

学習最適制御に基づく軌道学習と身体パラメータ調整による最適歩容生成

佐藤 訓志*・藤本 健治**・佐伯 正美*

Optimal gait generation via trajectory learning and robot parameter tuning
based on learning optimal control

Satoshi SATOH*, Kenji FUJIMOTO** and Masami SAEKI*

This paper concerns an optimal gait generation with respect to energy consumption by learning trajectory and adjusting robot parameters based on learning optimal control. In this method, learning optimal control of Hamiltonian systems, which unifies learning control and parameter tuning, plays a key role. It allows one to simultaneously obtain an optimal trajectory and tuning parameters for a plant system, which (at least locally) minimize a given cost function. The proposed method is applied to the compass gait biped on a shallow slope and the one with a torso on the level ground, respectively. Consequently, a passive dynamic walking is generated for the first case, and an energy-efficient walking trajectory is generated for the latter case.

Key Words: Gait generation, Biped robots, Iterative learning control, Iterative feedback tuning, Hamiltonian systems

1. はじめに

筆者らはこれまでに、あるクラスの評価関数を(局所的に)最小化する意味での最適歩行軌道の生成に関する研究を行い、力学系の性質を利用した反復学習制御^{1),2)}に基づく方法を提案してきた^{3),4)}。この反復学習制御法^{1),2)}は、試行実験の反復により最適軌道を生成するフィードフォワード入力⁵⁾が獲得できる。さらに、ハミルトン力学系の変分対称性¹⁾という性質を利用することで、制御対象の詳細な情報を必要としない利点をもつ。しかしながら、出力の時間微分(機械系の場合には一般化速度に対応する)を含む評価関数や、離散的な状態遷移を伴う運動は扱えないなどの理由から、この手法をそのまま最適歩容生成問題には適用できないため、文献3)~5)において拡張・改良を行った。本研究の目的でもある消費エネルギーが最小という意味での最適な歩行として、受動歩行⁶⁾がよく知られており、これを規範とした歩容生成手法が数多く報告されている^{7)~10)}。その他にも、確率論に基づく探索により期待値の意味で最適な軌道を獲得する手法^{11),12)}や、制御対象のモデルを利用した最適化による方法¹³⁾など様々なアプローチがある。

筆者らの結果^{3)~5)}は、反復学習制御の枠組みであり、口

ボットが1周期の歩行を終える毎に実験の初期化が必要となる。そこで文献14)では、文献15)で提案されたパラメータチューニング法と前述の反復学習制御の統合手法を提案し、歩行を継続しながら学習を行う繰返し制御¹⁶⁾型の最適歩容生成法へと応用した。文献15)の手法は、ハミルトン系の変分対称性を利用し、反復学習制御と同様のアルゴリズムで、フィードバック制御器の可調整パラメータを最適化することができる。本論文では、このパラメータチューニングと反復学習制御の統合手法のことを学習最適制御法とよぶ。学習最適制御法¹⁴⁾に基づく歩容生成はつぎの3ステップから成り、各試行実験を連続して行いながら、最適な周期歩行軌道に収束させることができる。ステップ1: フィードバックにより、ロボットの運動をある対称軌道上に拘束する仮想ポテンシャルエネルギーを付加し、転倒回避を達成する。ステップ2: 不連続な状態遷移を考慮した反復学習制御法⁴⁾を適用し、1歩毎に学習入力を更新する。ステップ3: それと同時に統合手法を用いて、パラメータチューニングを行い、学習進度に応じて自動的に仮想拘束の強さを表す拘束力パラメータを最適化する。これにより、ロボットは歩き続けながら、最終的に拘束の十分小さい最適な周期軌道が生成される。

筆者らのこれまでの手法では、ロボットの身体パラメータが一定の下で、学習を繰り返しながら最適な歩行軌道を生成してきた。しかしながら文献17)では、1脚受動走行ロボットの腰部の弾性要素の剛性を、適応制御を用いて調整することで、入力が零となる受動走行軌道を生成している。この他にも文献18),19)などにおいて、弾性要素を有するロボットの剛性を調整することで、エネルギー効率の高い歩行軌道を生

* 広島大学大学院工学研究院 東広島市鏡山 1-4-1

** 京都大学大学院工学研究科 京都市左京区吉田本町

* Faculty of Engineering, Hiroshima University, 1-4-1 Kagamiyama, Higashi-Hiroshima

** Graduate School of Engineering, Kyoto University, Yoshida-Honmachi, Sakyo-ku, Kyoto

成する手法が提案されている。そこで本論文では、著者らのこれまでの結果¹⁴⁾において用いたパラメータチューニング法を、仮想拘束の強さの調節だけでなく、ロボットの身体パラメータの最適化にも適用することで、ロボットが歩行軌道を学習しながら最適な身体パラメータも同時に獲得できる手法を提案する。本研究で用いているパラメータチューニング法¹⁵⁾がもつ、複数の可調整パラメータを同時に扱えるという利点を活かすことで、提案手法は仮想拘束の強さの調整と身体パラメータの最適化が同時に達成できる。さらに、反復学習制御において入力飽和を扱う手法²⁾を応用することで、指定した範囲内で最適な身体パラメータを獲得する方法についても述べる。この方法は、例えば決められた脚長内での最適な質量配置や、指定した長さの範囲内での最適関節長の決定等に利用できる。先行研究では弾性要素をもつロボットの剛性を調節するものが多いが、本手法ではロボットの質量、質量分布、関節長なども最適化することができるため、最適軌道の学習と同時にロボットの最適設計を行うことも可能となる。

2. 準備

2.1 ハミルトン系と変分対称性

本論文では、制御対象として次式で表されるハミルトン系 $\Sigma^{x_i^0} : U \rightarrow Y : u \mapsto y$ を考える。

$$\Sigma^{x_i^0} : \begin{cases} \dot{x} = (J-R) \frac{\partial H(x, u, \rho)}{\partial x}^\top, & x(t^0) = x_{t^0} \\ y = -\frac{\partial H(x, u, \rho)}{\partial u}^\top \end{cases} \quad (1)$$

$x(t) \in \mathbb{R}^n, u \in U, y \in Y$ はそれぞれ状態、入力、出力を表す。 $H(x, u, \rho) \in \mathbb{R}$ はハミルトン関数を、 $\rho \in \mathbb{R}^s$ は制御対象の可調整パラメータを表す。ただし、 $U=Y=L_2^m[t^0, t^1]$ とする。また、 $J, R \in \mathbb{R}^{n \times n}$ はそれぞれ歪対称、半正定対称行列である。

本章では、ハミルトン系の変分対称性に基づく反復学習制御^{1), 2)}とパラメータチューニング¹⁵⁾に共通する概要を述べ、その後変分対称性に関するいくつかの結果¹⁾を示す。これらの手法は、与えられた評価関数の入力 u (または可調整パラメータ ρ) に関する勾配を計算し、その最急降下方向へと u (または ρ) を更新することで、逐次的に最適化を行う。勾配の計算には変分随伴系 $(\delta\Sigma^{x_i^0})^*$ が現れるため、一般的に制御対象 $\Sigma^{x_i^0}$ の詳細な情報が必要となるが、ハミルトン系の変分対称性より $(\delta\Sigma^{x_i^0})^*$ が変分系 $\delta\Sigma^{x_i^0}$ の情報から計算できるようになる。簡単にいうと変分系と変分随伴系とは、制御対象を入出力の作用素と考えたときの、Fréchet 微分^(注1)と、その随伴作用素を表す系である。そして、後述する条件の下で $(\delta\Sigma^{x_i^0})^*$ と $\delta\Sigma^{x_i^0}$ の動特性が互いの時間反転と一致する性質である変分対称性¹⁾を利用し、 $\Sigma^{x_i^0}$ の入出力情報のみから成る更新則を導出する。

(注1) 作用素 f の Fréchet 微分 δf とは、各 ξ に対して $\delta f(\xi)(\eta) = f(\xi + \eta) - f(\xi) + o(\|\eta\|)$ を満たす η に関して線形な作用素をいう。

補題 1.¹⁾ ハミルトン系 $\Sigma^{x_i^0}$ (1) について、 J, R は定数行列であり、正則行列 $T \in \mathbb{R}^{n \times n}$ が存在して次式を満たすとする。ただし、 $0, I$ は適当な次元の零行列、単位行列を表わす。

$$J = -TJ T^{-1}, \quad R = TR T^{-1} \quad (2)$$

$$\frac{\partial^2 H(x, u, \rho)}{\partial(x, u)^2} = \begin{bmatrix} T & 0 \\ 0 & I \end{bmatrix} \frac{\partial^2 H(x, u, \rho)}{\partial(x, u)^2} \begin{bmatrix} T^{-1} & 0 \\ 0 & I \end{bmatrix} \quad (3)$$

さらに $J - R$ が正則ならば、 $\Sigma^{x_i^0}$ の変分随伴系 $(\delta\Sigma^{x_i^0})^*$ は変分系 $\delta\Sigma^{x_i^0}$ を時間反転した状態空間表現をもつ。この性質を変分対称性とよぶ。

定義 1. 配位座標 q と一般化速度 \dot{q} が、 $\forall t \in [t^0, t^1]$ において

$$q(t) = q(t^1 - t + t^0), \quad \dot{q}(t) = -\dot{q}(t^1 - t + t^0) \quad (4)$$

を満たす軌道は、時間区間 $[t^0, t^1]$ の中心 $t = (t^0 + t^1)/2$ に関して対称な運動を表す。本論文ではこれを対称軌道とよぶ。

定理 1.²⁾ 補題 1 の仮定を満たす (1) 式のハミルトン系を考える。ある入力 u に対する状態の軌跡 x が対称軌道であるとき、任意の $v \in L_2^m[t^0, t^1]$ に対して、 $\Sigma^{x_i^0}$ の変分随伴系 $(\delta\Sigma^{x_i^0})^*$ と変分系 $\delta\Sigma^{x_i^0}$ の間に次式が成立する。

$$(\delta\Sigma^{x_i^0}(u))^*(v) = \mathcal{R} \circ (\delta\Sigma^{x_i^0}(u)) \circ \mathcal{R}(v) \quad (5)$$

ただし、 \circ は合成写像を表し、 \mathcal{R} は次式で定義される時間区間 $[t^0, t^1]$ 上での時間反転作用素を表している。

$$\mathcal{R}(u)(t) = u(t^1 - t + t^0), \quad \forall t \in [t^0, t^1] \quad (6)$$

注意 1. 変分対称性に基づく反復学習制御、パラメータチューニング法は、システムの軌道が対称軌道ではない一般的な場合においても適用できる。このような場合における (5) 式の一般化に関しては、文献 2) で述べられている。

2.2 学習制御とパラメータチューニングの統合手法

前節で述べた学習制御とパラメータチューニング法は、ハミルトン系の変分対称性を利用し、同様の方法で更新則が得られるが、これらの手法は互いに干渉するため、同時に使用することはできなかった。簡単に述べると、この干渉の問題はパラメータチューニングを適用すると制御対象の動特性が変化するが、従来の反復学習制御法は試行実験の途中での制御対象の変化は想定していないために起こる。本節で概要を述べる学習最適制御法¹⁴⁾は、変分対称性を保存するある拡大系を構成することでこの問題を解決しており、この方法により反復学習制御とパラメータチューニングを同時に利用できることから、本論文において重要な役割を果たす。

まずは文献 15) と同様に、0 次ホールド作用素 $h : \mathbb{R}^s \rightarrow L_2^s[t^0, t^1]$ を用いて制御対象 (1) の可調整パラメータ ρ に対応する仮想入力 $u_\rho \in U_\rho = L_2^m[t^0, t^1]$ と、対応する仮想出力 $y_\rho \in Y_\rho = L_2^s[t^0, t^1]$ をそれぞれ次式で定義する。

$$u_\rho(t) := (h(\rho))(t) \equiv \rho, \quad y_\rho := -\frac{\partial H(x, u, u_\rho)}{\partial u_\rho}^\top \quad (7)$$

つぎに、拡大入力を $u_e := (u^\top, u_\rho^\top)^\top \in U_e = U \times U_\rho$, 拡大出力を $y_e := (y^\top, y_\rho^\top)^\top \in Y_e = Y \times Y_\rho$ と定義する. すると, u_e から y_e への動特性 $\Sigma_e^{x_{t^0}} : U_e \rightarrow Y_e : u_e \mapsto y_e$ は, 新たにハミルトン関数を $H_e(x, u_e) := H(x, u, u_\rho)$ と定義することで次式となる.

$$\Sigma_e^{x_{t^0}} : \begin{cases} \dot{x} = (J - R) \frac{\partial H_e(x, u_e)}{\partial x}^\top, & x(t^0) = x_{t^0} \\ y_e = -\frac{\partial H_e(x, u_e)}{\partial u_e}^\top \end{cases} \quad (8)$$

(1), (8) 式より, このように構成した拡大系 $\Sigma_e^{x_{t^0}}$ は元のハミルトン系の構造を保存する. そのため, $\Sigma_e^{x_{t^0}}$ において制御入力 u を反復学習制御で, パラメータ ρ をパラメータチューニングで別々に最適化する際に生じる干渉の問題を, 拡大系 $\Sigma_e^{x_{t^0}}$ の制御入力 u_e に関する反復学習制御に置き換えることで解消できる. 元の系における干渉の影響は, $\Sigma_e^{x_{t^0}}$ の随伴変分系 $(\delta \Sigma_e^{x_{t^0}})^*$ に反映されるため $(\delta \Sigma_e^{x_{t^0}})^*$ よりも構造が複雑になる. しかし, $\Sigma_e^{x_{t^0}}$ がもつ変分対称性により結局 $(\delta \Sigma_e^{x_{t^0}})^*$ は直接計算する必要がなく, 最適化が達成できることがこの手法の核である.

つぎに学習最適制御法の更新則を導出する. 反復学習制御^{1), 2)}とパラメータチューニング¹⁵⁾が扱える評価関数はそれぞれ u, y と, u_ρ, y_ρ に関する汎関数であるため, ここではつぎの評価関数 $\hat{\Gamma}_e(u_e, y_e) : U_e \times Y_e \rightarrow \mathbb{R}$ を考える.

注意 2. 疑似共役微分作用素を用いた拡張手法³⁾により, \dot{y}_e を含む評価関数も扱うことができるが, 詳細は省略する.

拡大入力 u_e の定義から, 入力変分 δu_e は次式のようになる.

$$\delta u_e = \begin{bmatrix} \delta u \\ \delta u_\rho \end{bmatrix} = \begin{bmatrix} \delta u \\ \delta h(\rho) d\rho \end{bmatrix} = \begin{bmatrix} \delta u \\ h(d\rho) \end{bmatrix} \quad (9)$$

最後の等式は 0 次ホールド作用素 h の線形性による. (9) 式と $\delta y_e = (\delta \Sigma_e^{x_{t^0}}(u_e))(\delta u_e)$ の関係を用いて, 勾配を求めるために $\hat{\Gamma}_e$ の Fréchet 微分を計算すると次式を得る.

$$\begin{aligned} & \delta \hat{\Gamma}_e(u_e, y_e)(\delta u_e, \delta y_e) \\ &= \langle \nabla_{u_e} \hat{\Gamma}_e, \delta u_e \rangle_{U_e} + \langle \nabla_{y_e} \hat{\Gamma}_e, \delta y_e \rangle_{Y_e} \\ &= \left\langle \begin{bmatrix} \text{id} & 0 \\ 0 & h^* \end{bmatrix} \left(\nabla_{u_e} \hat{\Gamma}_e + (\delta \Sigma_e^{x_{t^0}}(u_e))^* (\nabla_{y_e} \hat{\Gamma}_e) \right), \begin{bmatrix} \delta u \\ d\rho \end{bmatrix} \right\rangle_{U \times \mathbb{R}^s} \\ &= \left\langle \begin{bmatrix} \nabla_{\bar{u}} \hat{\Gamma}_e \\ h^*(\nabla_{u_\rho} \hat{\Gamma}_e) \end{bmatrix} + \begin{bmatrix} \text{id} & 0 \\ 0 & h^* \end{bmatrix} \mathcal{R} \circ (\delta \Sigma_e^{x_{t^0}}(u_e)) \right. \\ & \quad \left. \circ \mathcal{R}(\nabla_{y_e} \hat{\Gamma}_e), \begin{bmatrix} \delta u \\ d\rho \end{bmatrix} \right\rangle_{U \times \mathbb{R}^s} \\ &\approx \left\langle \begin{bmatrix} \nabla_{\bar{u}} \hat{\Gamma}_e \\ h^*(\nabla_{u_\rho} \hat{\Gamma}_e) \end{bmatrix} + \begin{bmatrix} \mathcal{R} & 0 \\ 0 & h^* \end{bmatrix} \times \right. \\ & \quad \left. \left(\frac{\Sigma_e^{x_{t^0}}(u_e + \epsilon_e \mathcal{R}(\nabla_{y_e} \hat{\Gamma}_e)) - \Sigma_e^{x_{t^0}}(u_e)}{\epsilon_e} \right), \begin{bmatrix} \delta u \\ d\rho \end{bmatrix} \right\rangle_{U \times \mathbb{R}^s} \end{aligned} \quad (10)$$

ただし, id は恒等写像を表し, $\nabla_{u_e} \hat{\Gamma}_e(u_e, y_e)$ と $\nabla_{y_e} \hat{\Gamma}_e(u_e, y_e)$ はそれぞれ $\hat{\Gamma}_e(u_e, y_e)$ の u_e と y_e に関する偏勾配を表す. これ

らは, 評価関数が与えられれば直ちに計算できる. 2.1 節の説明と対応させながら (10) 式を順に説明する. 第三式は (5) 式の変分対称性を利用して, 第二式の変分随伴系 $(\delta \Sigma_e^{x_{t^0}}(u_e))^*$ への入力 $\nabla_{y_e} \hat{\Gamma}_e$ に対する出力信号を, 変分系 $\delta \Sigma_e^{x_{t^0}}(u_e)$ に時間反転した入力 $\mathcal{R}(\nabla_{y_e} \hat{\Gamma}_e)$ を入れたときの出力信号の時間反転で表したものである. 変分系の出力信号 $(\delta \Sigma_e^{x_{t^0}}(u_e))\mathcal{R}(\nabla_{y_e} \hat{\Gamma}_e)$ は, 制御対象 $\Sigma_e^{x_{t^0}}$ への入力 u_e に入力変分 $\mathcal{R}(\nabla_{y_e} \hat{\Gamma}_e)$ を加えたときの出力変分を表している. よって, これは変分を加える前後の出力の差分で近似でき, 第四式が得られる. ただし, 微小正定数 ϵ_e は入力変分を相対的に小さくし, 任意の精度で近似を成立させるために用いる. 第四式の計算には, $\mathcal{R}^* = \mathcal{R}$ (証明は文献 3) を参照) と, 文献 15) で示された

$$h^*(\xi) = \int_{t^0}^{t^1} \xi(t) dt, \quad \forall \xi \in L_2^s[t^0, t^1]$$

の関係から, $h^* \mathcal{R} = h^* \mathcal{R}^* = (\mathcal{R} h)^* = h^*$ を用いた.

本論文では次章から, 文献 14) で提案した方法をさらに拡張して, 軌道学習とロボットの身体パラメータの同時最適化による最適歩容生成を考える. この枠組の中でロボットは, 1 章でも述べたように学習が十分進み, 拘束力パラメータが小さくなるまで対称軌道へ拘束されることから, 各試行 (周期) においてロボットの軌道が対称軌道に近いものとなる. このとき, 更新則は次式で与えられる.

$$\begin{cases} u_{(2i+1)} = u_{(2i)} + \epsilon_{e(i)} \mathcal{R}(\nabla_{y_e} \hat{\Gamma}_{e(2i)}) \\ u_{\rho(2i+1)} = h(\rho_{(2i)}) + \epsilon_{e(i)} \mathcal{R}(\nabla_{y_\rho} \hat{\Gamma}_{e(2i)}) \\ \begin{cases} u_{(2i+2)} = u_{(2i)} - K_{(i)} \left(\nabla_{\bar{u}} \hat{\Gamma}_{e(2i)} + \frac{1}{\epsilon_{e(i)}} \mathcal{R}(y_{(2i+1)} - y_{(2i)}) \right) \\ \rho_{(2i+2)} = \rho_{(2i)} - K_{\rho(i)} \left(\int_{t^0}^{t^1} \nabla_{u_\rho} \hat{\Gamma}_{e(2i)} \right. \\ \quad \left. + \frac{1}{\epsilon_{e(i)}} \mathcal{R}(y_{\rho(2i+1)} - y_{\rho(2i)}) dt \right) \end{cases} \end{cases} \quad (11)$$

ただし, i は i 回目の試行実験を表す. 適当な正定行列 $K_{(i)} \in \mathbb{R}^{m \times m}$, $K_{\rho(i)} \in \mathbb{R}^{s \times s}$ はそれぞれ学習ゲインを表す. また, 初期状態 $x_{t^0(0)}$, 初期パラメータ $\rho_{(0)}$, 初期入力 $u_{(0)}$ は適当に定めるものとする. (11) 式より, 更新則は 2 回の試行実験で 1 ステップの学習となっており, これは (10) 式で変分系の出力を制御対象の二組の出力の差分から生成するためである.

3. 軌道学習と身体パラメータ調整による最適歩容生成

1 章で概要を述べた歩容生成法¹⁴⁾では, ステップ 2 の学習入力の更新による軌道学習と, ステップ 3 の仮想拘束の強さの調整を, 2.2 節で紹介した学習最適制御法を用いることで同時に達成していた. しかし, これまでの手法はロボットの身体パラメータが一定の下で学習を行っていたため, 本章では, パラメータチューニング法をこれまでのステップ 3 だけでなく, ロボットの身体パラメータの最適化にも適用することで, ロボットが歩行軌道を学習しながら最適な身体パラメータも同時に獲得できる手法を提案する. さらに, 反復学習制御において入力飽和を扱う手法²⁾を応用することで, 指定した範囲内で最適な身体パラメータを獲得する方法についても述べ

る。本論文では、二つの具体的な制御対象を考える。一つ目が緩斜面上のコンパスロボットであり、このロボットは適切な条件下では制御入力为零となる受動歩行を行うことが知られている。そこで、提案手法をこのロボットに適用することで、学習による最適軌道の獲得の過程で、能動歩行から受動歩行への遷移が起こるかを確認する。二つ目に、平地における胴体付きコンパスロボットを考え、能動歩行としての最適な歩行軌道の生成を目標とする。

3.1 歩行ロボットへの適用

本論文で扱う、傾斜 γ [rad] の緩斜面上を運動するコンパスロボットと平地における胴体付きコンパスロボットを **Fig. 1** に示す。Fig. 1 のそれぞれにおいて $u = (u_1, \dots, u_m)^\top$ は制

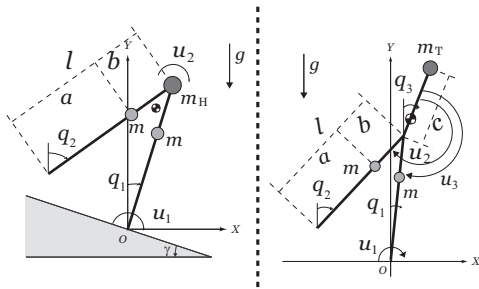


Fig. 1 The compass gait biped (left) and that with a torso (right)

御入力を表す。ただし、コンパスロボットの場合は $m = 2$ であり、胴体付きコンパスロボットの場合は $m = 3$ である。以降ではこれらの次元をいちいち記述することは避ける。適宜読み替えていただきたい。 u_j の具体的な説明は、3.2 節と 3.3 節でそれぞれ述べる。これらのロボットの動特性は、出力 $y = q^\top \in \mathbb{R}^m$ 、状態 $x = (q^\top, p^\top)^\top \in \mathbb{R}^{2m}$ ただし $p := M(q)\dot{q} \in \mathbb{R}^m$ 、次式で表されるハミルトン関数 $H(x, u)$ 、構造行列 J 、散逸行列 $R = 0$ とする (1) 式のハミルトン系として表現される。

$$H(x, u) = \frac{1}{2} p^\top M(q)^{-1} p + V(q) - u^\top q, \quad J = \begin{bmatrix} 0 & I \\ -I & 0 \end{bmatrix}$$

学習制御によるフィードフォワード入力は、ローカルなフィードバックで系を漸近安定化した後に参照入力として加えることが多い。その際、閉ループ系がハミルトン系 (1) の構造を保存する必要があるが、Fig. 1 のロボットを含む一般的な機械系は PD フィードバックがその条件を満たすことが知られている¹⁾。さらに、この系はもし慣性行列 M が q に依存しなければ、補題 1 において $T = \text{diag}(I, -I)$ と選ぶことで、仮定 (2), (3) が満たされる。さらに、慣性行列が q に依存する場合でも、PD ゲイン K_P, K_D を大きくとることで、近似的にこの仮定を満たすことが文献 1) に示されている。

つぎに、1 章のステップ 1 で述べた、ロボットの転倒回避のための仮想ポテンシャルエネルギーを導入する。文献 20) では、出力零化制御による $q_1(t) = -q_2(t), \forall t \in [t^0, t^1]$ を満た

す対称軌道への拘束により、転倒回避を保証しているが、この方法は制御対象の詳細な情報を必要とする。そこで本論文では、仮想ポテンシャルエネルギー

$$P_c := \frac{k_c}{2} (q_1 + q_2)^2 \quad (12)$$

の付加による拘束を考える。ポテンシャルエネルギーだけでは状態 $q_1 + q_2 = 0$ の不変多様体に閉じ込めることは保証されないが、学習の始めは拘束の強さを表す拘束力パラメータ k_c を十分大きく設定することで、文献 20) と同様の効果が期待できる。この方法の利点は、まず (12) 式で定められる P_c は P フィードバックにより制御対象の情報を用いることなく実現できることと、拘束力パラメータ k_c がハミルトン関数に陽に現れるため、これを可調整パラメータとみなすことで、2.1 節で述べたパラメータチューニングが適用でき、学習進度に応じて自動的に拘束力を調整できることにある。系の漸近安定化と仮想ポテンシャルを付加するための PD フィードバックを加えた新しい入力 \bar{u} は次式で与えられる。

$$u = -K_P q - K_D \dot{q} + \bar{u} - k_c A_c q, \quad (13)$$

仮想ポテンシャルを付加するためのフィードバックの構造を決める行列である $A_c \in \mathbb{R}^{m \times m}$ の具体的な説明は後述する。このとき、**Fig. 2** に示す閉ループ系は (1) 式の構造を保存し、次式で表されるハミルトン関数 H_c 、構造行列 J_c 、散逸行列 R_c を持つ新たなハミルトン系となる。

$$H_c(x, \bar{u}) = \frac{1}{2} p^\top M(q)^{-1} p + V(q) + \frac{1}{2} q^\top (K_P + k_c A_c) q - \bar{u}^\top q, \quad J_c = J, \quad R_c = \text{diag}(0, K_D) \quad (14)$$

本論文ではこの閉ループ系を考え、新しい入力 \bar{u} を学習入力として、後述する更新則に従い更新していく。

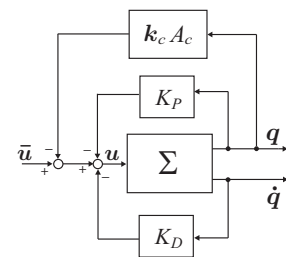


Fig. 2 Feedback system

本手法で用いるパラメータチューニング法¹⁵⁾は、ハミルトン関数に現れる任意のパラメータに適用できるため、(14), (21), (24) 式よりロボットの質量、質量分布、関節長などの様々な身体パラメータを軌道学習と同時に調整できる。本論文で調整するロボットの身体パラメータに関するパラメータチューニングを適用するための仮想入力 u_ρ と、それに対応する仮想出力 y_ρ の具体的な記述は、それぞれ 3.2 節と 3.3 節で述べる。ここからは $u_c := (\bar{u}^\top, u_\rho^\top)^\top$ として 2.2 節の手法を適用し、軌道学習と身体パラメータ調整による最適な歩

行軌道を生成する．駆動入力と拘束力を最小とする周期軌道を生成するために次式の評価関数を定義する．

$$\begin{aligned} \hat{\Gamma}(y_e, u_e) := & \quad (15) \\ & \frac{1}{2} \int_{t^0}^{t^1} (y(\tau) - C\mathcal{R}(y)(\tau))^\top \nu_1(\tau) \Lambda_y (y(\tau) - C\mathcal{R}(y)(\tau)) d\tau \\ & + \frac{1}{2} \int_{t^0}^{t^1} (\bar{u}(\tau)^\top \Lambda_{\bar{u}} \bar{u}(\tau) + u_\rho(\tau)^\top \Lambda_{u_\rho} u_\rho(\tau)) d\tau \end{aligned}$$

ただし， $C \in \mathbb{R}^{m \times m}$ は着地後の支持脚と遊脚との入れ替わりを表す交換行列であり，後述する．また，適当な正定行列 $\Lambda_y, \Lambda_{\bar{u}}, \Lambda_{u_\rho} \in \mathbb{R}^{m \times m}$ はそれぞれ対応する項の重みを表す．また， $\nu_1(t) \in \mathbb{R}$ は次式で定義されるフィルタ関数である．

$$\nu_1(t) := \begin{cases} \frac{1}{2} \left(1 - \cos \left(\frac{t^0 + \Delta t - t}{\Delta t} \pi \right) \right) & (t^0 \leq t \leq t^0 + \Delta t) \\ 0 & (t^0 + \Delta t < t \leq t^1) \end{cases}$$

Δt は適当な正定数であり，Fig. 3 に $\nu_1(t)$ の概形を示す． $\hat{\Gamma}$

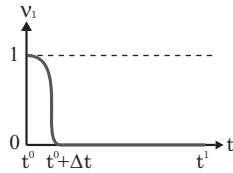


Fig. 3 Illustration of the filter function $\nu_1(t)$

中の各項の意味を簡単に述べる．交換行列 C と時間反転作用素 \mathcal{R} より，第一項は衝突直後の全関節角度が脚を入れ替えた初期関節角度と一致するという周期軌道の必要条件を表す拘束項である．文献4)の手法により，角速度に関する必要条件も同様に扱えるが，簡単のためここでは省略する．第二項は駆動入力と拘束力パラメータの大きさを抑える役割を果たす．

注意 3. (15) 式では，(13) 式の制御入力 u における第三項と第四項を最小化するが， u の L_2 ノルムは直接最小化できない．しかしながら，例えば評価関数に $\frac{\kappa_y}{2} \int_{t^0}^{t^1} y(\tau)^\top y(\tau) d\tau + \frac{\kappa_{\dot{y}}}{2} \int_{t^0}^{t^1} \dot{y}(\tau)^\top \dot{y}(\tau) d\tau$ のような拘束を加え，重み $\kappa_y, \kappa_{\dot{y}}$ を適切に定めることで，周期軌道の拘束を満たしながら $\|q\|_{L_2}, \|\dot{q}\|_{L_2}$ の大きさを抑えることができる ($y = q$ であることに注意)．これにより，間接的に $\|u\|_{L_2}$ を抑えることができる．

本論文では，ロボットの身体パラメータの最適化を行うが，それらは多くの場合とり得る値の範囲に制限がある．例えば Fig. 1 において，脚の質量配置パラメータ b は脚長 l を越えられないため $0 \leq b \leq l$ であり， l も正の実数であるから $l \geq 0$ という制限の中で最適化されなければならない．そこで本論文では，従来反復学習制御法で用いられていた入力飽和を扱う手法²⁾を，パラメータチューニングの仮想入力に適用し，飽和関数を適切に与えることで指定した範囲内でパラメータの最適化を行う．今仮想入力の指令値を $\hat{u}_\rho = h(\hat{\rho})$ とし，既知の微分可能な飽和関数 ϕ を用いて制御対象に働く実行入力が

$$u_\rho(t) = \phi(\hat{u}_\rho(t)) \quad (16)$$

とする．このとき， \hat{u}_ρ に対して変分対称性を導く出力 \hat{y}_ρ は，

$$\begin{aligned} \hat{y}_\rho &= -\frac{\partial H_c}{\partial \hat{u}_\rho}^\top \\ &= -\frac{\partial \phi(\hat{u}_\rho)}{\partial \hat{u}_\rho}^\top \frac{\partial H_c}{\partial u_\rho}^\top = \frac{\partial \phi(\hat{u}_\rho)}{\partial \hat{u}_\rho}^\top y_\rho \end{aligned} \quad (17)$$

となるため，(11) 式の更新則において仮想出力 y_ρ の代わりに $\partial \phi / \partial \hat{u}_\rho^\top y_\rho$ を用いればよいことがわかる．本論文では次式で与えられるような関数を飽和関数として用いる．

$$F_1(\alpha, \xi) := \begin{cases} 0 & (\xi < 0) \\ -\frac{\xi^3}{\alpha^2} + \frac{2\xi^2}{\alpha} & (0 \leq \xi < \alpha) \\ \xi & (\xi \leq \alpha) \end{cases} \quad (18)$$

$$F_2(\alpha, \xi) := \frac{\alpha}{2} \left(1 + \tanh \left(\frac{2\xi - \alpha}{\alpha} \right) \right) \quad (19)$$

さいごに，更新則を示す．ここでは計算の詳細は省略するが，(10)，(15) 式より， $\nabla_{\bar{u}} \hat{\Gamma}_e = \Lambda_{\bar{u}} \bar{u}$ ， $\nabla_{u_\rho} \hat{\Gamma}_e = \Lambda_{u_\rho} u_\rho$ ， $\nabla_y \hat{\Gamma}_e = (\text{id} - \mathcal{R}C) \nu_1 \Lambda_y (\text{id} - C\mathcal{R})(y)$ ， $\nabla_{y_\rho} \hat{\Gamma}_e = 0$ と計算でき，(11)，(16)，(17) 式から \bar{u} と u_ρ の更新則は次式となる．

$$\begin{cases} \bar{u}_{(2i+1)} = \bar{u}_{(2i)} + \epsilon_{e(i)} (\mathcal{R} - C) \nu_1 \Lambda_y (\text{id} - C\mathcal{R})(y_{(2i)}) \\ \hat{\rho}_{(2i+1)} = \hat{\rho}_{(2i)} \\ \rho_{(2i+1)} = \phi(\hat{\rho}_{(2i+1)}) \\ \bar{u}_{(2i+2)} = (\text{id} - K_{(i)} \Lambda_{\bar{u}})(\bar{u}_{(2i)}) - \frac{K_{(i)}}{\epsilon_{e(i)}} \mathcal{R}(y_{(2i+1)} - y_{(2i)}) \\ \hat{\rho}_{(2i+2)} = \hat{\rho}_{(2i)} - K_{\rho(i)} \left(\int_{t^0}^{t^1} \Lambda_{u_\rho} \hat{u}_{\rho(2i)} + \frac{1}{\epsilon_{e(i)}} \mathcal{R}(\hat{y}_{\rho(2i+1)} - \hat{y}_{\rho(2i)}) dt \right) \\ \rho_{(2i+2)} = \phi(\hat{\rho}_{(2i+2)}) \end{cases} \quad (20)$$

ただし， i はロボットの i 歩目における歩行時のデータを表す． $K_{(i)}, K_{\rho(i)} \in \mathbb{R}^{m \times m}$ はそれぞれ適当な正定行列であり，学習ゲインを表す．また，初期状態 $x_{t^0(0)}$ ，初期パラメータの指令値 $\hat{\rho}_{(0)}$ ，初期入力 $\bar{u}_{(0)}$ は適当に定めるものとする．

注意 4. 文献14)のように，(12) 式のような仮想拘束の強さのみを調整する場合は，制御対象(1)の情報は必要ない．しかし，ロボットの身体パラメータ ρ_i を調整する際は，更新則(20)中の \hat{y}_{ρ_i} の計算に必要な $y_{\rho_i} = -\frac{\partial H_c}{\partial u_{\rho_i}}$ ((7)，(17) 式参照)を導出するために H_c のシンボリックな情報(ここでは H_c に含まれるロボットパラメータの具体的な値を同定する必要はない)と， $y_{\rho_i}(t)$ の時系列データを計算するために $\frac{\partial H_c}{\partial u_{\rho_i}}$ に含まれるロボットパラメータの具体的な値が必要となる．この場合でも，運動方程式の導出といった動特性のモデル化は必要ない．機械系の場合，ハミルトン関数は各リンクの力学的エネルギーの和で与えられるため，シンボリックな情報は比較的容易に得られる．一方， H_c のどのパラメータの値が必要となるかは，調整パラメータ ρ_i に依存する．

注意 5. 本論文の定式化では身体パラメータ最適化の項が，

最小化を行う評価関数 (15) に含まれているが, 制御入力や拘束力と異なり, 身体パラメータの最適化は一般に最小化とは限らない. そこで本論文では, 身体パラメータの初期値を調整範囲の上限付近に設定して学習を行うことで, 評価関数の最小化により身体パラメータの最適化を行っている.

別の定式化として, Λ_{u_ρ} 中の身体パラメータに関する重みを 0 とし, 間接的に身体パラメータを最適化することも考えられる. 本手法では学習入力と可調整パラメータとの干渉を考慮した拡大系に対する変分対称性を用いて更新則を導出しているため, 身体パラメータに関する重みを 0 とした場合でも, (20) 式を用いて評価関数を間接的に最小化するためのパラメータの更新則が導出できる.

3.2 緩斜面上のコンパスロボットへの適用例

ここでは前節で提案した手法を, Fig. 1 左に示す傾斜 γ [rad] の緩斜面上を運動するコンパスロボットに適用する. まず, 制御入力である u_1, u_2 はそれぞれ足首, 股関節のトルクを表す. また, ハミルトン関数に現れる慣性行列 $M(q)$ とポテンシャルエネルギー $V(q)$ は次式で与えられる.

$$M(q) = \begin{bmatrix} m_H l^2 + m a^2 + m l^2 & -m b l \cos(q_1 - q_2) \\ -m b l \cos(q_1 - q_2) & m b^2 \end{bmatrix}$$

$$V(q) = \{(m_H l + m a + m l) \cos q_1 - m b \cos q_2\} g \quad (21)$$

このロボットに仮想ポテンシャルエネルギー (12) を加えるための, (13) 式のフィードバック行列 A_c は次式で与えられる.

$$A_c := \begin{bmatrix} 1 & 1 \\ 1 & 1 \end{bmatrix}$$

本節では簡単な例題として, 脚質量 m の分布位置を表すパラメータ b と, 脚の全長を表すパラメータ l を調整する場合を考える (それぞれ Fig. 1 左を参照). (7), (14), (21) 式を用いて, 仮想入力 u_ρ とそれに対応する仮想出力 y_ρ をそれぞれの場合について求めると次式となる.

(b を調整する場合)

$$u_\rho = (\mathfrak{h}(k_c), \mathfrak{h}(b))^\top$$

$$y_{\rho 1} = -\frac{\partial H_c}{\partial k_c} = -\frac{(q_1 + q_2)^2}{2}$$

$$y_{\rho 2} = -\frac{\partial H_c}{\partial b} = -m(l - b)\dot{q}_1^2 + m b \dot{q}_2^2 - m l \cos(q_1 - q_2)\dot{q}_1 \dot{q}_2 + m g(\cos q_1 + \cos q_2) \quad (22)$$

(l を調整する場合)

$$u_\rho = (\mathfrak{h}(k_c), \mathfrak{h}(l))^\top$$

$$y_{\rho 1} = -\frac{\partial H_c}{\partial k_c} = -\frac{(q_1 + q_2)^2}{2}$$

$$y_{\rho 2} = -\frac{\partial H_c}{\partial l} = (m_H l + 2m l - m b)\dot{q}_1^2 - m b \cos(q_1 - q_2)\dot{q}_1 \dot{q}_2 - (m_H + 2m)g \cos q_1 \quad (23)$$

さいごに, 評価関数 (15) に現れる, 着地後の支持脚と遊脚との入れ替わりを表す行列 C は次式で与えられる.

$$C := \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}$$

3.3 胴体付きコンパスロボットへの適用例

ここでは 3.1 節で提案した手法を, Fig. 1 右に示す平地における胴体付きコンパスロボットに適用する. まず, 制御入力である $u = (u_1, u_2, u_3)^\top$ について述べる. 地面と足首, 胴体と遊脚, 胴体と支持脚間の相対トルクをそれぞれ v_1, v_2, v_3 と表すと, 制御入力は $u := (v^1 - v^3, -v^2, v^2 + v^3)^\top$ で与えられる. 本論文では, ロボットの動特性を (1) 式で表した際に, 対応する入出力関係を簡単にするため, このような入力変換を用いた. また, ハミルトン関数に現れる慣性行列 $M(q)$ とポテンシャルエネルギー $V(q)$ は次式で与えられる.

$$M(q) = \begin{bmatrix} m_T l^2 + m l^2 + m a^2 & -m b l \cos(q_1 - q_2) & m_T c l \cos(q_1 - q_3) \\ -m b l \cos(q_1 - q_2) & m b^2 & 0 \\ m_T c l \cos(q_1 - q_3) & 0 & m_T c^2 \end{bmatrix} \quad (24)$$

$$V(q) = m g \{(a + l) \cos q_1 - b \cos q_2\} + m_T g \{l \cos q_1 + c \cos q_3\}$$

このロボットに (12) 式の仮想ポテンシャルエネルギーを加えるための, フィードバック行列 A_c は次式で与えられる.

$$A_c := \begin{bmatrix} 1 & 1 & 0 \\ 1 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix}$$

本節では, 脚質量 m の分布位置を表すパラメータ b と胴体の長さ c を同時に調整する. (Fig. 1 右を参照). (7), (14), (24) 式を用いて, 仮想入力 u_ρ とそれに対応する仮想出力 y_ρ をそれぞれの場合について求めると次式となる.

$$u_\rho = (\mathfrak{h}(k_c), \mathfrak{h}(b), \mathfrak{h}(c))^\top$$

$$y_{\rho 1} = -\frac{\partial H_c}{\partial k_c} = -\frac{(q_1 + q_2)^2}{2}$$

$$y_{\rho 2} = -\frac{\partial H_c}{\partial b} = -m(l - b)\dot{q}_1^2 + m b \dot{q}_2^2 - m l \cos(q_1 - q_2)\dot{q}_1 \dot{q}_2 + m g(\cos q_1 + \cos q_2) \quad (25)$$

$$y_{\rho 3} = -\frac{\partial H_c}{\partial c} = m_T l \cos(q_1 - q_3)\dot{q}_1 \dot{q}_3 + c m_T \dot{q}_3^2 - m_T g \cos q_3$$

さいごに, 評価関数 (15) に現れる, 着地後の支持脚と遊脚との入れ替わりを表す行列 C は次式で与えられる.

$$C := \begin{bmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

4. シミュレーション結果

本章では, 前章で得られた更新則を用いた数値結果を示す. 4.1 節では 3.2 節で述べた緩斜面上のコンパスロボットに提案手法を適用する. 続いて 4.2 節では, 3.3 節で述べた平地における胴体付きコンパスロボットを扱う.

以降のシミュレーションにおける各周期の積分区間 $[t^0, t^1]$ の設定は, 制御対象は時不変であるため, 各周期の開始時刻を t^0 とする. さらに, 学習の前に設定した初期状態と初期入力による運動からおよその歩行周期 T_p を算出し, $t^1 := t^0 + T_p$

としている。提案法では仮想拘束によりロボットの運動を対称軌道上に拘束しながら学習を行うため、各周期の T_p からのずれは小さいものとし、ある周期の歩行時間が T_p よりも大きいときは学習に用いる時系列データを T_p までで打ち切り、 T_p よりも小さいときは最終時刻のデータを t^1 までホールドすることで一定の積分区間における学習を実現している。

4.1 コンパスロボットの受動歩行への遷移

ロボットの物理パラメータは、 $m_H = 10, m = 5$ [kg], $l = 1.0$ [m] を用いた。本節では学習により最適軌道を獲得する過程において、能動歩行から受動歩行への遷移が生じるかどうか注目するため、(13) 式のフィードバックゲインは、 $K_P = 0, K_D = 0$ とした。学習に関する設計パラメータは、 $\Lambda_y = \text{diag}(0.5, 1), \Lambda_{\bar{u}} = \text{diag}(1, 1), \Lambda_{u_p} = \text{diag}(5, 2), \Delta t = 1.0 \times 10^{-2}, K_{(\cdot)} = \text{diag}(0.1, 0.1), K_{\rho(\cdot)} = \text{diag}(5.0 \times 10^{-3}, 1.0 \times 10^{-5}), \epsilon_{e(\cdot)} = 1$ を用いた。初期状態を $(q_{1t^0}, q_{2t^0}, \dot{q}_{1t^0}, \dot{q}_{2t^0}) = (-0.24, 0.27, 1.2, 0.35)$ とした。

本節では、以降で示す **Fig. 4** から **Fig. 8** において、左側に学習による入力トルクの最適化と同時に、脚質量の分布位置を表すパラメータ b を調整した結果を、右側に脚長 l を調整した結果を示す。身体パラメータ調整における仮想入出力 u_p, y_p はそれぞれ (22) 式と (23) 式を用いた。また、(16) 式の入力飽和関数として、 b の調整では $\phi = (\phi_1, \phi_2)^\top = (F_1(0.25, \cdot), F_2(l, \cdot))^\top$ を、 l の調整では $\phi = (F_1(0.25, \cdot), F_1(0.25, \cdot))^\top$ とした。ただし、飽和関数 F_1, F_2 の定義は (18), (19) 式を参照のこと。 b の調整では、初期脚質量位置 $b(0) = 0.65$ 、脚長は $l = 1$ で固定とし、 l の調整では、 $b = 0.65$ で固定とし、初期脚長は $l(0) = 1$ とした。(12) 式における初期拘束力パラメータはどちらも $k_c(0) = 5$ とした。初期学習入力をどちらの場合も零入力として、 b の調整では 500 ステップの学習、 l の調整では 130 ステップの学習、つまりそれぞれ 1000 歩と 260 歩の連続歩行を続けながら学習を行った。Fig. 4 は、左右の図のどちらも各学習ステップにおいて評価関数 (15) が単調に減少し、一定値になることから、速やかに(局所的)最適軌道に収束していることを示している。**Fig. 5** 上図から拘束力パラメータ k_c がどちらの場合も学習が進むに従い自動的に零になっていることがわかり、下図から調整した身体パラメータが最適値に収束していることもわかる。**Fig. 6** に各学習ステップにおける、(13) 式で与えられる全ての制御入力 $u_{(i)}$ のノルム $\|u_{(i)}\|_{L_2}$ を示す。これにより、制御入力の大きさがどちらの場合も零に収束しており、能動歩行から受動歩行への遷移が学習により達成されていることがわかる。**Fig. 7** の位相平面図において、点線が学習を始める前の自由運動を表しており、閉軌道になっておらず周期軌道ではないことがわかる。一方、実線が学習後に得られた最適軌道の図であり、周期軌道が得られており、さらに受動歩行特有の位相平面図を描いていることがわかる。さいごに、Fig. 8 に得られた受動歩行のアニメーションを示す。

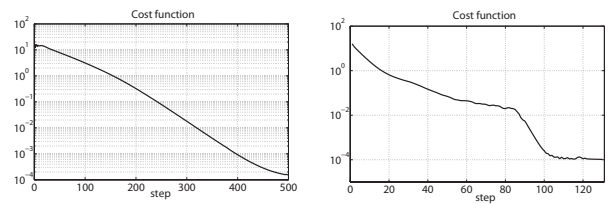


Fig. 4 Cost functions: with tuning b (left) and with tuning l (right)

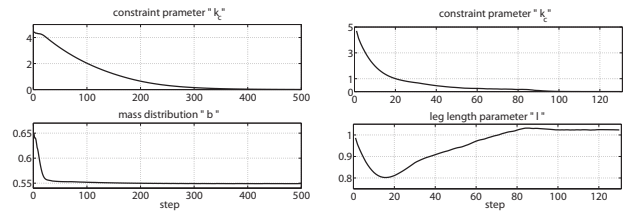


Fig. 5 Constraint parameter k_c and robot parameters: b (left) and l (right)

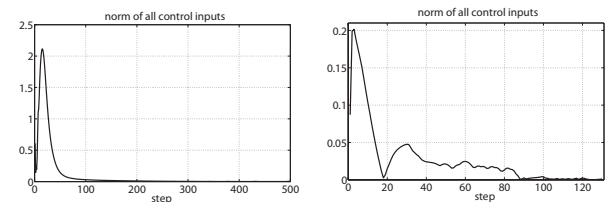


Fig. 6 Norm of the all control inputs u : with tuning b (left) and with tuning l (right)

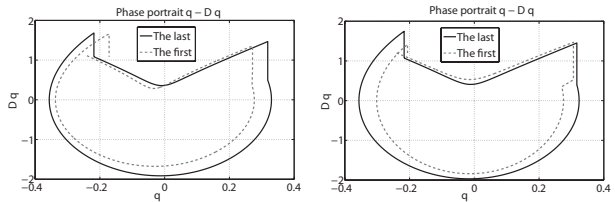


Fig. 7 Phase portraits w.r.t $q - \dot{q}$: with tuning b (left) and with tuning l (right)

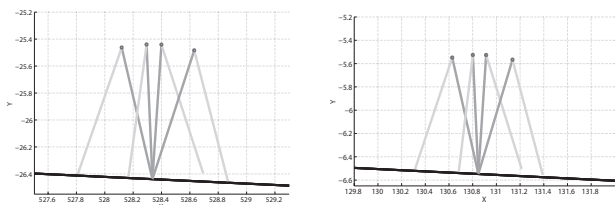


Fig. 8 Stick diagrams: with tuning b (left) and with tuning l (right)

4.2 胴体付きコンパスロボットの最適歩容生成

ロボットの物理パラメータは、 $m_T = 5, m = 1.2$ [kg], $l = 0.4, c = 0.3$ [m] を用いた。(13) 式のフィードバックゲインは、 $K_P = \text{diag}(4, 4, 10), K_D = \text{diag}(2, 2, 8)$ とした。ゲインの (3,3)-成分が比較的大きな理由は、胴体の不必要に大きな傾斜を防ぐためである。学習パラ

メータとして, $\Lambda_y = \text{diag}(10, 10, 10)$, $\Lambda_{\bar{u}} = \text{diag}(1 \times 10^{-4}, 1 \times 10^{-4}, 1 \times 10^{-4})$, $\Lambda_{u_\rho} = \text{diag}(1 \times 10^{-2}, 1 \times 10^{-3}, 1 \times 10^{-3})$, $\Delta t = 1.0 \times 10^{-2}$, $K_{(\cdot)} = \text{diag}(2, 2, 2)$, $K_{\rho(\cdot)} = \text{diag}(3, 1.5 \times 10^{-3}, 5.0 \times 10^{-4})$, $\epsilon_{e(\cdot)} = 1$ を用いた. 初期状態を $(q_{1t^0}, q_{2t^0}, q_{3t^0}, \dot{q}_{1t^0}, \dot{q}_{2t^0}, \dot{q}_{3t^0}) = (-0.18, 0.2, 0.0, 1.1, 0.5, 0.0)$ とした.

本節では, **Fig. 9** から **Fig. 13** において, 左側に身体パラメータの学習を行わず入力トルクの最適化のみを行った結果, 右側に入力トルクの最適化と, 脚質量の分布パラメータ b と胴体長 c を同時に調整した結果を示す. 身体パラメータ調整における仮想入出力 u_ρ, y_ρ は (25) 式を用いた. また, (16) 式の入力飽和関数として, $\phi = (\phi_1, \phi_2, \phi_3)^\top = (F_1(0.25, \cdot), F_2(l, \cdot), F_1(0.25, \cdot))^\top$ を用いた. 初期脚質量位置 $b_{(0)} = 0.3$, 初期胴体長は $c_{(0)} = 0.3$, 初期拘束力パラメータは $k_{c(0)} = 30$ とした. 傾斜による重力の推進力もなく, 胴体の影響もあるため, $k_{c(0)}$ は前節よりも大きな値に設定した. 一歩目からポテンシャルバリアを越えず, 歩行が停止して学習が継続されないことを防ぐため, 適当な一定値入力 $\bar{u}_{(0)}(t) = (0.5, 1.5, 0)$ を初期学習入力として加え, どちらも 800 ステップの学習, つまり 1600 歩の連続歩行を続けながら学習を行った. Fig. 9 より, どちらの場合も速やかに (局所的) 最適軌道に収束していることがわかる. **Fig. 10** 上図から拘束力パラメータ k_c がどちらの場合も学習が進むに従い自動的に十分小さな値へと収束しており, 右下図から調整した身体パラメータも最適化されている. **Fig. 11** に最終的に学習で得られた (13) 式の制御入力 u を示す. **Fig. 12** に位相平面図から, どちらも閉軌道になっており, 学習により周期軌道が得られていることがわかる. さいごに, Fig. 13 に得られた歩行のアニメーションを示す. 胴体傾斜による反トルクから発生する水平方向への推進力²¹⁾を積極的に利用するためか, どちらも胴体が傾斜している. しかし右図から, 胴体長の最適化を行うことで, 過度な胴体傾斜が抑えられ, Fig. 11 から胴体を支持するための入力 u_3 の大きさも減少していることがわかる. 身体パラメータ調整の効果を定量的に比較するため, 歩行効率の評価指標としてよく知られる Specific Resistance (SR)²²⁾ を計算した. これは, 単位質量を単位距離移動させるのに必要なエネルギーを表す. その結果身体パラメータ調整をしない場合は $SR = 5.27$, 調整も同時に行った場合は $SR = 1.88$ となり, 身体パラメータの同時最適化により移動効率が 2 倍以上も向上している. 受動歩行規範の手法と比べると依然として SR は大きい, これは漸近安定化のための PD フィードバックに因るものと考えられ, 注意 3 の手法などで u の大きさを押えることでさらなる改善が期待できる.

5. おわりに

本論文では, 筆者らがこれまでに提案した学習制御とパラメータチューニングの統合手法を応用し, 軌道学習とロボットの身体パラメータ調整による最適な歩行軌道の生成法を提案した. さらに, 従来反復学習制御において用いられていた

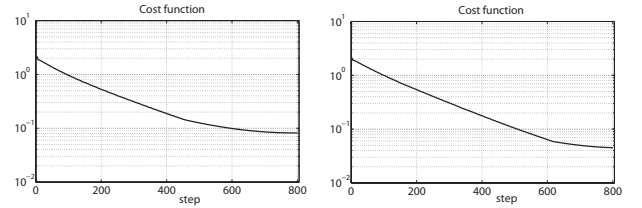


Fig. 9 Cost functions: without tuning (left) and with tuning b and c (right)

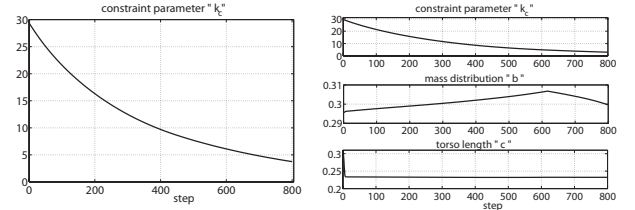


Fig. 10 Constraint parameter k_c (left), and k_c and robot parameters b and c (right)

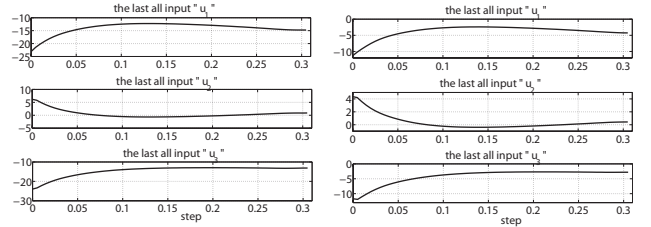


Fig. 11 The generated control inputs u : without tuning (left) and with tuning b and c (right)

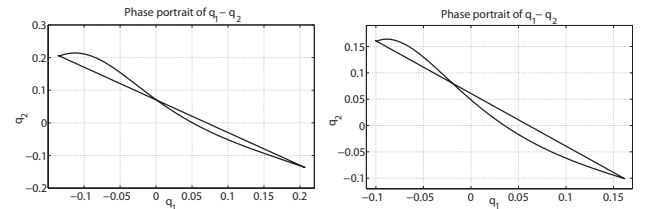


Fig. 12 Phase portraits w.r.t $q_1 - q_2$: without tuning (left) and with tuning b and c (right)

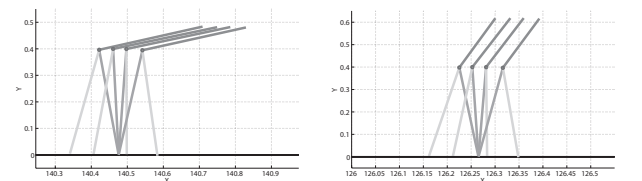


Fig. 13 Stick diagrams: without tuning (left) and with tuning b and c (right)

入力飽和を扱う手法を応用することで, 指定した範囲内で最適な身体パラメータを獲得できる方法についても述べた. 今後は, 歩行から走行のような, より動的な歩容遷移の発現や, より複雑なロボットへの適用について考えていきたい.

謝辞 本研究は科学研究費補助金 研究活動スタート支援 (No.22860041) の助成を受けました. ここに謝意を表します.

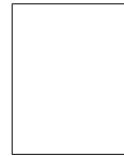
参考文献

- 1) K. Fujimoto and T. Sugie : Iterative learning control of Hamiltonian systems: I/O based optimal control approach, *IEEE Trans. Autom. Contr.*, **48**-10, 1756/1761 (2003)
- 2) 藤本, 堀内, 杉江 : ハミルトン系の変分対称性に基づく学習最適制御, システム制御情報学会論文誌, **21**-1, 10/17 (2008)
- 3) 佐藤, 藤本, 玄 : ハミルトン系の変分対称性に基づく1脚ロボットの最適歩容生成, 計測自動制御学会論文集, **43**-12, 1103/1110 (2007)
- 4) 佐藤, 藤本, 玄 : 不連続な状態遷移を考慮した学習最適制御による歩行軌道の生成手法, 日本ロボット学会誌, **29**-2, 90/100 (2011)
- 5) S. Satoh, M. Ikeda, K. Fujimoto and Y. Hayakawa : Modification of learning optimal gait generation method in considering discontinuous velocity transitions, *Proc. SICE 2010 Annual Conf.*, 2794/2799 (2010)
- 6) T. McGeer : Passive Dynamic Walking, *Int. J. Robotics Research*, **9**-2, 62/82 (1990)
- 7) A. Goswami, B. Thuilot and B. Espiau : Compass-like biped robot Part I: Stability and bifurcation of passive gaits, *INRIA Research Report*, -2996 (1996)
- 8) K. Osuka and K. Kirihara : Motion analysis and experiments of passive walking robot QUARTET II, *Proc. IEEE Int. Conf. Robotics and Automation*, 3052/3056 (2000)
- 9) A. Sano, Y. Ikemata and H. Fujimoto : Analysis of Dynamics of Passive Walking from Storage Energy and Supply Rate, *Proc. IEEE Int. Conf. Robotics and Automation*, 2478/2483 (2003)
- 10) F. Asano, M. Yamakita, N. Kamamichi and Z. W. Luo : A novel gait generation for biped walking robots based on mechanical energy constraint, *IEEE Trans. Robotics and Automation*, **20**-3, 565/573 (2004)
- 11) R. Tedrake, T. W. Zhang and H. S. Seung : Stochastic policy gradient reinforcement learning on a simple 3D biped, *Proc. IEEE/RSJ Int. Conf. Intelligent Robots and Systems*, 2849/2854 (2004)
- 12) E. Theodorou, J. Buchli and S. Schaal : A generalized path integral control approach to reinforcement learning, *J. Machine Learning Research*, **11**, 3153/3197 (2010)
- 13) J. Morimoto and C. Atkeson : Robust low torque biped walking using differential dynamic programming with a minimax criterion, *Proc. 5th Int. Conf. Climbing and Walking Robots*, 453/459 (2002)
- 14) S. Satoh, K. Fujimoto and S. Hyon : Gait generation via unified learning optimal control of Hamiltonian systems, *Robotica*, **31**-5, 717/732 (2013)
- 15) K. Fujimoto and I. Koyama : Iterative feedback tuning for Hamiltonian systems, *Proc. 17th IFAC World Congress*, 15678/15683 (2008)
- 16) S. Hara, Y. Yamamoto, T. Omata and M. Nakano : Repetitive Control System: A New Type Servo System for Periodic Exogenous Signals, *IEEE Trans. Automatic Control*, **33**-7, 659/668 (1988)
- 17) S. Hyon and T. Emura : Energy-preserving control of passive one-legged running robot, *Advanced Robotics*, **18**-4, 357/381 (2004)
- 18) D. Owaki, K. Osuka and A. Ishiguro : Gait transition between passive dynamic walking and running by changing the body elasticity, *Proc. SICE 2008 Annual Conf.*, 2513/2518 (2008)
- 19) M. Uemura, K. Kimura and S. Kawamura : Generation of energy saving motion for biped walking robot through resonance-based control method, *Proc. IEEE/RSJ Int.*

- Conf. Intelligent Robots and Systems*, 2928/2933 (2009)
- 20) S. Hyon and T. Emura : Symmetric walking control: Invariance and global stability, *Proc. IEEE ICRA*, 1455/1462 (2005)
- 21) 春名, 荻野, 細田, 浅田 : 上半身を持つ受動歩行ロボット, ロボティクスメカトロニクス講演会講演論文集, 2A1/E8 (2001)
- 22) P. Gregorio, M. Ahmadi and M. Buehler : Design, control, and energetics of an electrically actuated legged robot, *IEEE Trans. Systems, Man, and Cybernetics*, **27**-4, 626/634 (1997)

[著者紹介]

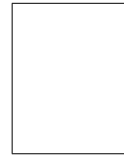
佐藤 訓志 (正会員)



2007年名古屋大学大学院工学研究科博士課程前期課程修了。2009-2010年日本学術振興会特別研究員(DC2)。2010年名古屋大学大学院工学研究科博士課程後期課程修了。同年より広島大学大学院工学研究院助教。2011年Radboud University Nijmegen 客員研究員。非線形制御, 確率制御の研究に従事。博士(工学)。2008年IEEE Robotics Automation Society Japan Chapter Young Award, 2009年計測自動制御学会制御部門研究奨励賞, 2010年計測自動制御学会学術奨励賞研究奨励賞などを受賞。システム制御情報学会, 日本ロボット学会, IEEEの会員。

[著者紹介]

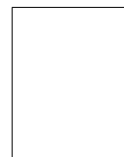
藤本 健治 (正会員)



1996年京都大学大学院工学研究科修士課程応用システム科学専攻修了, 1997年同大学院博士後期課程を中途退学, 同年京都大学大学院工学研究科助手, 2004年名古屋大学大学院工学研究科助教授等を経て, 2012年より京都大学大学院工学研究科教授。その間, 1999年オーストラリア国立大学客員研究員, 1999-2000年および2002年デルフト工科大学客員研究員, 2009年より理化学研究所客員研究員。非線形制御, 確率システムの研究に従事。博士(情報学)。IEEE, 計測自動制御学会, 日本機械学会, 日本鉄鋼協会の会員。

[著者紹介]

佐伯 正美 (正会員)



1981年, 京都大学大学院工学研究科博士課程単位取得退学。同年京都大学工学部助手, 82年筑波大学電子・情報工学系講師, 助教授を経て, 92年広島大学工学部第一類教授, 03年に大学院工学研究科教授となり, 現在に至る。ロボスタ制御系の設計の研究に従事(工学博士)。システム制御情報学会, 日本機械学会, 電気学会, IEEEの会員。