

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[†]

佐藤訓志*・藤本健治*・玄相昊**

Optimal gait generation for a one-legged robot based on variational symmetry of Hamiltonian systems

Satoshi SATOH*, Kenji FUJIMOTO* and Sang-Ho HYON**

This paper proposes a novel framework to generate an optimal gait trajectory with respect to the energy consumption for a planar one-legged robot via iterative learning control. The proposed method is based on the symmetric property of the variational systems of Hamiltonian systems called *variational symmetry*. This technique allows one to obtain solutions of a class of optimal control problems without using precise knowledge of the plant system. Furthermore, its application to the robot produces a passive gait trajectory with zero control input. Some numerical examples demonstrate the effectiveness of the proposed method.

Key Words: Nonlinear control, Hamiltonian systems, Iterative learning control, Gait generation, Passive dynamic walking

1. はじめに

近年の歩行ロボットに関する技術の進歩により、歩行制御の課題は、ただ歩かせるという段階から歩行の高速・高効率化、歩容の自然さの追求へ移行しつつある。簡単な構造のロボットが制御入力が必要とすることなく、ロボット自身のダイナミクスのみで緩やかな坂を歩き下るといった運動が McGeer¹⁾により報告された。受動歩行とよばれるこの歩行形態は、高効率かつより自然な歩行の手本になり得るものとして、挙動解析や制御応用など多岐にわたり多くの研究がなされてきた^{2),3)}。この受動歩行を規範とした動的歩容生成手法として文献4)~6)などがある。受動歩行は適当な斜面上でなければ実現できないが、Raibertらによって提案された線形バネで駆動する1脚ロボットモデルは、適切な初期状態のもとでは無入力で平地を走行できることが知られており^{7),8)}、受動歩行ロボットモデルとよばれている。

一方筆者らは、ハミルトン系の変分対称性を利用した反復学習制御法^{9)~11)}を提案している。この方法は制御目標の達成が期待される評価関数を定義し、勾配法に基づく更新則により各試行実験でこれを減少させる入力を設計する。その際に変分対称性という性質を利用することで、制御対象の詳細な情報を用いることなく、ある種の最適制御を学習により実

現できるという利点をもつ。そこで本論文の目的は、受動歩行ロボットに反復学習制御を適用することで、エネルギー効率の優れた最適な走行軌道を学習により生成することである。

本論文ではまず筆者らの過去の報告¹²⁾を基にし、入力 L_2 ノルムを最小とする準最適対称軌道を生成するための評価関数を提案する。しかしながら、文献9)~11)の反復学習制御が扱える評価関数は、入力と出力の汎関数のみという制限があり、そのままでは提案した評価関数は扱えない。そこで、二つの新しい手法を提案する。一つ目に、擬似共役微分作用素を定義し、これを用いた反復学習制御手法の拡張方法を提案する。本手法により、従来法がもつ評価関数の制限を緩和でき、準最適走行軌道が生成できるようになる。この軌道は入力 L_2 ノルムが最小ではあるが、制御入力を必要としない受動走行軌道になるとは限らない。これは、受動走行軌道はその初期状態に大きく依存しているが¹³⁾、一般的に反復学習制御では一定の初期状態の下での試行実験を仮定しているためである¹⁴⁾。そこで二つ目として、入力と同時に初期状態の更新も行うアルゴリズムを提案する。本論文の手法を用いることで、最適な初期状態も自動的に得ることができ、入力 L_2 ノルムが0となる受動走行軌道を学習により生成することが可能となる。最後にシミュレーションを用いて本論文で提案する手法の有効性を確認する。

2. 変分対称性を利用した反復学習制御

本節では、筆者らが提案している反復学習制御法^{9)~11)}を簡単に取り上げる。特にハミルトン系がもつ変分対称性、入力の更新則について述べる。

2.1 ハミルトン系と変分対称性

制御対象は、次式で表されるハミルトン系である。

[†] 第6回計測自動制御学会制御部門大会で一部発表(2006.5)

* 名古屋大学 大学院 工学研究科

** 科学技術振興機構 ICORP / ATR 脳情報研究所

* Department of Mechanical Science and Engineering, Graduate School of Engineering, Nagoya University, Frocho, Chikusa-ku, Nagoya-shi, Aichi, 464-8603, Japan

** ICORP, JST / ATR Computational Neuroscience Laboratory, 2-2-2, Hikaridai, Kyoto, 619-0288

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

ここで, $x(t) \in \mathbb{R}^n$, $u(t), y(t) \in \mathbb{R}^m$ はそれぞれシステムの状態, 入力, 出力を表す. また, $H(x, u, t) \in \mathbb{R}$ はハミルトン関数でありシステムのエネルギーを表している. 構造行列 $J \in \mathbb{R}^{n \times n}$ は歪対称行列で保存系の性質を表し, $R \in \mathbb{R}^{n \times n}$ は半正定対称行列でエネルギー散逸を表している. 有限時間区間 $[t^0, t^1]$ 上でのこの制御系の挙動を考え, この系を $\Sigma: \mathbb{R}^n \times L_2^m[t^0, t^1] \rightarrow \mathbb{R}^n \times L_2^m[t^0, t^1]$ の作用素ととらえる. 本論文では, 時間微分 d/dt と区別するために L_2 空間の Fréchet 微分は δ と表すことにする. ハミルトン系のもつ変分対称性として, 以下の定理が知られている⁹⁾.

《定理 1》⁹⁾ 式 (1) で表されるハミルトン系 Σ 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仮定する.

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

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

このとき制御対象の変分系は, 線形なハミルトン系として次式で表される.

$$(x_v^1, y_v) = \delta \Sigma((x^0, u), (x_v^0, u_v)) : \begin{cases} \dot{x} = (J - R) \frac{\partial H(x, u, t)}{\partial x}^\top, & x(t^0) = x^0 \\ \dot{x}_v = (J - R) \frac{\partial H_v(x, u, x_v, u_v, t)}{\partial x_v}^\top, & x_v(t^0) = x_v^0 \\ y_v = -\frac{\partial H_v(x, u, x_v, u_v, t)}{\partial u_v}^\top \\ x_v^1 = x_v(t^1) \end{cases} \quad (4)$$

ただし, ハミルトン関数 $H_v(x, u, x_v, u_v, t)$ は次式である.

$$H_v(x, u, x_v, u_v, t) = \frac{1}{2} \begin{pmatrix} x_v \\ u_v \end{pmatrix}^\top \frac{\partial^2 H(x, u, t)}{\partial(x, u)^2} \begin{pmatrix} x_v \\ u_v \end{pmatrix}$$

さらに $J - R$ が正則であると仮定すると, 変分随伴系は次式で表される.

$$(x_a^0, y_a) = (\delta \Sigma(x^0, u))^*(x_a^1, u_a) : \begin{cases} \dot{x} = (J - R) \frac{\partial H(x, u, t)}{\partial x}^\top, & x(t^0) = x^0 \\ \dot{\bar{x}}_v = -(J - R) \frac{\partial H_v(x, u, \bar{x}_v, u_a, t)}{\partial \bar{x}_v}^\top, \\ \bar{x}_v(t^1) = -(J - R)T x_a^1 \\ y_a = -\frac{\partial H_v(x, u, \bar{x}_v, u_a, t)}{\partial u_a}^\top \\ x_a^0 = -T^{-1}(J - R)^{-1} \bar{x}_v(t^0) \end{cases} \quad (5)$$

式 (5) より, 制御対象 Σ の変分系 $\delta \Sigma$ と, 変分随伴系 $(\delta \Sigma)^*$ は互いに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.

(注意) 変分対称性を用いると, 状態と入力 (x, u) について, ハミルトン関数の Hessian が*,

$$\frac{\partial^2 H(x, u, t)}{\partial(x, u)^2}(t - t^0) = \frac{\partial^2 H(x, u, t)}{\partial(x, u)^2}(t^1 - t), \quad \forall t \in [t^0, t^1] \quad (6)$$

を満たすとき, 式 (7) が成立し, 変分随伴系の入出力信号が制御対象の入出力信号から生成できる.

$$\begin{aligned} \pi_U \circ (\delta \Sigma(x^0, u))^*(\xi, v) \\ = \mathcal{R} \circ \pi_Y \circ (\delta \Sigma(x^0, u))(- (J - R)T\xi, \mathcal{R}(v)) \\ \approx \mathcal{R} \circ \pi_Y \circ \left(\Sigma(x^0 - (J - R)T\xi, u + \mathcal{R}(v)) - \Sigma(x^0, u) \right) \end{aligned} \quad (7)$$

ただし, \mathcal{R} は時間区間 $[t^0, t^1]$ 上での時間反転作用素を表し, 次式を満たす.

$$\mathcal{R}(u)(t - t^0) = u(t^1 - t), \quad \forall t \in [t^0, t^1] \quad (8)$$

さらに文献 10), 11) では, 式 (6) を満たさない場合についても議論されている.

2.2 勾配法を用いた入力の更新則

本節では, 勾配法を用いた入力の更新則について述べる. 式 (1) の制御対象 $\Sigma: X \times U \rightarrow X \times Y$ と, 評価関数 $\Gamma: X^2 \times U \times Y \rightarrow \mathbb{R}$ を考える. ただし X, U, Y はヒルベルト空間を表し, それぞれ $X = \mathbb{R}^n, U = Y = L_2^m[t^0, t^1]$ とする. 反復学習制御の目的は, 勾配法を用いてこの評価関数を最小化する入力を求めることである. $\Gamma(x^0, u, x^1, y)$ の Fréchet 微分は $(x^1, y) = \Sigma(x^0, u)$ に注意して次式のように計算できる.

$$\begin{aligned} \delta(\Gamma(x^0, u, x^1, y))(\delta x^0, \delta u, \delta x^1, \delta y) \\ = \delta \Gamma((x^0, u), \Sigma(x^0, u))((\delta x^0, \delta u), \delta \Sigma(x^0, u)(\delta x^0, \delta u)) \\ = \langle \nabla \Gamma((x^0, u), \Sigma(x^0, u)), \begin{pmatrix} \text{id}_{X \times U} \\ \delta \Sigma(x^0, u) \end{pmatrix} (\delta x^0, \delta u) \rangle_{X^2 \times U \times Y} \\ = \langle (\text{id}_{X \times U}, (\delta \Sigma(x^0, u))^*) \nabla \Gamma(x^0, u, x^1, y), (\delta x^0, \delta u) \rangle_{X \times U} \end{aligned} \quad (9)$$

ただし id は恒等写像を表す. また Riesz の表現定理より, 式 (9) を満たす勾配関数 $\nabla \Gamma(x^0, u, x^1, y)$ の存在が保証されている. したがって, 入力を次式のように更新すれば評価関数の値を最も減少させることができる.

$$u_{(i+1)} = u_{(i)} - K_{(i)}(0_{UX}, \text{id}_U)(\text{id}_{X \times U}, (\delta \Sigma(x_{(i)}^0, u_{(i)}))^*) \times \nabla \Gamma(x_{(i)}^0, u_{(i)}, x_{(i)}^1, y_{(i)}) \quad (10)$$

ここで添字 i は i 回目の試行実験を表し, $K_{(i)}$ は適当な正のゲインであるとする.

3. 擬似共役微分作用素を用いた拡張

式 (1) の制御対象は, 入力 u の定義から出力 y が一意に決

まり, 文献9)~11)の反復学習制御手法が適用できる評価関数は, 入力と出力の汎関数であるという制限があるため, 出力の時間微分 \dot{y} は扱うことができない. しかしながら, 例えば機械系では \dot{y} は運動エネルギーに直接関係する速度項を表し, 最適走行軌道の生成問題を考えるとき, その挙動は重要である. そこで本論文では, 擬似共役微分作用素を導入することで, 従来は扱うことができなかった出力の時間微分を含む評価関数に対しても, 変分対称性を用いた反復学習制御が適用できる方法を提案する.

3.1 擬似共役微分作用素の定義

本節ではまず微分作用素を定義し, その後本論文で導入する擬似共役微分作用素について述べる.

【定義】 ξ を $\xi \in L_2[t^0, t^1]$ であり, 少なくとも一階微分可能な信号であるとする. このとき, 信号 $\xi(t)$ とその時間微分を対応させる式 (11) の作用素 $D(\cdot)$ を微分作用素と定義する.

$$D(\xi)(t) = \frac{d\xi(t)}{dt} \quad (11)$$

一般的に微分作用素は非有界作用素であり, 共役作用素を定義することはできない. そこで本論文では, 扱う信号にある条件を付けることで, 共役作用素と類似の性質が利用できることを示す. そこで以下の補題を用意する.

[補題1] ξ, η をそれぞれ $\xi, \eta \in L_2[t^0, t^1]$ であり, 少なくとも一階微分可能な信号であるとする. また, 信号 ξ が次式を満たすと仮定する.

$$\xi(t^0) = \xi(t^1) = 0 \quad (12)$$

このとき, 次式が成立する.

$$\langle \eta, D(\xi) \rangle_{L_2} = -\langle D(\eta), \xi \rangle_{L_2} \quad (13)$$

(証明) η と $D(\xi)$ との内積を考える.

$$\begin{aligned} \langle \eta, D(\xi) \rangle_{L_2} &= \int_{t^0}^{t^1} \eta(t)^T \frac{d\xi(t)}{dt} dt \\ &= \left[\eta(t)^T \xi(t) \right]_{t^0}^{t^1} - \int_{t^0}^{t^1} \frac{d\eta(t)}{dt} \xi(t) dt \quad (14) \end{aligned}$$

ここで, 式 (12) より

$$\left[\eta(t)^T \xi(t) \right]_{t^0}^{t^1} = 0$$

であるため, 次式が成立する.

$$\begin{aligned} \langle \eta, D(\xi) \rangle_{L_2} &= - \int_{t^0}^{t^1} \frac{d\eta(t)}{dt} \xi(t) dt \\ &= -\langle D(\eta), \xi \rangle_{L_2} \quad (15) \end{aligned}$$

よって, 式 (13) の成立が示された. ■

補題1は, 式 (12) を満たす信号を D に作用させるとき, $D^* = -D$ が成立することを示しており, D^* を擬似共役微分作用素とよぶことにする.

3.2 反復学習制御への適用

本節では, 前節で定義した擬似共役微分作用素を用いて,

文献9)~11)の反復学習制御手法を拡張する方法を述べる.

ここでは, 次式のような評価関数を例にとって説明する.

$$\Gamma(\dot{y}) = \frac{1}{2} \int_{t^0}^{t^1} \left((\dot{y}(t) - \dot{y}^d(t))^T \Lambda_{\dot{y}} (\dot{y}(t) - \dot{y}^d(t)) \right) dt \quad (16)$$

ただし, \dot{y}^d は $\dot{y}^d \in L_2[t^0, t^1]$ であり, 少なくとも一階微分可能な目標速度軌道である. また, $\Lambda_{\dot{y}} \in \mathbb{R}^{m \times m}$ は適当な正定行列である. ここで, 以下の仮定をおく.

[仮定1] $y_{(i)}$ を i 回目の試行実験における出力とする. このとき, 以下の条件が満たされる.

$$|\delta y(t^0)| := |y_{(i)}(t^0) - y_{(i-1)}(t^0)| \ll 1 \quad (17)$$

$$|\delta y(t^1)| := |y_{(i)}(t^1) - y_{(i-1)}(t^1)| \ll 1 \quad (18)$$

一般的に反復学習制御では, 各試行実験において初期状態は常に一定であると仮定されているため, 式 (17) は常に成立する. 一方, 式 (18) は常に成立するわけではない. しかしながら, 軌道追従や終端時刻における目標出力が既知である最適制御の場合は, 例えば, 文献10), 11) で用いられている評価関数 (19) を元の評価関数 (16) に付加し, 重み K_y を十分大きくとり, 終端時刻の出力を速やかに目標出力 $y^d(t^1)$ に近づけることで, 近似的に満足することが期待できる.

$$\int_{t^1-\epsilon}^{t^1} K_y \|y(t) - y^d(t)\|^2 dt \quad (19)$$

ただし, K_y は正のゲイン, ϵ は微小な正定数である.

ここで, 出力 y が仮定1を満たすとする. 式 (16) の評価関数に対して, 次式のように計算できる.

$$\delta(\Gamma(\dot{y}))(\delta\dot{y}) = \langle \Lambda_{\dot{y}}(\dot{y} - \dot{y}^d), \delta\dot{y} \rangle_{L_2}$$

さて \dot{y} は出力ではないため, そのままでは文献9)~11)の方法を用いることはできない. そこで, 式 (11) の微分作用素 $D(\cdot)$ を用いて \dot{y} を $\dot{y} = D(y)$ と書き直す. すると次式が成立する.

$$\delta\dot{y} = \delta D(y)(\delta y) = D(\delta y) \quad (20)$$

仮定1と式 (13), (20) より, 次式のように計算できる.

$$\begin{aligned} \delta(\Gamma(\dot{y}))(\delta\dot{y}) &= \langle \Lambda_{\dot{y}}(\dot{y} - \dot{y}^d), D(\delta y) \rangle_{L_2} \\ &= -\langle (\delta\Sigma(u))^* (D(\Lambda_{\dot{y}}(\dot{y} - \dot{y}^d))), \delta u \rangle_{L_2} \quad (21) \end{aligned}$$

このように, 本論文で提案する手法により出力の時間微分 \dot{y} を含む評価関数に対しても, 変分対称性を利用した反復学習制御手法を適用することができるようになる.

4. 最適走行軌道の生成

本節ではまず, 本論文の制御対象である受動走行ロボットについて述べる. つぎに走行軌道を獲得するための制御問題をまとめ, これに基づき評価関数を提案する. 前節で提案した手法においてこの評価関数を用いることで, 準最適走行軌道が学習により生成できる. 生成された軌道は, 与えられた

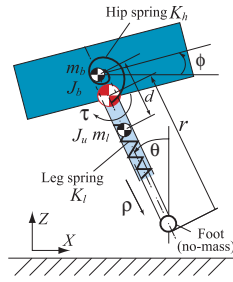


Fig. 1 Model of the passive running robot

Table 1 Parameters and variables

Notation	Meaning	Unit
m_b	body mass	kg
m_l	leg mass	kg
$m = m_b + m_l$	total mass	kg
J_b	body inertia	kgm ²
J_l	equivalent leg inertia	kgm ²
r_0	natural leg length	m
g	gravity acceleration	m/s ²
K_l	leg spring stiffness	N/m
K_h	hip spring stiffness	Nm/rad
ϕ	body angle	rad
θ	leg angle	rad
r	leg length	m
x	horizontal position of the center of mass	m
z	vertical position of the center of mass	m
τ	hip torque	Nm
ρ	leg force	N
T_s	stance time	s
T_f	flight time	s

初期状態の下で入力 L_2 ノルムが最小となるが、一般的に受動走行軌道とはならない。そこで本論文ではさらに、初期状態の更新則を導出し、入力と同時に初期状態の更新を行うアルゴリズムを提案する。本手法により入力が 0 となる受動走行軌道も学習により生成することが可能となる。

4.1 受動走行ロボットモデル

本論文では、Fig. 1 に示す受動走行ロボットモデル⁸⁾を考え、さらに地面との衝突も考慮する¹³⁾。このロボットの物理パラメータと変数を Table 1 に示す。走行運動は、脚が地面との接触を保ったまま運動を行う接地期 (Stance phase) と、脚が地面から離れて運動を行う浮遊期 (Flight phase) から構成される。1 周期における走行のアニメーションを Fig. 2 に示す。また、このモデルにはいくつかの仮定がおかれているが、ここでは重要なものだけを示す。その他については文献 13) を参照されたい。

[仮定 2] 衝突は完全非弾性衝突とする。

接地期におけるロボットのダイナミクスは、一般化座標を $q := (r, \theta, \phi)^T \in \mathbb{R} \times \mathbb{S} \times \mathbb{S}$ 、慣性行列を

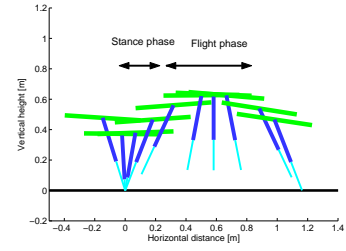


Fig. 2 Locomotion phases during one cycle

$$M(q) := \begin{pmatrix} m & 0 & 0 \\ 0 & J_l + mr^2 & 0 \\ 0 & 0 & J_b \end{pmatrix} \in \mathbb{R}^{3 \times 3}$$

、一般化運動量を $p := (p_r, p_\theta, p_\phi)^T = M(q)\dot{q} \in \mathbb{R}^3$ 、入力を $\bar{u} := (\rho, \tau)^T \in \mathbb{R}^2$ とすると、次式のハミルトン関数をもつ式 (1) のようなハミルトン系で記述できる。

$$H(q, p, \bar{u}) = \frac{1}{2} p^T M(q)^{-1} p - mgr(1 - \cos \theta) + \frac{1}{2} K_l (r - r_0)^2 + \frac{1}{2} K_h (\theta - \phi)^2 - \bar{u}^T \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & -1 \end{pmatrix} q$$

つぎに、浮遊期におけるダイナミクスを示す。

$$\begin{cases} \ddot{x} = 0 \\ \ddot{z} = -g \\ J_l \ddot{\theta} + J_b \ddot{\phi} = 0 \\ J_b \ddot{\phi} = K_h (\theta - \phi) - \tau_f \end{cases} \quad (22)$$

ただし、 τ_f は浮遊期において胴体に加えられるトルクを表す。

4.2 走行軌道獲得のための制御問題

周期的な走行の継続に直接関係しているのは脚の絶対角 θ であるが、本ロボットは足首トルクを利用できないため、接地期に θ を制御することは難しい。また、浮遊期のダイナミクス (22) より、浮遊期に利用できる制御入力はトルク τ_f のみである。そこで本論文でも、文献 15) で用いられている、浮遊期に τ_f を用いて転倒回避に直接関係する θ を制御し、接地期は無入力による自由運動にまかせるという制御戦略をとる。文献 15) によると、胴体姿勢 ϕ は走行の継続に直接関係していないため制御を施さず、 ϕ の挙動が自由運動では不満であれば、必要に応じて接地期に何らかの制御を別途施す。例えば、接地期においては ϕ に関する学習を行なうことも可能であるが、本論文では θ の制御についてのみ考える。そこで制御入力を $u := \tau_f$ とし、浮遊期の θ に関するダイナミクスを書き直す。文献 13) より、初期状態を適切に選ぶと式 (22) から θ を抜きだしたダイナミクスは、次式のような出力が θ であるハミルトン系で表すことができる。

$$\begin{pmatrix} \dot{q}_\theta \\ \dot{p}_\theta \end{pmatrix} = \begin{pmatrix} 0 & 1 \\ -1 & 0 \end{pmatrix} \begin{pmatrix} \frac{\partial H_\theta(q_\theta, p_\theta, u)^T}{\partial q_\theta} \\ \frac{\partial H_\theta(q_\theta, p_\theta, u)^T}{\partial p_\theta} \end{pmatrix}$$

$$y = -\frac{\partial H_\theta(q_\theta, p_\theta, u)^T}{\partial u} = q_\theta = \theta \quad (23)$$

ここで, $q_\theta := \theta, p_\theta := J_l \dot{\theta}$ であり, ハミルトン関数は次式である.

$$H_\theta(q_\theta, p_\theta, u) := \frac{1}{2J_l} p_\theta^2 + \frac{1}{2} \frac{K_h(J_b + J_l)}{J_b} q_\theta^2 - q_\theta u.$$

本論文では着地における衝突も考慮にしているため, 着地直前と直後にエネルギーの散逸が起り得る. このときのエネルギーをそれぞれ E_-, E_+ と書くことにすると, 文献 13) より散逸エネルギー ΔE は次式のように求まる.

$$\Delta E := E_- - E_+ = \frac{mJ_l}{2(J_l + mr_0^2)} \mu_-^2 \quad (24)$$

ただし μ_- は次式で定義され, 散逸係数と呼ばれている¹³⁾.

$$\mu_- := v_{x-} \cos \theta_- + v_{z-} \sin \theta_- + \frac{r_0}{J_l + mr_0^2} p_{\theta-}$$

ただし v_{x-}, v_{z-} は着地直前の重心の速度とする. したがって $\Delta E = 0$ が満たされれば走行中に制御エネルギー以外のエネルギーの授受はない.

文献 13), 15) より, 浮遊期における $\theta, \dot{\theta}$ が式 (25), (26) を満たす軌道は対称走行軌道とよばれ, $\Delta E = 0$ を満たし, かつ転倒しない走行の十分条件となっている.

$$\theta^d := \theta(t - t^0) = -\theta(t^1 - t) \quad (25)$$

$$\dot{\theta}^d := \dot{\theta}(t - t^0) = \dot{\theta}(t^1 - t), \quad \forall t \in [t^0, t^1] \quad (26)$$

文献 13) では浮遊期に有限整定制御を用いることで制御目標 (25), (26) を達成しているが, そのためには制御対象の詳細な情報が必要となる. そこで本論文では, 新しい評価関数を定義し, これを反復学習制御において用いることで, 制御対象の詳細な情報を用いることなく, 入力を最小とする対称走行軌道を学習により生成することを制御目標とする.

最後に注意すべきことは, 本来走行軌道は周期軌道であるが, 本手法は反復学習制御の枠組であるため, 直接周期軌道を生成することができないことである. しかしながら, 本手法で生成される軌道は $\Delta E = 0$ を満たす. そこで, 得られた浮遊期の軌道と接地期における軌道を最適周期軌道の一周期分に相当する軌道とみなし, これを繋ぎ合わせることで周期走行軌道が獲得できる.

4.3 対称走行軌道の生成

本節では, 前節で述べた制御目標を達成するために, 次式のような評価関数を提案する.

$$\Gamma(\theta, \dot{\theta}, u) := \frac{K_\theta}{2} \|\theta + \mathcal{R}(\theta)\|_{L_2}^2 + \frac{K_{\dot{\theta}}}{2} \|\dot{\theta} - \mathcal{R}(\dot{\theta})\|_{L_2}^2 + \frac{K_u}{2} \|u\|_{L_2}^2 \quad (27)$$

ただし, $K_\theta, K_{\dot{\theta}}, K_u$ はそれぞれ適当な正定数であり, $\mathcal{R}(\cdot)$ は式 (8) で定義した時間反転作用素である. この評価関数を反復学習制御により最小化することで, 式 (25), (26) の制御目標を満たし, かつ入力の L_2 ノルムを最小とする対称走行軌道が生成できると期待される. さらに式 (27) の評価関数は, 時間反転作用素を利用することで, 事前の目標軌道に関する情報などを必要としない.

式 (23) より出力が $y = \theta$ であることに注意すると, 評価関数 (27) は出力の時間微分 $\dot{y} = \dot{\theta}$ を含むため従来の反復学習制御法では扱うことができない. そこで, 本論文で提案した拡張法を用いて入力の更新則を導出する. はじめに, 以下の補題を用意する.

[補題 2] 式 (8) で定義される時間反転作用素 $\mathcal{R}(\cdot)$ を考える. このとき, 次式が成立する.

$$\mathcal{R}^* = \mathcal{R} \quad (28)$$

(証明) 信号 $\xi, \eta \in L_2[t^0, t^1]$ を考える. ξ と $\mathcal{R}(\eta)$ との内積を計算すると次式となる.

$$\begin{aligned} \langle \xi, \mathcal{R}(\eta) \rangle_{L_2} &= \int_{t^0}^{t^1} \xi(t)^T \eta(t^0 + t^1 - t) dt \\ &= \int_{t^0}^{t^1} \xi(t^0 + t^1 - s)^T \eta(s) ds \\ &= \langle \mathcal{R}(\xi), \eta \rangle_{L_2} \end{aligned} \quad (29)$$

ただし $s = t^0 + t^1 - t$ である. 式 (29) より, 式 (28) が示された. ■

式 (13), (28) より, 評価関数 (27) の Fréchet 微分は次式のように計算できる. ただし $\mathcal{R}(\cdot)$ の線形性から, $\delta \mathcal{R}(\cdot) = \mathcal{R}(\delta(\cdot))$ であることに注意する.

$$\begin{aligned} \delta(\Gamma(\theta, \dot{\theta}, u))(\delta\theta, \delta\dot{\theta}, \delta u) &= \langle K_\theta(\theta + \mathcal{R}(\theta)), \delta\theta + \mathcal{R}(\delta\theta) \rangle_{L_2} \\ &\quad + \langle K_{\dot{\theta}}(\dot{\theta} - \mathcal{R}(\dot{\theta})), \delta\dot{\theta} - \mathcal{R}(\delta\dot{\theta}) \rangle_{L_2} + \langle K_u u, \delta u \rangle_{L_2} \\ &= \langle 2K_\theta(\text{id} + \mathcal{R})(\theta) - 2K_{\dot{\theta}}(\text{id} - \mathcal{R})(\dot{\theta}), \delta\theta \rangle_{L_2} + \langle K_u u, \delta u \rangle_{L_2} \\ &= \langle K_u u + 2(\delta\Sigma(u))^* (K_\theta(\text{id} + \mathcal{R})(\theta) - K_{\dot{\theta}}(\text{id} - \mathcal{R})(\dot{\theta})), \delta u \rangle_{L_2} \end{aligned} \quad (30)$$

(注意) 変分対称性を利用するため, 受動走行ロボットが, 定理 1 の仮定を満たすことを確かめる. 式 (23) より

$$J = \begin{pmatrix} 0 & 1 \\ -1 & 0 \end{pmatrix}, \quad R = \begin{pmatrix} 0 & 0 \\ 0 & 0 \end{pmatrix}, \quad T = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix} \quad (31)$$

とすると, 式 (2) が満たされることがわかる. さらにハミルトン関数の Hessian は

$$\frac{\partial^2 H_\theta(q_\theta, p_\theta, u)}{\partial(q_\theta, p_\theta, u)^2} = \begin{pmatrix} \frac{K_h(J_b + J_l)}{J_b} & 0 & -1 \\ 0 & \frac{1}{J_l} & 0 \\ -1 & 0 & 0 \end{pmatrix}$$

であるから, これより式 (3) も満たされることが容易にわかる. よって式 (7), (10) にしたがって, 入力の更新則は次式となる.

$$u_{(2i+1)} = u_{(2i)} + 2\epsilon_{(i)} \mathcal{R} \left(K_\theta(\text{id} + \mathcal{R})(\theta_{(2i)}) - K_{\dot{\theta}}(\text{id} - \mathcal{R})(\dot{\theta}_{(2i)}) \right) \quad (32)$$

$$u_{(2i+2)} = u_{(2i)} - K_{(2i)} \left(K_u u_{(2i)} + \frac{1}{\epsilon_{(i)}} \mathcal{R}(\theta_{(2i+1)} - \theta_{(2i)}) \right) \quad (33)$$

ただし, $\epsilon_{(\cdot)}$ は適当な微小正定数, $K_{(\cdot)}$ は適当なゲインを表し, 初期入力 $u_{(0)} \equiv 0$ とする. また初期状態は適当に与えられ, 各試行実験で同じものとする. 式 (32), (33) は, 2 組の試行実験で 1 回の学習になっていることを示している. まず式 (32) は, 式 (30) の $(\delta\Sigma(u))^*$ の出力情報を, 式 (7) を用いて計算するため $\Sigma(0, u + \mathcal{R}(v))$ を生成する入力に相当する. $\Sigma(0, u + \mathcal{R}(v))$, $\Sigma(0, u)$ に相当する出力情報 $\theta_{(2i+1)}, \theta_{(2i)}$ から, 式 (7) を用いて $(\delta\Sigma(u))^*$ の出力を計算し, 式 (30) に代入したものが式 (33) に相当する. $\epsilon_{(\cdot)}$ は $\mathcal{R}(v)$ に相当する入力の変分を十分小さくするために用いる.

4.4 受動走行軌道の生成

すでに述べたように, 式 (32), (33) の入力の更新則を用いた試行実験の反復により, 準最適軌道の生成が期待できるが, それは必ずしも受動走行軌道となるわけではない. 本論文で対象とする受動走行ロボットモデル (Fig. 1) は, ある特別な初期状態の下では制御入力を必要とせずに走行できることが知られている^{7), 8)}. 従来の反復学習制御手法では, 一定の初期状態の下での試行実験を仮定しているが, 本論文では新たに初期状態に対する更新則を導出し, さらにこれと前節までの入力の更新と合わせることで, 入力が 0 となる受動走行軌道を学習により生成するアルゴリズムを提案する.

Table 1 より脚長初期状態は r^0 なので, 残りの初期状態 $(\dot{x}^0, \dot{z}^0, \theta^0, \dot{\theta}^0, \phi^0, \dot{\phi}^0)$ を定め, これを X^0 と書くことにする. 本論文では文献 13) の方法に従い \dot{x}^0, θ^0 をフリーパラメータとして, 残りの状態 $(\dot{z}^0, \dot{\theta}^0, \phi^0, \dot{\phi}^0)$ を \dot{x}^0, θ^0 から決定し, さらに θ^0 に対する更新則を導出する. \dot{x}^0 をフリーパラメータとして残すことで, 得られる最適軌道の並進速度を指定できる.

以下では θ^0 の更新則を導出する. ここでは式 (9) においてさらに $\nabla\Gamma(x^0, u, x^1, y) =: (\nabla\Gamma_{x^0}, \nabla\Gamma_u, \nabla\Gamma_{x^1}, \nabla\Gamma_y)^T$ とすることで, 次式のように計算できる.

$$\begin{aligned} & \delta(\Gamma((x^0, u), \Sigma(x^0, u))) (\delta x^0, \delta u) \\ &= \langle \nabla\Gamma_{x^0} + \pi_{\mathbb{R}^n} \circ (\delta\Sigma(x^0, u))^* (\nabla\Gamma_{x^1}, \nabla\Gamma_y), \delta x^0 \rangle \\ &+ \langle \nabla\Gamma_u + \pi_U \circ (\delta\Sigma(x^0, u))^* (\nabla\Gamma_{x^1}, \nabla\Gamma_y), \delta u \rangle \end{aligned}$$

ここで, $\pi_{(\cdot)}$ は (\cdot) への投影写像を表す. 勾配法により適当なゲイン K_{x^0} を用いて次式のように初期状態を更新する.

$$\delta x^0 = -K_{x^0} \left(\nabla\Gamma_{x^0} + \pi_{\mathbb{R}^n} \circ (\delta\Sigma(x^0, u))^* (\nabla\Gamma_{x^1}, \nabla\Gamma_y) \right) \quad (34)$$

以下では $t = t^0$ における状態を初期状態, $t = t^1$ における状態を終端状態と書くことにする. 式 (34) において, $(\nabla\Gamma_{x^1}, \nabla\Gamma_y)$ を入力としたときの変分随伴系 $(\delta\Sigma(x^0, u))^*$ の初期状態, つまり $x_a^0 := \pi_{\mathbb{R}^n} \circ (\delta\Sigma(x^0, u))^* (\nabla\Gamma_{x^1}, \nabla\Gamma_y)$ は, 制御対象の詳細な情報なしでは計算できない. そこで変分対称性を利用する. 式 (5) より, 変分随伴系の初期状態は $x_a^0 = -T^{-1}(J - R)^{-1}\dot{x}_v(t^0)$ である. 変分対称性より, 式 (5) の状態空間表現が, 式 (4) を時間反転したものであることを利用すると, $x_a^0 = -T^{-1}(J - R)^{-1}\dot{x}_v(t^1)$ と書ける. た

だし, $\dot{x}_v(t^1)$ は式 (4) の変分系の終端状態である. よって式 (34) は次式のように求まる.

$$\delta x^0 = -K_{x^0} (\nabla\Gamma_{x^0} - T^{-1}(J - R)^{-1}\dot{x}_v(t^1)) \quad (35)$$

式 (31) より, $x_a^0 =: (q_{\theta_a^0}, p_{\theta_a^0})^T$ は以下のように計算できる.

$$\begin{aligned} x_a^0 &= - \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}^{-1} \left(\begin{pmatrix} 0 & 1 \\ -1 & 0 \end{pmatrix} - \begin{pmatrix} 0 & 0 \\ 0 & 0 \end{pmatrix} \right)^{-1} \begin{pmatrix} q_{\theta_v(t^1)} \\ p_{\theta_v(t^1)} \end{pmatrix} \\ &= \begin{pmatrix} p_{\theta_v(t^1)} \\ q_{\theta_v(t^1)} \end{pmatrix}, \end{aligned} \quad (36)$$

ただし, $(q_{\theta_v(t^1)}, p_{\theta_v(t^1)})^T =: x_v(t^1)$ である. 式 (35), (36) より, 初期状態 θ^0 を次式のように更新すればよい.

$$\delta\theta^0 = -K_{\theta^0} p_{\theta_v(t^1)} \quad (37)$$

初期状態の更新則 (37) が導出できたので, 最後に本節で提案する入力と初期状態の更新を組み合わせたアルゴリズムをまとめる. ただし, 設計パラメータである $\epsilon_{(\cdot)}, K_{(\cdot)}, K_{\theta^0(\cdot)}$ は仮定 1 を満たすように適当に設定する. 以下では, $X^0_{(j)} \in \mathbb{R}^6$ を j 回目の学習に対する初期状態を表すものとする (1 回の学習に, 同一初期状態の下での 2 組の試行実験が必要であることを注意する).

Step 0: フリーパラメータである, 並進速度 \dot{x}^0 と脚の絶対対角の初期状態 $\theta^0_{(0)}$ を定め, 文献 13) に従い残りの初期状態を求め, これを $X^0_{(0)}$ とする. $j=0$ として Step 1 へ.

Step $2j+1$: 初期状態 $X^0_{(j)}$ の下で, $2j+1$ 回目の入力 (32) を用いて試行実験を行う.

Step $2j+2$: 初期状態 $X^0_{(j)}$ の下で, $2j+2$ 回目の入力 (33) を用いて試行実験を行い, j 回目の学習を完了する. ここでもし, $\|u_{(2j)}\|_{L_2} \leq \gamma$ ならば試行を終了する. ただし, γ は微小な正定数である. そうでなければ, 初期状態の更新則 (37) を用いて $\theta^0_{(j+1)}$ を求め, これと \dot{x}^0 から $j+1$ 回目の学習に対する初期状態 $X^0_{(j+1)}$ を得る.

$j=j+1$ として Step $2j+3$ へ.

5. シミュレーション結果

本節では本論文で提案したアルゴリズムを, Fig. 1 の受動走行ロボットモデルに適用した数値例を示す. ロボットの物理パラメータは文献 13) と同じものとし, $m_b = 10$ [kg], $m_l = 2$ [kg], $J_b = 0.50$ [kgm²], $J_l = 0.11$ [kgm²], $K_h = 10$ [Nm/rad], $K_l = 3000$ [N/m], $r^0 = 0.5$ [m] である.

5.1 対称走行軌道

本節では, 4.3 節で導出した入力の更新則 (32), (33) を用いて, 与えられた初期状態の下で入力の L_2 ノルムが最小となる準最適な対称走行軌道を学習により生成する. 初期状態は

$$\begin{aligned} & (\dot{x}^0, \dot{z}^0, \theta^0, \dot{\theta}^0, \phi^0, \dot{\phi}^0) \\ &= (2, -1.9516, 0.30, -2.6679, -0.0660, 0.5872) \end{aligned}$$

, ゲインパラメータは $K_\theta = 1$, $K_{\dot{\theta}} = 1 \times 10^{-4}$, $K_u =$

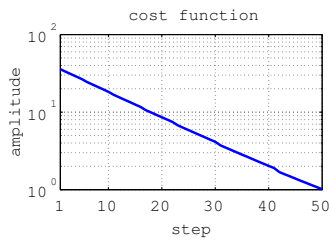


Fig. 3 Cost function

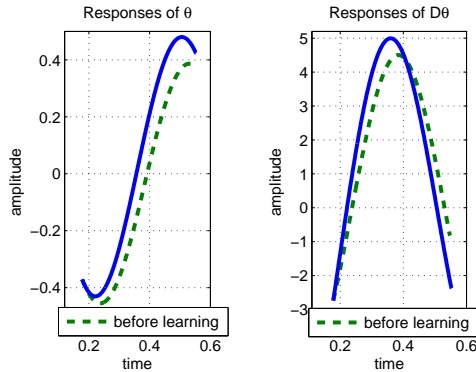
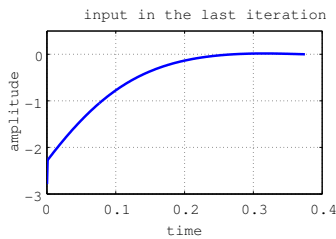
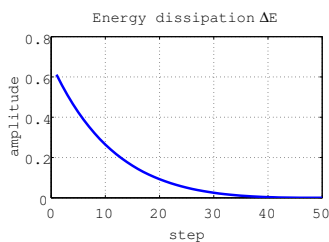
Fig. 4 Responses of θ and $\dot{\theta}$ 

Fig. 5 Control input in the last iteration

Fig. 6 Dissipation of the energy at the touch down ΔE

1×10^{-3} , $K_{(\cdot)} = 0.6$ とし, $\epsilon_{(\cdot)} = 0.01$ の下で, 50 回の学習つまり 100 回のシミュレーションを行った結果を Fig. 3 ~ 6 に示す. Fig. 3 より, 評価関数が各試行実験において単調に減少しており, これは速やかに最適軌道が生成されていることを示している. Fig. 4 に θ と $\dot{\theta}$ の学習前の自由運動における出力と, 最終的に生成された出力を示す. θ , $\dot{\theta}$ ともに制御目標 (25), (26) を満たし, Fig. 6 より式 (24) のエネルギー散逸 ΔE が 0 に収束していることがわかる. しかしながら, Fig. 5 より, 生成された入力ノルムが 0 ではないため, 得られた軌道は準最適ではあるが受動走行軌道ではない.

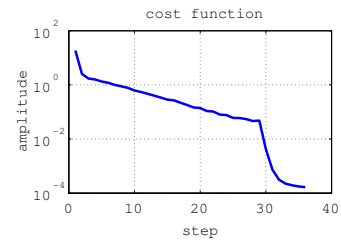
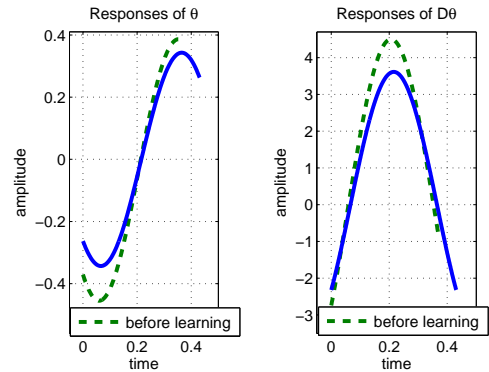
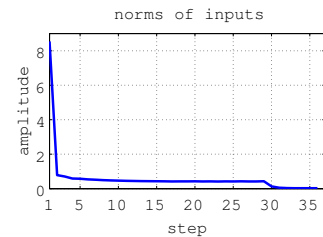
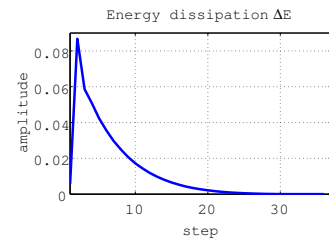


Fig. 7 Cost function

Fig. 8 Responses of θ and $\dot{\theta}$ Fig. 9 L_2 norm of the control inputsFig. 10 Dissipation of the energy at the touch down ΔE

5.2 受動走行軌道

本節では 4.4 節で提案したアルゴリズムを適用することで, 受動走行軌道を生成する. 入力ノルムの収束を判定する微小正定数 γ は $\gamma = 0.025$, 学習開始時の初期状態は前節と同じものを用いてシミュレーションを行った結果, 72 回の試行実験で入力ノルムが γ 以下に収束した. ただし, ゲインパラメータは $K_\theta = 1$, $K_{\dot{\theta}} = 1 \times 10^{-5}$, $K_u = 0.5$, $K_{(\cdot)} = 2$ とし, $\epsilon_{(\cdot)} = 0.2$ である. Fig. 7 より, 評価関数が単調減少していることから, 速やかに最適軌道に収束していることがわかる. Fig. 8 に θ と $\dot{\theta}$ の学習前の出力と, 最終的に生成された出力を示す. θ , $\dot{\theta}$ ともに制御目標 (25), (26) を満たす

ような軌道となっている。Fig. 4 と比較すると、初期状態も更新されていることがわかる。さらに、Fig. 10 より、エネルギー散逸 ΔE が、Fig. 9 より入力 L_2 ノルムがともに 0 へ収束していくこともわかり、本手法により最適軌道として対称受動走行軌道が生成されていることがわかる。

6. おわりに

本論文では、変分対称性を利用した反復学習制御に基づく、1 脚ロボットの最適な走行軌道の生成手法を提案した。本手法は制御対象の詳細な情報や目標軌道に関する事前の情報を必要とせず、試行実験の繰り返しにより、入力 L_2 ノルムを最小とする最適対称軌道を生成できるという利点をもつ。さらに、1 脚走行ロボットによるシミュレーションを行い、本手法の有効性を確認した。今後は、より一般的なロボットへの適用が課題である。現在は、離散的な状態遷移を考慮した最適周期軌道を学習により生成する枠組を考案し、簡単な 2 足ロボットでその効果を確認している¹⁶⁾。また、実機検証において効率的に試行実験を繰り返す方法も今後の課題である。

参考文献

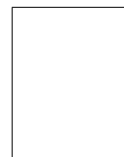
- 1) T. McGeer : Passive Dynamic Walking, Int. J. Robotics Research, **9**-2, 62/82 (1990)
- 2) K. Osuka and K. Kirihara : Motion analysis and experiments of passive walking robot QUARTET II, Proc. the 2000 IEEE Int. Conf. on Robotics and Automation, 3052/3056 (2000)
- 3) A. Sano, Y. Ikemata and H. Fujimoto : Analysis of Dynamics of Passive Walking from Storage Energy and Supply Rate, Proc. the 2003 IEEE Int. Conf. on Robotics and Automation, 2478/2483 (2003)
- 4) A. Goswami, B. Espiau and A. Keramane : Limit Cycles in a Passive Compass Gait Biped and Passivity-Mimicking Control Laws, Autonomous Robots, **4**-3, 273/286 (1997)
- 5) M. W. Spong : Passivity-base control of the compass gait biped, Proc. of IFAC World Congress, 19/23 (1999)
- 6) 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)
- 7) C. Thompson and M. Raibert : Passive dynamic running, Experimental Robotics I (Eds. by V. Hayward and O. khatib), Vol. 139 of Lecture Notes in Control and Information Science, Springer-Verlag, Berlin, 74/83 (1989)
- 8) M. Ahmadi and M. Buehler : Stable control of a simulated one-legged running robot with hip and leg compliance, IEEE Trans. Robotics and Automation, **13**-1, 96/104 (1997)
- 9) 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)
- 10) K. Fujimoto, T. Horiuchi and T. Sugie : Optimal control of Hamiltonian systems with input constraints via iterative learning, Proc. 42nd IEEE Conf. on Decision and Control, 4387/4392 (2003)
- 11) 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誌, 掲載予定 (2008)
- 12) 佐藤, 藤本, 玄 : ハミルトン系の繰り返し制御に関する一考察, 第 49 回システム制御情報学会 研究発表講演会予稿集, 431/432

(2005)

- 13) S. Hyon and T. Emura : Energy-preserving control of passive one-legged running robot, Advanced Robotics, **18**-4, 357/381 (2004)
- 14) S. Arimoto, S. Kawamura and F. Miyazaki : Bettering operation of robotics, Journal of Robotic Systems, **1**-2, 123/140 (1984)
- 15) 玄 相昊 : 動的脚口ポットのハミルトン系に基づく走行制御, システム制御情報学会誌, **49**-7, 260/265 (2005)
- 16) S. Satoh, K. Fujimoto and S. Hyon : Biped gait generation via iterative learning control including discrete state transitions, Submitted (2007)

[著者紹介]

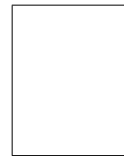
佐藤 訓志 (学生会員)



2005 年名古屋大学工学部機械・航空工学科卒業。2007 年名古屋大学大学院工学研究科博士前期課程修了。同年同博士後期課程に進学し、現在に至る。非線形制御の研究に従事。

[著者紹介]

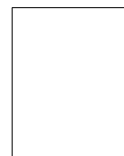
藤本 健治 (正会員)



1994 年京都大学工学部精密工学科卒業, 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, 1997 年京都大学大学院工学研究科助手等を経て, 2007 年より名古屋大学大学院工学研究科准教授。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。非線形制御の研究に従事。博士 (情報学)。2000 年計測自動制御学会論文賞武田賞, 2003 年計測自動制御学会制御部門大会賞, 2005 年 The IFAC Congress Young Author Prize, 2007 年計測自動制御学会制御部門パイオニア賞を受賞。IEEE, 計測自動制御学会, システム制御情報学会, 日本機械学会, 日本鉄鋼協会の会員。

[著者紹介]

玄 相昊 (正会員)



1998 年 3 月早稲田大学大学院修士課程修了 (機械工学)。2002 年 3 月東京工業大学大学院博士後期課程修了 (制御工学)。同年 4 月東北大学大学院工学研究科助手, 2005 年 1 月講師となる。2005 年 4 月から ATR 脳情報研究所研究員兼 JST 国際共同研究事業 ICORP 計算脳プロジェクト研究員となり、現在に至る。非線形力学系に基づくヒューマノイドロボットの運動制御の研究に従事する。博士 (工学)。日本ロボット学会, IEEE などの会員。