように改良を加えていく必要がある．以上の点の改良を進めるとともに，本研究のシステムを，ハードウェア構成を含めて実ロボットへの応用することを目指してゆきたい．

参 考 文 献

1) Sutton, R.S. and Barto, A.G.: *Reinforcement Learning*, MIT Press, Cambridge, MA (1998). 三上貞芳，皆川正章（共訳）：強化学習，森北出版 (2000).
2) Dudek, G. and Jenkin, M.: *Computational Principles of Mobile Robotics*, Cambridge University Press (2000).
3) Narendra, K.S. and Thathachar, M.A.L.: *Leaning Automata*, Prentice-Hall International (1989).
4) Bekey, G.A.: *Autonomous Robots*, MIT Press (2005). 松田晃一，細部博史（共訳）：自律ロボット概論，MYCOM (2006).
5) 三上貞芳，田野浩明，嘉数侑昇：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，日本機械学会論文集（C 編），pp.246–253 (1994).
6) 岡谷商工会議所，強化学習型（動作制御自己開発型）6 脚歩行ロボットの研究開発，NEDO プロジェクト，p.9, p.44 (2006).
7) 津田　剛，ト田祐輔，提　一義：強化学習に基づく多脚ロボットにおける歩行の学習的獲得，2P1-3F-A8, ロボティクスメカトロニクス講演会'03 (2003).
8) 上原真也，スニルラル，山田孝治ほか：五脚ロボットにおける GA を用いた前進歩容の検討，2A1-C35, ROBOMEC'06 (2006).
9) 田窪朋仁，新井建生，井上健司ほか：腕脚統合型ロボット「ASTERISK」の開発，ALL-N-004, ROBOMEC'05 (2005).
10) 宮内隆弘，田窪朋仁，井上健次ほか：腕脚統合型ロボット「ASTERISK」の歩行動作生成，2A1-C37, ROBOMEC'06 (2006).

(平成 20 年 3 月 19 日受付)
(平成 20 年 12 月 5 日採録)

推　薦　文

本研究の目的とされているシステムの開発は，災害現場などにおいてロボットが実用化されるうえで必要不可欠であり，その有用性が期待できる．

（北海道支部長　鈴木恵二）

新堀　航大（学生会員）

1985 年 5 月 9 日生．2008 年 3 月公立はこだて未来大学システム情報科学部情報アーキテクチャ学科卒業．同年公立はこだて未来大学大学院システム情報科学研究科博士（前期）課程入学．現在に至る．主として，強化学習を用いた適応的移動ロボットに関する研究に従事．2007 年情報処理学会北海道支部研究奨励賞受賞．

兵頭　和幸

1980 年 9 月 3 日生．2004 年 3 月公立はこだて未来大学システム情報科学部情報アーキテクチャ学科卒業．2006 年 3 月公立はこだて未来大学大学院システム情報科学研究科博士（前期）課程修了．システム情報科学修士．同年 4 月同研究科博士（後期）課程進学．現在に至る．主として，受動歩行を用いた 2 足歩行ロボット，強化学習等の研究に従事．IEEE，日本機械学会，日本ロボット学会等の会員．

砂山　享祐（正会員）

1983 年 1 月 7 日生．2006 年 3 月公立はこだて未来大学システム情報科学部情報アーキテクチャ学科卒業．2008 年 3 月公立はこだて未来大学大学院システム情報科学研究科博士（前期）課程修了．システム情報科学修士．同年から，東芝ソリューション（株）エンベデッドソリューション事業部勤務．在学時，移動型知能ロボット群に関する研究に従事．現在に至る．

三上　貞芳（正会員）

1962 年 6 月 4 日生．1990 年北海道大学大学院工学研究科博士課程修了．工学博士．北海道大学大学院工学研究科助教授，英国ウエストイングランド大学客員研究員等を経て，2000 年 4 月より公立はこだて未来大学教授．1997 年 JSME ROBOMEC 賞，同年 ANNIE Theoretical Development Award，2004 年 JSME ROBOMEC 部門貢献賞受賞等．協調行動・強化学習に関する研究や水産物トレーサビリティ技術の開発等に従事．訳書として『強化学習』（森北出版，2000）等．JSME，RSJ，JSAI，JSFS 等の会員．