

アーム搭載移動ロボットの駆動系故障時のための 強化学習を用いたリカバリモーション獲得

伊藤 翼^{*1}, 河野 仁^{*2}, 田村 雄介^{*2}, 山下 淳^{*2}, 浅間 一^{*2}

Recovery Motion Learning for Single-Armed Mobile Robot in Drive System's Fault

Tauku ITO^{*1}, Hitoshi KONO^{*2}, Yusuke TAMURA^{*2},
Atsushi YAMASHITA^{*2} and Hajime ASAMA^{*2}

^{*1} Department of Precision Engineering, Faculty of Engineering, The University of Tokyo
7-3-1 Hongo, Bunkyo-ku, Tokyo 113-8656, Japan

^{*2} Department of Precision Engineering, Graduate School of Engineering, The University of Tokyo
7-3-1 Hongo, Bunkyo-ku, Tokyo 113-8656, Japan

This study presents a new approach to obtain recovery motion for an arm mounted teleoperated crawler robot in drive system's failure. A robot in drive system's fault and in dangerous areas such as disaster sites needs to move. When one side crawler mechanism is in fault, the robot can use redundancies such as its arm for moving. It calls recovery motion in this study. However, it is difficult to know how to leverage these redundancies and to manipulate the robot. Our approach uses the reinforcement learning which makes robot do trial and error to maximize total reward it receives and finds the motion of purpose. To obtain the recovery motion, in advance by using reinforcement learning in the 3D dynamics simulator is effective for real robots. For reinforcement learning, three types of reward functions are used. In the 3D simulator, experiments on a crawler robot verified this approach.

Key Words : Crawler Robot, Learning, Recovery

1. 緒 言

近年、様々な遠隔操作のロボットが開発され、災害現場への投入により、多くの成果を挙げている⁽¹⁾⁽²⁾。2011年に発生した東日本大震災に起因する災害現場では、多くの遠隔操作のクローラロボットが投入された⁽³⁾。災害現場におけるがれき撤去作業などに活用するため、ロボットにはアームが取り付けられている⁽³⁾。

ロボットが投入される環境は危険が伴う環境であり、不測の事態に備える必要がある。そこで本研究では投入するロボットに故障が生じた場合について考える。ロボットが故障した場合には、他のロボットの作業の妨げにならない位置までの移動、修理のための帰還が必要である。クローラ以外が故障した場合、ロボットは移動し続けることが可能であるが、クローラの故障時は移動することが難しい。そこで、クローラの故障時における移動方法として、クローラ以外の残存機能

(アームなど)を活用した移動方法が有用である。本論文では、残存機能を活用したロボットの移動をリカバリモーションと呼ぶ。

クローラの故障はモータの故障や車輪、ベルト部分の破損など様々である。また、ロボットが実際に動作する環境は、傾斜や凹凸な路面など様々な環境が存在する。さらにはロボットにも様々な種類があり、例えばASTACO-SoRa⁽⁴⁾のように、複数のアームを持つロボットへの適用も考慮することが重要である。将来的に上記すべての要素に対応するリカバリモーションを作ることが重要になるが、すべての状況に適応できるリカバリモーションを設計することは難しい。そこで人間がモーションを設計するのではなく、事前にシミュレータを用いてリカバリモーションを学習しておくことが効果的である。本研究では学習手法の1つである、強化学習を用いる。

関連研究として六脚ロボットにおける強化学習を用いた故障時の移動法獲得⁽⁵⁾や強化学習とCPG (Central Pattern Generator) を用いた多脚ロボットの歩行動作獲得の研究⁽⁶⁾が挙げられる。また強化学習を用いた多脚ロボットの行動獲得⁽⁷⁾では、遺伝的アルゴリズムを組

^{*1} 東京大学工学部精密工学科 (〒 113-8656 東京都文京区本郷 7-3-1) ito@robot.t.u-tokyo.ac.jp

^{*2} 東京大学大学院工学系研究科精密工学科 (〒 113-8656 東京都文京区本郷 7-3-1) {kono, tamura, yamashita, asama}@robot.t.u-tokyo.ac.jp

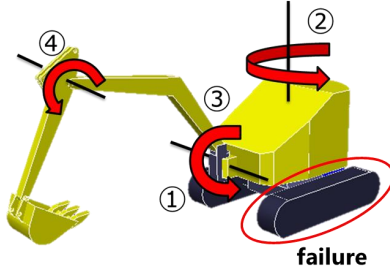


Fig. 1 Single-armed crawler robot

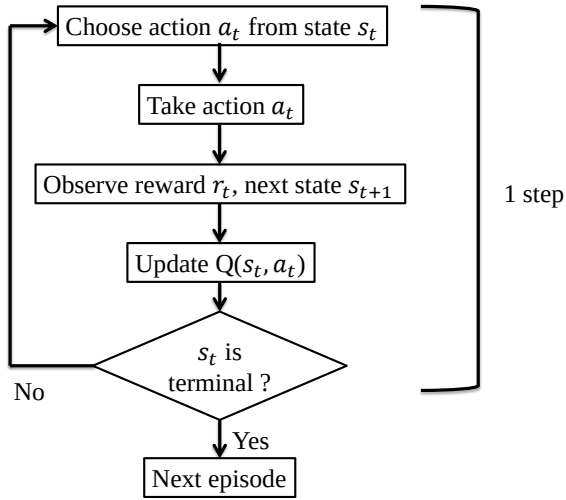


Fig. 2 Q-learning flow

み合わせた手法になっており、通常時の行動獲得のみならず、故障時においても行動獲得を行っている。以上の文献からもロボットの行動獲得に強化学習が有用であることがわかる。ロボットの移動する実際の災害現場は不整地であり、また、ロボットのリカバリモーションはアームを用いるため不安定になる可能性が想定される。そのため、リカバリモーションはロボットの移動中の安定性を考慮したものでなければならない。

本研究では、クローラロボットの故障時におけるリカバリモーションを、強化学習を用いて獲得する。基礎的検討として2つのクローラと1本のアームを搭載した単腕クローラロボット（図1）のリカバリモーション獲得手法を提案する。また、平らな地面における直進移動を考える。

2. アプローチ

2.1 前提条件 本論文では図1に示すようなアーム1本を搭載したクローラ型ロボットを用いる。ロボットの片方のクローラが故障により動かなくなった場合を想定する。ロボットの制御する機構はクローラ（①）、

スイング（②）、ブーム（③）、アーム（④）の4種類である（図1）。

2.2 強化学習 強化学習は、ロボットが試行錯誤を繰り返し、最適な行動を学習していく枠組みである⁽⁸⁾⁽⁹⁾。ロボットが確率的にある行動をとったときに、目的に合った行動をとると、報酬というスカラー量を得る。学習を進めることで、ロボットは報酬を最大化する行動をとるようになる。つまり強化学習を用いることにより、ロボットの目標に応じた報酬を与えることで環境に適応した動作を学習することができる。本研究では強化学習の中でも多くの研究で用いられているQ学習と、学習回数を増すごとに報酬の大きな行動が選択しやすくなるボルツマン選択を用いる。以下でQ学習とボルツマン選択について説明する。

2.3 Q学習 Q学習は初期状態からタスク完了を表す終端状態に至るまでの1エピソードの間に行動選択、価値関数の更新を繰り返す強化学習である（図2）。行動価値関数 $Q(s_t, a_t)$ を式(1)に示すTD (Temporal Difference) 誤差 δ を用いて、式(2)に示す更新を行う。 $Q(s_t, a_t)$ は状態 s_t において行動 a_t をとった場合の価値を表す関数であり、 r_t はそのときの報酬である。 $\max_a Q(s_{t+1}, a)$ は状態 s_{t+1} の時における最大行動価値関数 Q である。行動選択してから、価値関数の更新までを1ステップと呼び、エピソードが終わるまで繰り返される。1エピソードが終わると次のエピソードへ移る。

$$\delta = r_t + \gamma \max_a Q(s_{t+1}, a) - Q(s_t, a_t) \quad (1)$$

$$Q(s_t, a_t) \leftarrow Q(s_t, a_t) + \alpha \delta \quad (2)$$

ここで α は学習率、 γ は割引率である。

2.4 ボルツマン選択 ボルツマン選択は状態 s のときに行動 a ($s \in S, a \in A$) をとる確率 $\pi(s, a)$ を以下に示す式(3)によって定める行動選択法である。

$$\pi(s, a) = \frac{\exp(\frac{Q(s, a)}{T})}{\sum_{b \in A} \exp(\frac{Q(s, b)}{T})} \quad (3)$$

温度係数 T が非常に大きい場合、行動価値関数 Q の寄与が小さくなり、すべての行動が等確率に近づく。逆に T が小さい場合 Q の寄与が大きくなり、行動確率は Q の大きさに左右されやすくなる。今回は T をエピソードが進むにつれて0に近づく関数で以下のように定めた。

$$T = \frac{1}{\log(t+0.1)} \quad (4)$$

ただし t はエピソード数である。

2.5 報酬設計 報酬設計は強化学習の中でも最も重要な設定の1つである。本研究では環境が広く、ロボットの自由度が多いために学習の収束に多大な時間がかかると予想される。文献⁽⁵⁾⁽⁶⁾にもあるようにロボットの直進には移動ベクトル、目標地点への到達などが報酬として用いられる。さらに本研究ではすでに述べたようにロボットの転倒可能性についても考慮する。以上から学習を促すために以下の移動ベクトル報酬 $r_t^{(1)}$ 、ゴール報酬 $r_t^{(2)}$ の両方に加え、ロボットの安定余裕に対する報酬 $r_t^{(3)}$ を設計する。これらを時間 t における行動の報酬として (5) 式のように定める。

$$r_t = r_t^{(1)} + r_t^{(2)} + r_t^{(3)} \quad (5)$$

次項からそれぞれの報酬について説明する。

2.5.1 移動ベクトル Fig. 3 のようにロボットの向いている方向の単位ベクトル $\mathbf{e}$ と式 (6) に示す移動ベクトル $\mathbf{p}$ 、2つのなす角 θ ($-\pi \leq \theta \leq \pi$) を定める。式 (7) に示すように $\mathbf{p}$ の $\mathbf{e}$ に対する射影を d_1 とし、ベクトル $\mathbf{e}$ の方向を正とする。また式 (8) に示すように d_2 を定める。また時間ステップ t から時間ステップ $t+1$ におけるロボットの回転 ϕ_t を定める。

$$\mathbf{p} = [x_{t+1} - x_t \quad y_{t+1} - y_t]^T \quad (6)$$

$$d_1 = |\mathbf{p}| \cos \theta_t \quad (7)$$

$$d_2 = |\mathbf{p}| \sin \theta_t \quad (8)$$

これらの設定した値から (9) 式に示すように1ステップごとに直進の報酬 $r_t^{(1)}$ を設定する。

$$r_t^{(1)} = \eta d_1 - \lambda |d_2| - \tau |\phi_t| \quad (9)$$

ただし η と λ はそれぞれ正の係数である。右辺第1項によって、直進方向の移動ベクトルを評価し、第2項によって直進方向からのずれを評価している。また第3項によってロボットの向きのずれを評価する。

2.5.2 ゴール ロボットの前方 b [m] 先にゴールとゴールラインを定める (図4)。ロボットがゴールラインに到達するまでを1エピソードとし、ロボットは時間 $t = T$ においてゴールラインに到達すると以下の式にしたがって報酬が与えられる。 d_3 はゴールライン到着時のロボットとゴールとの間の距離である。

$$r_t^{(2)} = \begin{cases} \zeta \exp(-|d_3|) & \text{if robot reached goal line} \\ 0 & \text{otherwise} \end{cases} \quad (10)$$

ただし ζ は正の係数である。直進方向から離れるほ

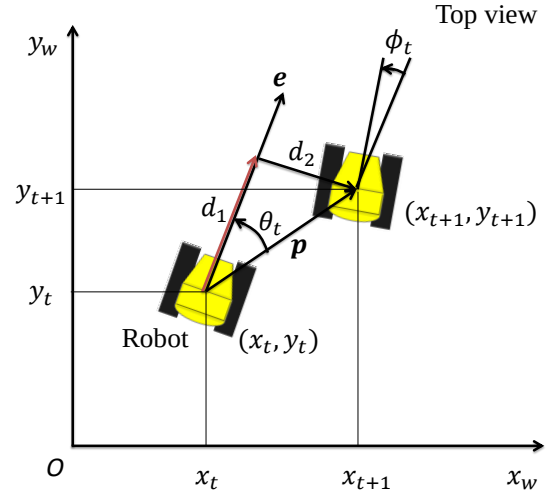


Fig. 3 Vector reward

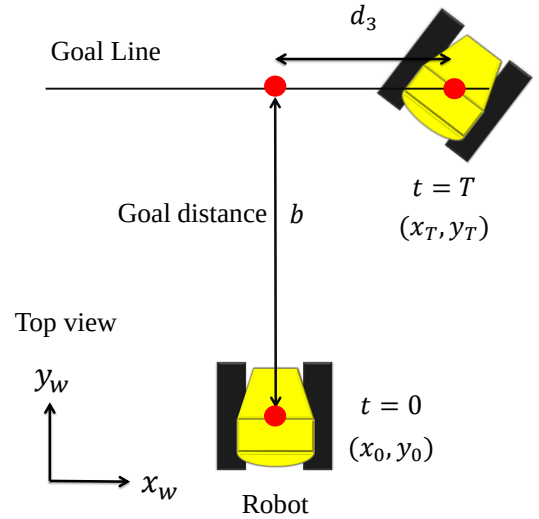


Fig. 4 Goal reward

ど少ない報酬しか得られないようにする。以上の設定をすることによりゴール到達までに行われたアームを用いた複合的な動きを評価できる。本研究ではゴールラインに到達するまでを1エピソードとする。ゴールライン到達後は、到達時点のロボットの向いている方向の b [m] 前方に再びゴールラインを作り、そのまま次のエピソードを開始する。

2.6 安定余裕 ロボットの移動する実際の災害現場は不整地であり、また、ロボットのリカバリモーションはアームを用いるため不安定になる可能性が想定される。そこで転倒安定性評価について報酬関数に組み込む。安定性評価として、正規化エネルギー安定余裕 (NE 安定余裕)⁽¹⁰⁾⁽¹¹⁾を用いる。本来、歩行機

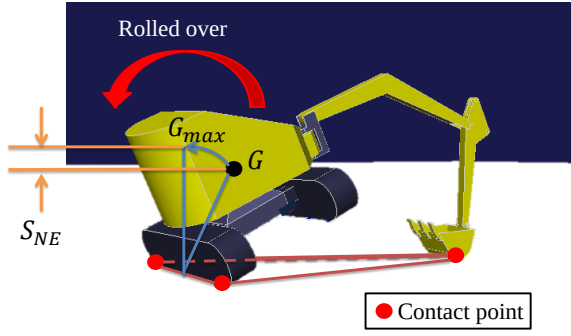


Fig. 5 Normalized energy stability margin

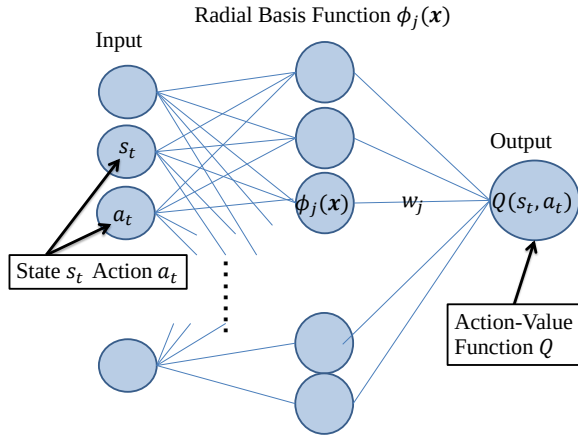


Fig. 6 RBF network

械の歩行中の安定性に関する評価方法であるが、本研究ではロボットのクローラ部分の前後をそれぞれ支持脚部分として仮定する。ロボットの重心の鉛直座標を z_g として、転倒するときの重心の鉛直座標の最高点を $z_{\max}$ とすると（図 5）、NE 安定余裕 S_{NE} は (11) 式で与えられる。

$$S_{NE} = z_{\max} - z_g \quad (11)$$

S_{NE} が大きいほど安定性が高い。さらに報酬関数として扱うために (12) 式のように負の報酬として設定する。

$$r_t^{(3)} = -\rho \exp(-S_{NE}) \quad (12)$$

1 ステップごとに NE 安定余裕に関して報酬を与える。転倒した場合、追加で大きな負の報酬を与え、初期位置へ戻し、次のエピソードへ移る。

2.7 関数近似 Q 学習では状態空間の広さから状態行動空間が指数的に増加する状態空間の爆発が発生することがある。そこで行動価値関数 Q 値の近似に Radial Basis Function (RBF) ネットワーク⁽¹²⁾（図 6）

Table 1 State s_t , Action a_t

Joint	Figure 1	State s_t	Action a_t
Right crawler	1	Torque (N · m)	0.3, 0, -0.3 m/s
Swing	2	Joint angle (rad)	0.3, 0, -0.3 rad/s
Boom	3	Joint angle (rad)	0.3, 0, -0.3 rad/s
Arm	4	Joint angle (rad)	0.3, 0, -0.3 rad/s

を用いる。入力値として状態値と行動値を与え、出力値に Q 値を与える。出力層を次の式に示す。

$$y = \sum_j w_j \phi_j(x) \quad (13)$$

RBF ネットワークは単純な 3 層ニューラルネットワークで、本論文では (14) 式に示す RBF のガウス関数を活性化関数としている

$$\phi_j(x) = \exp \left[-\sum_i \frac{(x_i - \mu_{ij})^2}{\sigma_j^2} \right] \quad (14)$$

ただし μ_{ij} , σ_j はそれぞれ j 番目の中間層ユニットの中心位置と標準偏差を表す。

次に RBF ネットワークのパラメータの更新について説明する。出力値 $Q(s_t, a_t)$ の更新後の値を $\hat{y}$ として式 (15) のように二乗誤差 E を定める。

$$E = \frac{1}{2}(\hat{y} - y)^2 \quad (15)$$

RBF ネットワークの出力の誤差を小さくするように、以下の更新式によって重み w_j , μ_{ij} を Q 値の更新ごとに更新する。 α_w , α_μ は更新率で、 α は Q 値の学習率である。

$$w_j \leftarrow w_j - \alpha_w \frac{\partial E}{\partial w_j} \quad (16)$$

$$\mu_{ij} \leftarrow \mu_{ij} - \alpha_\mu \frac{\partial E}{\partial \mu_{ij}} \quad (17)$$

これらの更新式は Q 値の更新のときに用いた TD 誤差 δ を用いて以下のように表される。

$$w_j \leftarrow w_j + \alpha_w \alpha^2 \delta \phi_j(x) \quad (18)$$

$$\mu_{ij} \leftarrow \mu_{ij} + \alpha_\mu \alpha^2 \delta w_j \frac{x_i - \mu_{ij}}{\sigma_j^2} \phi_j(x) \quad (19)$$

3. 計算機実験

3.1 動力学シミュレータ 本実験ではロボット用統合 GUI ソフトウェア Choreonoid⁽¹³⁾を利用する。Choreonoid は 3D 動力学シミュレーションエンジンを備えている⁽¹⁴⁾。計算機シミュレータの動作環境は Ubuntu 14.04 LTS, Intel Core i7-4720HQ 2.6GHz である。

Table 2 Experiment setup parameter

Parameter		Value
Learning rate	α	0.1
Discount rate	γ	0.9
Vector reward	η	10
Vector reward	λ	0.5
Vector reward	τ	0.01
Goal reward	ζ	10
Goal distance	b	1.5
Rolled over	ρ	1.0
RBF update rate	α_w	0.1
RBF update rate	α_μ	0.001

3.2 実験環境 ロボットは図 1 に示す外形で全長 1.6m（アームを除く）、幅 1.25m、高さ 1.36 m である。ロボットから見て左のクローラの故障を想定する。今回は状態と行動の入力を表 1 のように定める。制御対象 4 種類に対して状態 s 、行動 a の合計 8 つの値と、状態 s にロボットのロールとピッチの値を加えた合計 10 種類の値を、0 から 1 に正規化したうえで入力値とし、Q 値を出力する。ただし行動値 a に関しては表に示すような出力速度を PID 制御によって実現する。RBF の基底関数 ϕ は 20 個設定する。強化学習のパラメータ、RBF のパラメータの設定を表 2 に示す。転倒時の報酬を -10 とした。1 エピソードを 1.5m の直進として定める。RBF の初期値に関しては、 μ_{ij} は 0 から 1 の範囲でランダムに割り当て、 σ_j には 1.0 を与えた。試行エピソード回数は 35 回とし、シミュレータ上で実験を行う。

4. 実験結果

実験結果としてエピソード 1 とエピソード 29 のロボットの移動軌跡を図 7 で示す。ロボットの初期位置はワールド座標系で $(x, y) = (0, 0)$ であり、ベクトル $(0, -1)$ の方向を向いている。ゴールラインは $y = -1.5$ の直線である。2 つの移動軌跡を比べると、エピソード初期に見られた、試行錯誤している移動経路が少なくなり、ロボットの移動が直進に近づいていることが見て取れる。1 エピソードの総報酬値の収束の様子を図 8 に示す。時間と共に総報酬が増加していることがわかる。観測されたロボットの動きを図 9 に示す。故障している左クローラ側にアームを回し（図 9(a)）、左クローラを浮かすように突き（図 9(b)）、右クローラを動かすことで直進する動作が見られた（図 9(c)(d)）。

図 7 の移動軌跡では直進の移動軌跡としてはまだ改善の余地があると考えられるが、直進方向への移動が可能となっており、提案手法の有効性が確認できた。

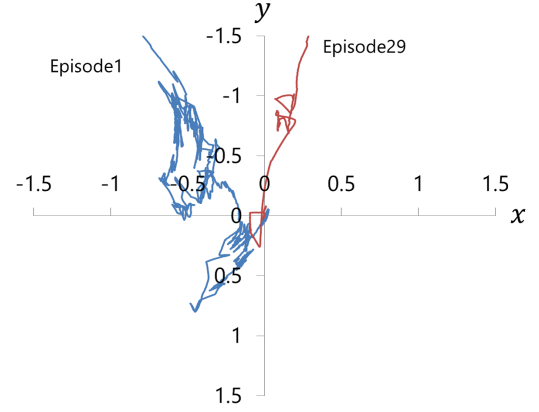


Fig. 7 Robot's trajectory

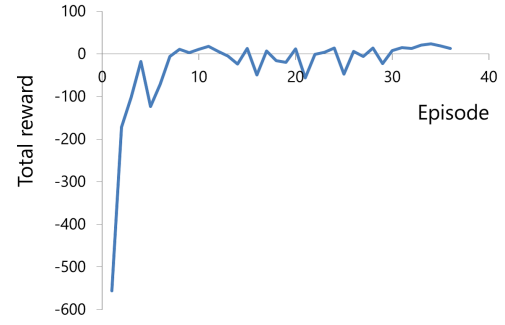
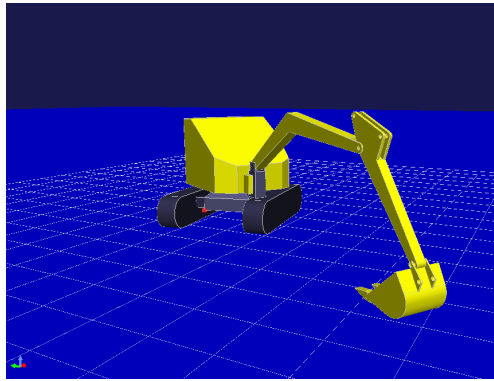


Fig. 8 Total reward

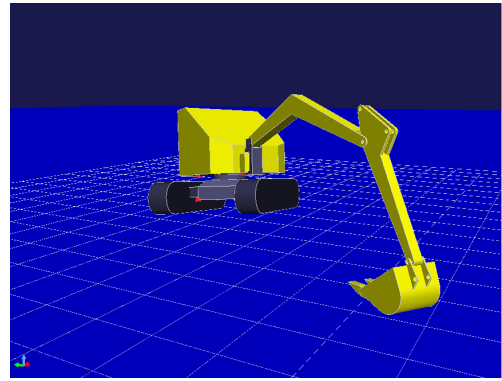
5. 結 言

本研究ではアーム搭載クローラロボットにおける片側クローラ故障時を想定した直進方向のリカバリモーション生成を目標とし、動力学シミュレータと強化学習によるモーション獲得を行った。移動ベクトル、ゴール、ロボットの安定性について報酬を設定し、学習を促す報酬設計法を提案し、計算機実験により評価した。実験結果から、本研究で設定した報酬設計を用いることで、強化学習によりクローラ故障時における直進移動モーションの獲得が可能であった。

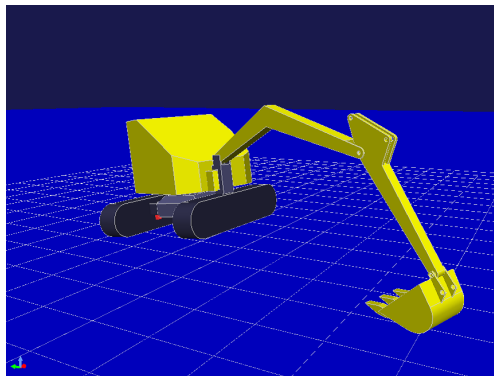
今後の課題として、学習回数を増やすことを始め、学習アルゴリズムに関係するパラメータチューニングが必要である。また、本研究は不整地で活躍する災害対応ロボットを想定している。そのため複数アクチュエータを用いた場合や複数環境下における提案手法の有用性を確認し、実機による評価も必要である。



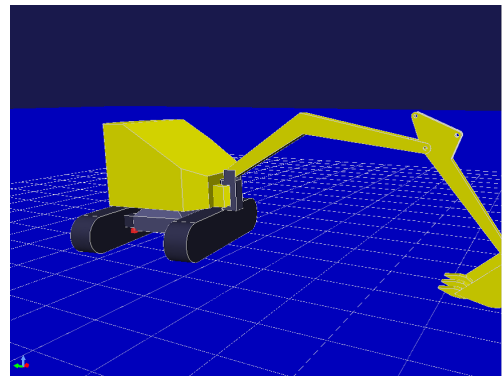
(a)



(b)



(c)



(d)

Fig. 9 Recovery motion of straight movement

謝 辞

本研究の一部は、総合科学技術・イノベーション会議により制度設計された革新的研究開発促進プログラム (ImPACT)「タフ・ロボティクス・チャレンジ」の援助を受けた。

参 考 文 献

- (1) 浅間一, “災害時に活用可能なロボット技術の研究開発と運用システムの構築”, 日本ロボット学会誌, Vol.32, No.1 (2014), pp. 37–41.
- (2) 田所諭, “防災ロボットについて我が国が取り組むべき中長期的課題”, 日本ロボット学会誌, Vol.32, No.2 (2014), pp.154–161.
- (3) 久武経夫, 中里邦子, “大災害に立ち向かうロボットの開発”, 建設の施工企画, Vol.740 (2011), pp.42–47.
- (4) 江川栄治, “小型双腕住器型ロボット「ASTACO-SoRa」”, 日本機械学会誌, Vol.117, No.1151 (2014), pp.682–683.
- (5) 新堀航太, 兵頭和幸, 砂山亨祐, 三上貞芳, “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”, 情報処理学会論文誌, Vol.50, No.3 (2009), pp.1170–1180.
- (6) 石倉裕貴, 岸本良一, 堀内匡, “CPG と強化学習を用いた多脚ロボットの目標到達行動の獲得”, 電気学会論文誌 C, Vol.136, No.3 (2016), pp.333–339.
- (7) 伊藤一之, 松野文俊, “QDSEGA による多足ロボットの歩行運動の獲得”, 人工知能学会論文誌, Vol.17, No.4 (2002), pp.363–372.
- (8) R. S. Sutton, A. Gbarito 著, 三上貞芳, 皆川雅章, 強化学習, 森北出版, (2000).
- (9) 木村元, 宮寄和光, 小林重信, “強化学習システムの設計指針”, 計測と制御, Vol.38, No.10 (1999), pp.618–623.
- (10) 広瀬茂男, 塚越秀行, 米田完, “不整地における歩行機械の静的安定性評価基準”, 日本ロボット学会誌, Vol.16, No.8 (1998), pp.1076–1082.
- (11) D. A. Messuri, and C. A. Klein, “Automatic body regulation for maintaining stability of a legged vehicle during rough-terrain locomotion”, *IEEE Journal on Robotics and Automation*, Vol.1, No.3(1985), pp.132–141.
- (12) J. Platt, “A Resource-Allocating Network for Function Interpolation”, *Neural Computation*, Vol.3, No.2 (1991), pp.213–225.
- (13) 中岡慎一郎, “拡張可能なロボット用統合 GUI 環境 Choreonoid”, 日本ロボット学会誌, Vol.31, No.3 (2013), pp.12–17.
- (14) 中村晋也, 吉灘裕, 倉鋪圭太, 谷本貴頌, 近藤大祐, “複合ロボットのための動力学シミュレータの開発”, 日本機械学会ロボティクス・メカトロニクス講演会 2016 講演論文集, No.16-2 (2016).