Proceedings of the 17th World Congress
The International Federation of Automatic Control
Seoul, Korea, July 6-11, 2008

Biped gait generation via iterative learning control including discrete state transitions

Satoshi Satoh * **Kenji Fujimoto** ** **Sang-Ho Hyon** ***

* *Department of Mechanical Science and Engineering, Graduate School of Engineering, Nagoya University, Furo-cho, Chikusa-ku, Nagoya, 464-8603, Japan (e-mail: s.satoh@nagoya-u.jp)*
** *Department of Mechanical Science and Engineering, Graduate School of Engineering, Nagoya University, Furo-cho, Chikusa-ku, Nagoya, 464-8603, Japan (e-mail: k.fujimoto@ieee.org)*
*** *JST, ICORP, Computational Brain Project, Saitama 332-0012, Japan. and ATR Computational Neuroscience Laboratories, Kyoto 619-0288, Japan (e-mail: sangho@atr.jp)*

Abstract: This paper is concerned with a gait generation for legged robots via iterative learning control (ILC) including discrete state transitions. This method allows one to obtain solutions of a class of optimal control problems without using precise knowledge of the plant model by iteration of laboratory experiments. Generally in walking motion, there are discrete state transitions caused by landing. The proposed framework can also deal with such state transitions without using the parameters of the transition model by combining ILC method and the least-squares. It is applied to the compass gait biped to generate optimal gait on the level ground. Furthermore, some numerical examples demonstrate the effectiveness of the proposed method.

Keywords: Intelligent robotics; Mobile robots; Iterative learning control; Optimal control theory; Nonlinear systems.

1. INTRODUCTION

Recently, the walking control has become an active research area. As the technology for walking robots evolves, an optimization problem of gaits with respect to the energy consumption becomes important increasingly. In this point of view, *passive dynamic walking* studied by McGeer [1990] attracts attention. The gait is stable and periodic on a gentle slope and it is generated with no actuation of any kind, i.e., powered only by gravity. Behavior analysis of passive walkers were studied by several researchers, Osuka and Kirihara [2000], Sano et al. [2003]. In Goswami et al. [1997], Spong [1999], Asano et al. [2004], gait generation methods based on passive dynamic walking, i.e., designing appropriate feedback control systems so that the closed loop systems behave like passive walkers are also proposed.

In the meanwhile, we proposed an iterative learning control method based on a property of Hamiltonian systems called *variational symmetry*. It allows one to solve a class of optimal control problems by iteration of laboratory experiments. Taking advantage of variational symmetry, this method does not require the precise knowledge of the plant model. We also studied on optimal gait generation with respect to the energy consumption for legged robots via this technique in Satoh et al. [2006c,a]. Although there are discrete state transitions in general walking motion caused by a collision between a leg and the ground, our former technique could not deal with them directly. Instead, we restricted the desired walking trajectories to symmetric ones to avoid this problem in Satoh et al. [2006c,a].

In this paper, we propose a gait generation framework including discrete state transitions. Transition equations derived by the conservation law of angular momentum are often used in walking analysis. However, this law does not hold exactly with real robots because the law depends on a theoretical assumptions. For example, It is assumed that there exists no double support phase at the touchdown. The proposed method generates a feedforward input for an optimal gait and estimates the transition mapping via the least-squares by iteration of experiments. Our framework does not require the information of the robot parameters nor the transition model. Applying this technique to a simple planar biped robot, we generate optimal gait trajectories. Numerical simulations demonstrate the validity of the proposed framework.

2. ITERATIVE LEARNING CONTROL OF HAMILTONIAN SYSTEMS BASED ON VARIATIONAL SYMMETRY

This section refers to iterative learning control (ILC) based on variational symmetry in Fujimoto and Sugie [2003].

2.1 Variational symmetry of Hamiltonian systems

Consider a Hamiltonian system Σ with a controlled Hamiltonian $H(x, u, t)$ described as $(x^1, y) = \Sigma(x^0, u)$:

$$\begin{cases} \dot{x} \;=\; (J-R)\,\dfrac{\partial H(x,u,t)}{\partial x}^{\mathrm{T}},\; x(t^0)=x^0 \\ y \;=\; -\dfrac{\partial H(x,u,t)}{\partial u}^{\mathrm{T}} \\ x^1 \;=\; x(t^1) \end{cases} \quad . \tag{1}$$

Here $x(t) \in \mathbb{R}^n$ and $u(t), y(t) \in \mathbb{R}^m$ describe the state, the input and the output, respectively. The structure matrix $J \in \mathbb{R}^{n\times n}$ and the dissipation matrix $R \in \mathbb{R}^{n\times n}$ are skew-symmetric and symmetric positive semi-definite, respectively. In this paper, we describe the time derivative and the Fréchet derivative on L_2 space as $\mathrm{d}/\mathrm{d}t$ and δ, respectively. For the system (1), the following theorem holds. This property is called *variational symmetry* of Hamiltonian systems.

Theorem 1. Fujimoto and Sugie [2003] Consider the Hamiltonian system (1). *Suppose that J and R are constant and that there exists a nonsingular matrix $T \in \mathbb{R}^{n\times n}$ satisfying*

$$J = -TJ\,T^{-1}, \qquad R = TR\,T^{-1} \tag{2}$$

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

Then the Fréchet derivative of Σ is described by another linear Hamiltonian system $(x_v^1, y_v) = \delta\Sigma(x^0, u)(x_v^0, u_v)$:

$$\begin{cases} \dot{x} \;=\; (J-R)\dfrac{\partial H(x,u,t)}{\partial x}^{\mathrm{T}}, \qquad x(t^0)=x^0 \\ \dot{x}_v \;=\; (J-R)\dfrac{\partial H_v(x,u,x_v,u_v,t)}{\partial x_v}^{\mathrm{T}}, x_v(t^0)=x_v^0 \\ y_v \;=\; -\dfrac{\partial H_v(x,u,x_v,u_v,t)}{\partial u_v}^{\mathrm{T}} \\ x_v^1 \;=\; x_v(t^1) \end{cases} \tag{3}$$

with a controlled Hamiltonian $H_v(x,u,x_v,u_v,t)$

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

Furthermore, suppose that $J-R$ is nonsingular. Then the adjoint of the variational system $(x_a^1, u_a) \mapsto (x_a^0, y_a) = (\delta\Sigma(x^0, u))^*(x_a^1, u_a)$ is given by the same state-space realization (4) *with the initial state $x(t^0) = x^0$ and the terminal state $\bar{x}_v(t^1) = -(J-R)T\,x_a^1$ and $x_a^0 = -T^{-1}(J-R)^{-1}\bar{x}_v(t^0)$ as*

$$\begin{cases} \dot{x} \;=\; (J-R)\dfrac{\partial H(x,u,t)}{\partial x}^{\mathrm{T}}, \quad x(t^0)=x^0 \\ \dot{\bar{x}}_v \;=\; -(J-R)\dfrac{\partial H_v(x,u,\bar{x}_v,u_a,t)}{\partial \bar{x}_v}^{\mathrm{T}}, \\ \qquad \bar{x}_v(t^1) = -(J-R)T\,x_a^1 \\ y_a \;=\; -\dfrac{\partial H_v(x,u,\bar{x}_v,u_a,t)}{\partial u_a}^{\mathrm{T}} \\ x_a^0 \;=\; -T^{-1}(J-R)^{-1}\bar{x}_v(t^0) \end{cases} \quad . \tag{4}$$

Remark 1. This theorem reveals that the variational system and its adjoint of a Hamiltonian system (1) have almost the same state-space realizations. This means that the input-output mapping of the adjoint can be calculated by the input-output data of the original system as

$$\begin{aligned} &\pi_U \circ (\delta\Sigma(x^0, u))^*(\xi, v) \\ &= \mathcal{R} \circ \pi_Y \circ (\delta\Sigma(x^0, u))\big(-(J-R)T\xi, \mathcal{R}(v)\big) \\ &\approx \mathcal{R} \circ \pi_Y \circ \Big(\Sigma\big(x^0 - (J-R)T\xi, u + \mathcal{R}(v)\big) - \Sigma(x^0, u)\Big) \end{aligned} \tag{5}$$

provided appropriate boundary conditions are selected, where $\pi_{(\cdot)}$ denotes the projection operator onto $(\cdot)$ and $\mathcal{R}$ is the time reversal operator defined by

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

2.2 Optimal control via iterative learning

Let us consider the system $\Sigma : X \times U \to X \times Y$ in (1) and a cost function $\hat{\Gamma} : X^2 \times U \times Y \to \mathbb{R}$ with Hilbert spaces X, U and Y. Typically, $X = \mathbb{R}^n$ and $U, Y = L_2^m[t^0, t^1]$. The objective is to find an optimal input minimizing the cost function $\hat{\Gamma}(x^0, u, x^1, y) = \Gamma(x^0, u)$ by $(x^1, y) = \Sigma(x^0, u)$. Here we can calculate

$$\begin{aligned} &\delta\Gamma(x^0, u)(\delta x^0, \delta u) \\ &= \delta\hat{\Gamma}((x^0, u), \Sigma(x^0, u))\,\big((\delta x^0, \delta u), \delta\Sigma(x^0, u)(\delta x^0, \delta u)\big) \\ &= \langle \Gamma'(x^0, u, x^1, y), \begin{pmatrix} \mathrm{id} \\ \delta\Sigma(x^0, u) \end{pmatrix} (\delta x^0, \delta u) \rangle_{X^2 \times U \times Y} \\ &= \langle \big(\mathrm{id}, (\delta\Sigma(x^0, u))^*\big)\,\Gamma'((x^0, u), \Sigma(x^0, u)), (\delta x^0, \delta u) \rangle_{X \times U}, \end{aligned} \tag{7}$$

where id represents the identity mapping. Well-known Riesz's representation theorem and the linearity of Fréchet derivative guarantee that an operator $\Gamma'(x^0, u, x^1, y)$ satisfying Eq. (7) exists. Therefore, if the adjoint $(\delta\Sigma(x^0, u))^*$ is available, we can reduce the cost function down at least to a local minimum by the iteration law with $K_{(i)} > 0$ as

$$\begin{aligned} u_{(i+1)} = u_{(i)} - K_{(i)}\pi_X \circ \Big(\mathrm{id}, (\delta\Sigma(x_{(i)}^0, u_{(i)}))^*\Big) \\ \times \Gamma'(x_{(i)}^0, u_{(i)}, x_{(i)}^1, y_{(i)}). \end{aligned} \tag{8}$$

Here i denotes the i-th iteration in laboratory experiment.

The results in Subsection 2.1 enable one to execute this procedure without using the parameters of the original system Σ by Eq. (5), provided Σ is a Hamiltonian system and the boundary conditions are selected appropriately.

3. OPTIMAL GAIT GENERATION FOR THE COMPASS GAIT BIPED

In this section, we propose a framework to generate an optimal gait trajectory via iterative learning control method mentioned in Section 2. First, we consider a simple biped robot. Then, let us define a cost function, by which one can generate optimal periodic trajectories minimizing the L_2 norm of the control input. Such periodic trajectories satisfy one of the necessary conditions for periodic gaits. Following that, the iteration law with respect to the cost function is derived.

Table 1. Parameters and variables

Notation	Meaning	Unit
m_H	hip mass	kg
m	leg mass	kg
a	length from m to ground	m
b	length from hip to m	m
$l = a + b$	total leg length	m
g	gravity acceleration	m/s^2
θ_1	stance leg angle w.r.t vertical	rad
θ_2	swing leg angle w.r.t vertical	rad
u_1	ankle torque	Nm
u_2	hip torque	Nm
x_g	horizontal position of C.o.M	m
y_g	vertical position of C.o.M	m

3.1 Description of the plant

Let us consider a full-actuated planar compass-like biped robot called the *compass gait biped* Goswami et al. [1996] depicted in Fig. 1. The robot can walk down a gentle slope

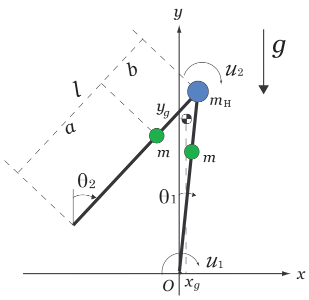

Fig. 1. Model of the compass gait biped

without any control inputs under appropriate initial conditions McGeer [1990]. Table 1 shows physical parameters and variables. Furthermore, We assume some assumptions on this robot. Some important ones are as follows. The rest of them conforms Goswami et al. [1996].

Assumption 1. The foot does not bounce back nor slip on the ground (inelastic impulsive impact).

Assumption 2. Transfer of support between the stance and the swing legs is instantaneous.

Table 2. Some notations

Notation	Meaning
$q := (\theta_1, \theta_2)^{\mathrm{T}}$	generalized coordinate
$p := (p_1, p_2)^{\mathrm{T}}$	generalized momentum
$x := (q^{\mathrm{T}}, p^{\mathrm{T}})^{\mathrm{T}}$	state
$\theta := (\theta_1, \theta_2)^{\mathrm{T}}$	angles of legs
$\dot\theta := (\dot\theta_1, \dot\theta_2)^{\mathrm{T}}$	angular velocities of legs
$\Theta := (\theta^{\mathrm{T}}, \dot\theta^{\mathrm{T}})^{\mathrm{T}}$	angles and their velocities
$x^0 := ({q^0}^{\mathrm{T}}, {p^0}^{\mathrm{T}})^{\mathrm{T}}$	initial state
$x^1 := ({q^1}^{\mathrm{T}}, {p^1}^{\mathrm{T}})^{\mathrm{T}}$	terminal state
$.^{-(+)}$	just before (after) transfer Note that $\theta \equiv q$ and $x^- \equiv x^1$.

We use number of notations with respect to the state. Table 2 summaries these ones. Here is a new input defined as $\bar u := (\bar u_1, \bar u_2)^{\mathrm{T}} = (u_1 + u_2, -u_2)^{\mathrm{T}}$ Then, the dynamics of this robot is described by a Hamiltonian system in (1) with the Hamiltonian

$$H(\theta, \dot\theta, \bar u) = \frac{1}{2}\dot\theta^{\mathrm{T}} M(\theta)\dot\theta + V(\theta) - \bar u^{\mathrm{T}}\theta$$

where a positive definite matrix $M(\theta) \in \mathbb{R}^{2\times2}$ denotes the inertia matrix and a scalar function $V(\theta) \in \mathbb{R}$ denotes the potential energy of the system. The details are as follows

$$M(\theta) = \begin{pmatrix} m_H l^2 + ma^2 + ml^2 & -mbl\cos(\theta_1 - \theta_2) \\ -mbl\cos(\theta_1 - \theta_2) & mb^2 \end{pmatrix}$$

$$V(\theta) = \{(m_H l + ma + ml)\cos\theta_1 - mb\cos\theta_2\}g. \qquad (9)$$

Note that using (9), the generalized momentum is described as $p = M(\theta)\dot\theta$.

Assumption 1 and Assumption 2 imply that there exists no double support phase. Following the law of conservation of the angular momentum, we can obtain a transition equation as

$$Q^+(\theta^+)\dot\theta^+ = Q^-(\theta^-)\dot\theta^-, \qquad (10)$$

where $\dot\theta^-$ and $\dot\theta^+$ denote the angular velocities just before and just after the state transition caused by a collision between a leg and the ground, respectively. θ^+ can be obtained by

$$\theta^+ = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}\theta^- =: C\theta^-,$$

where the matrix C exchanges the support and the swing leg angles. The matrices in (10) are as follows

$$Q^-(\theta^-) = \begin{pmatrix} (m_H l^2 + 2mal)\cos(\theta_1^- - \theta_2^-) - mab & -mab \\ -mab & 0 \end{pmatrix}$$

$$Q^+(\theta^+) = \qquad (11)$$
$$\begin{pmatrix} m_H l^2 + ma^2 + ml(l - b\cos(\theta_1^+ - \theta_2^+)) & mb(b - l\cos(\theta_1^+ - \theta_2^+)) \\ -mbl\cos(\theta_1^+ - \theta_2^+) & mb^2 \end{pmatrix}$$

Here we rewrite the transition equation in (10) as

$$\dot\theta^+ = \mathbf{Q}(\theta^-)\dot\theta^-, \qquad (12)$$

where $\mathbf{Q}(\theta^-) := {Q^+}^{-1}(C\theta^-)\, Q^-(\theta^-)$.

3.2 Definition of the cost function

Let us propose a cost function as

$$\Gamma_\Theta(\Theta^0, \bar u, \Theta^1) := \frac{1}{2}(\Theta^0 - \Phi(\Theta^1))^{\mathrm{T}}\Lambda_x\,(\Theta^0 - \Phi(\Theta^1)) + \frac{1}{2}\int_{t^0}^{t^1} \bar u(\tau)^{\mathrm{T}}\Lambda_{\bar u}\,\bar u(\tau)\mathrm{d}\tau$$

$$= \frac{1}{2}(\psi_0(x^0) - \psi_1(x^1))^{\mathrm{T}}\Lambda_x\,(\psi_0(x^0) - \psi_1(x^1)) + \frac{1}{2}\int_{t^0}^{t^1} \bar u(\tau)^{\mathrm{T}}\Lambda_{\bar u}\,\bar u(\tau)\mathrm{d}\tau \quad =: \hat\Gamma(x^0, \bar u, x^1), \qquad (13)$$

where Λ_x and $\Lambda_{\bar u}$ represent appropriate positive definite matrices. Here $\Phi(\Theta^1)$ is defined to be the angles and its velocities just after the transition with exchanging legs as

$$\Phi(\Theta^1) := \begin{pmatrix} \theta^+ \\ \dot\theta^+ \end{pmatrix} = \begin{pmatrix} C & 0 \\ 0 & \mathbf{Q}(\theta^-) \end{pmatrix}\Theta^1. \qquad (14)$$

Regarding the relation between x and Θ, we have

$$x = \begin{pmatrix} q \\ p \end{pmatrix} = \begin{pmatrix} \theta \\ M(\theta)\dot{\theta} \end{pmatrix} = \begin{pmatrix} I & 0 \\ 0 & M(\theta) \end{pmatrix} \Theta. \tag{15}$$

Equations (14) and (15) imply that $\psi_1(x^1)$ and $\psi_0(x^0)$ are also defined as

$$\psi_1(x^1) := \begin{pmatrix} C & 0 \\ 0 & \mathbf{Q}(\theta^-)M(\theta^-)^{-1} \end{pmatrix} x^1$$

$$\psi_0(x^0) := \begin{pmatrix} I & 0 \\ 0 & M(\theta^0)^{-1} \end{pmatrix} x^0. \tag{16}$$

3.3 Derivation of the iteration law

This subsection derives the iteration law with respect to the cost function defined in (13) based on the steepest descent method referred in Subsection 2.2. Let us calculate the Fréchet derivative of the cost function (13) as follows

$$\begin{aligned}
&\delta\,\hat{\Gamma}(x^0, \bar{u}, x^1)(\delta x^0, \delta\bar{u}, \delta x^1, \delta y) \\
&= \langle \big(\Lambda_x(\psi_0(x^0) - \psi_1(x^1)), \Lambda_{\bar{u}}\bar{u}\big), \big(\delta\psi_0(x^0)(\delta x^0), \delta\bar{u}\big)\rangle \\
&+\langle \big(\Lambda_x(\psi_0(x^0) - \psi_1(x^1)), 0\big), \big(-\delta\psi_1(x^1)(\delta x^1), \delta y\big)\rangle \\
&= \langle \big(\delta\psi_0(x^0)^*\Lambda_x(\psi_0(x^0) - \psi_1(x^1)), \Lambda_{\bar{u}}\bar{u}\big) \\
&+\delta\Sigma(x^0, \bar{u})^* \left(-\delta\psi_1(x^1)^*\Lambda_x(\psi_0(x^0) - \psi_1(x^1)), 0\right), (\delta x^0, \delta\bar{u})\rangle \\
&.
\end{aligned} \tag{17}$$

Suppose Eq. (5) holds, then using Eqs. (5) and (17), iterative learning control law for the control input is given by

$$\begin{cases} x^0_{(2i)} = x^0_{(1)} - \epsilon_{(i)}(J-R)T\delta\psi_1(x^1_{(2i-1)})^* \\ \qquad\qquad \times \Lambda_x(\psi_0(x^0_{(2i-1)}) - \psi_1(x^1_{(2i-1)})) \\ \bar{u}_{(2i)} = \bar{u}_{(2i-1)} \end{cases} \tag{18}$$

$$\begin{cases} x^0_{(2i+1)} = x^0_{(1)} \\ \bar{u}_{(2i+1)} = \bar{u}_{(2i-1)} - K_{(i)}\big(\Lambda_{\bar{u}}\bar{u}_{(2i-1)} \\ \qquad\qquad + \dfrac{1}{\epsilon_{(i)}}\mathcal{R}(y_{(2i)} - y_{(2i-1)})\big) \end{cases} \tag{19}$$

provided that the initial input $\bar{u}_{(1)} \equiv 0$ and the first initial condition $x^0_{(1)}$ is appropriately chosen. Here $\epsilon_{(\cdot)}$ denotes a sufficiently small positive constant and an appropriate positive definite matrix $K_{(\cdot)}$ represents a gain. The pair of iteration laws (18) and (19) implies that this learning procedure needs two experiments to execute a single update step in the steepest decent method. In the $2i$-th iteration, we can get the output signal of $\Sigma\big(x^0-(J-R)T\xi, \bar{u}+\mathcal{R}(v)\big)$ in Eq. (5) (note that in this case $v \equiv 0$.) and then we can calculate the input and output signals of $(\mathrm{d}\Sigma)^*$ from Eq. (5). The input for the $(2i+1)$-th iteration is generated by Eq. (8) with these signals.

Equations (18) and (19) imply that the procedure requires the precise knowledge of $\delta\psi_1(x^1)$ which includes the transition equation and the information of the inertia matrix is also necessary because the state x includes the momentum $p = M(\theta)\dot{\theta}$. Regarding real robots, however, the numerical model of the state transition does not hold exactly. For example, the angular momentum is not always preserved. And it is sometimes difficult to get the accurate inertia information of robots. So in the next section, we derive a learning algorithm with respect to Θ not the state x which estimates the transition mapping by experimental data.

4. MODIFICATION OF THE ITERATION LAW

In this section, we reconsider the iteration laws (18) and (19) in order not to use the knowledge of the state transition mapping and the inertia matrix. This method allows one to execute the learning procedure by only using the information of the output and the angular velocities just before and after transition.

Let us approximate the cost function (13) by the following one so that we deal with the functional of the output $y = \theta$ and its time derivative instead of the state

$$\begin{aligned}
&\tilde{\Gamma}(y, \dot{y}, \bar{u}) := \\
&\frac{1}{2}\int_{t^0}^{t^1} (y(\tau) - C\mathcal{R}(y)(\tau))^{\mathrm{T}}\Lambda_y(\tau)(y(\tau) - C\mathcal{R}(y)(\tau))\mathrm{d}\tau \\
&+\frac{1}{2}\int_{t^0}^{t^1}(\dot{y}(\tau)-\mathcal{R}(f_Q(y,\dot{y}))(\tau))^{\mathrm{T}}\Lambda_{\dot{y}}(\tau)(\dot{y}(\tau)-\mathcal{R}(f_Q(y,\dot{y}))(\tau)) \\
&+\frac{1}{2}\int_{t^0}^{t^1} \bar{u}(\tau)^{\mathrm{T}}\Lambda_{\bar{u}}(\tau)\bar{u}(\tau)\mathrm{d}\tau,
\end{aligned} \tag{20}$$

where $\Lambda_{(\cdot)}$'s represent the weighting function with respect to y, $\dot{y}$ and $\bar{u}$ respectively. Since the transition mapping (12) is generally a nonlinear function with respect to θ^- and $\dot{\theta}^-$, we consider the transition mapping to be a nonlinear function described by $\dot{\theta}^+ = f_Q(\theta^-, \dot{\theta}^-)$ in what follows. Equation (12) gives a special case of f_Q. We define the weighting functions Λ_y and $\Lambda_{\dot{y}}$ as $\mathrm{diag}(k_{y_1}, k_{y_2})\Lambda(t)$ and $\mathrm{diag}(k_{\dot{y}_1}, k_{\dot{y}_2})\Lambda(t)$, where $k_{(\cdot)}$'s are positive constants and

$$\Lambda(t) := \begin{cases} \dfrac{1}{2}\left(1 - \cos\left(\dfrac{\Delta t - t}{\Delta t - t^0}\pi\right)\right) & (t^0 \le t \le \Delta t) \\ 0 & (\Delta t < t \le t^1) \end{cases} . \tag{21}$$

Due to the weighting function $\Lambda(t)$, we can evaluate θ^0 and $\dot{\theta}^0$ approximately by choosing Δt sufficiently close to t^0.

In the previous work of the iterative learning control in Fujimoto and Sugie [2003], it is not possible to choose a functional of the time derivative of the output $\dot{y}$. We proposed an extension by employing a pseudo adjoint of the time derivative operator to take the time derivative of the output into account in Satoh et al. [2006b].

Lemma 1. Satoh et al. [2006b] Consider differentiable signals ξ and $\eta \in L_2[t^0, t^1]$ and a time derivative operator $D(\cdot)$ which maps the signal ξ into its time derivative is defined by

$$D(\xi)(t) := \frac{\mathrm{d}\xi(t)}{\mathrm{d}t}.$$

Suppose that the signal ξ satisfies the condition

$$\xi(t^0) = \xi(t^1) = 0.$$

Then the following equation holds.

$$\langle \eta, D(\xi)\rangle_{L_2} = \langle -D(\eta), \xi\rangle_{L_2} \tag{22}$$

Assumption 3. In order to utilize Eq. (22), we assume that the following conditions hold

$$\|y_{(j)}(t^0) - y_{(j-1)}(t^0)\| \ll 1 \tag{23}$$

$$\|y_{(j)}(t^1) - y_{(j-1)}(t^1)\| \ll 1. \tag{24}$$

In the existing iterative learning control framework, it is assumed that all the initial conditions are the same, which is almost equivalent to the condition (23). In our method, an additional condition (24) is assumed. In order to let Eq. (24) approximately hold, we add an extra cost function (25) to Eq. (20) with an appropriately large positive gain K_y

$$\int_{t^1-\epsilon}^{t^1} K_y \|y(\tau) - C\theta^0 \mathbf{1}(\tau)\|^2 \mathrm{d}\tau, \tag{25}$$

where ϵ represents a sufficiently small positive constant and $\mathbf{1}(t)$ represents an identity operator $\mathbf{1}(t) = 1(t^0 \leq t \leq t^1)$. The cost function (25) weights the difference between the output and its desired value around $t = t^1$ and penalizes the left hand side of the inequality (24).

Let us calculate the Fréchet derivative of the cost function (20) as follows

$$\begin{aligned}
&\delta\,\tilde{\Gamma}(y,\dot{y},\bar{u})(\delta y,\delta\dot{y},\delta\bar{u}) \\
&= \langle \Lambda_y(y - C\mathcal{R}(y)), \delta y - C\mathcal{R}(\delta y)\rangle + \langle \Lambda_{\dot{y}}(\dot{y} - \mathcal{R}(f_Q(y,\dot{y}))), \\
&\quad \delta\dot{y} - \mathcal{R}(\partial_y f_Q(y,\dot{y})(\delta y)) + \partial_{\dot{y}} f_Q(y,\dot{y})(\delta\dot{y})\rangle + \langle \Lambda_{\bar{u}}\bar{u}, \delta\bar{u}\rangle \\
&= \langle (\mathrm{id} - \mathcal{R}C)\Lambda_y(\mathrm{id} - C\mathcal{R})(y) - \partial_y f_Q(y,\dot{y})^*\mathcal{R} \\
&\quad \Lambda_{\dot{y}}(\dot{y} - \mathcal{R}(f_Q(y,\dot{y}))) - D\big((\mathrm{id} - \partial_{\dot{y}} f_Q(y,\dot{y})^*\mathcal{R}) \\
&\quad \Lambda_{\dot{y}}(\dot{y} - \mathcal{R}(f_Q(y,\dot{y})))\big), \delta y\rangle + \langle \Lambda_{\bar{u}}\bar{u}, \delta\bar{u}\rangle \\
&=: \langle \tilde{\Gamma}'_y, \delta y\rangle + \langle \tilde{\Gamma}'_{\bar{u}}, \delta\bar{u}\rangle \\
&= \langle \tilde{\Gamma}'_{\bar{u}} + (\delta\Sigma^{x^0}(\bar{u}))^*\tilde{\Gamma}'_y, \delta\bar{u}\rangle.
\end{aligned} \tag{26}$$

Here $\Sigma^{x^0}(\bar{u}) : U \to Y$ represents a simpler notation of $\Sigma(x^0, \bar{u})$ under a fixed initial condition x^0. $\partial_y f_Q(y,\dot{y})$ and $\partial_{\dot{y}} f_Q(y,\dot{y})$ represent the partial Fréchet derivative of $f_Q(y,\dot{y})$ with respect to y and $\dot{y}$, respectively. Using Eqs. (5) and (26), the iterative learning control law for the control input is calculated as below. The detail of the derivation is omitted due to the limitation of space

$$\bar{u}_{(2i)} = \bar{u}_{(2i-1)} + \epsilon_{(i)}\mathcal{R}\Big(\tilde{\Gamma}'_{y(2i-1)}\Big) \tag{27}$$

$$\bar{u}_{(2i+1)} = \bar{u}_{(2i-1)} - K_{(i)}\big(\tilde{\Gamma}'_{\bar{u}(2i-1)} + \frac{1}{\epsilon_{(i)}}\mathcal{R}(y_{(2i)} - y_{(2i-1)})\big) \\ . \tag{28}$$

We can calculate $\tilde{\Gamma}'_{y(\cdot)}$ by estimating the Jacobian of f_Q at the touchdown, since $\frac{\partial f_Q}{\partial(y,\dot{y})} = (\partial_y f_Q, \partial_{\dot{y}} f_Q)$ holds and the weighting function Λ_t defined by Eq. (21) dismisses for $t \in [\Delta t, t^1]$ by its definition. Since the following equation holds

$$\mathrm{d}\dot{y}^+ = \frac{\partial f_Q}{\partial(y,\dot{y})}\begin{pmatrix}\mathrm{d}y^- \\ \mathrm{d}\dot{y}^-\end{pmatrix}, \tag{29}$$

let us estimate the Jacobian $\frac{\partial f_Q}{\partial(y,\dot{y})}$ by experimental data via the least-squares method. We define the following data sets as

$$\Delta Y^-_{(n)} := \begin{bmatrix} y^{-\mathrm{T}}_{(1)} - y^{-\mathrm{T}}_{(n-1)}, \dot{y}^{-\mathrm{T}}_{(1)} - \dot{y}^{-\mathrm{T}}_{(n-1)} \\ \vdots \\ y^{-\mathrm{T}}_{(n-2)} - y^{-\mathrm{T}}_{(n-1)}, \dot{y}^{-\mathrm{T}}_{(n-2)} - \dot{y}^{-\mathrm{T}}_{(n-1)} \end{bmatrix},$$

$$\Delta\dot{Y}^+_{(n)} := \begin{bmatrix} \dot{y}^{+\mathrm{T}}_{(1)} - \dot{y}^{+\mathrm{T}}_{(n-1)} \\ \vdots \\ \dot{y}^{+\mathrm{T}}_{(n-2)} - \dot{y}^{+\mathrm{T}}_{(n-1)} \end{bmatrix}. \tag{30}$$

The size of $\Delta Y^-_{(n)}$ is $(n-2)\times 4$ and that of $\Delta\dot{Y}^+_{(n)}$ is $(n-2)\times 2$. By Eq. (29), we have

$$\Delta\dot{Y}^+_{(n)} = \Delta Y^-_{(n)}\frac{\partial f_Q}{\partial(y,\dot{y})}^{\mathrm{T}}. \tag{31}$$

By solving Eq. (31), we obtain

$$\frac{\partial f_Q}{\partial(y,\dot{y})}^{\mathrm{T}} = \Delta Y^{-\,\dagger}_{(n)}\Delta\dot{Y}^+_{(n)}, \tag{32}$$

where $(\cdot)^\dagger$ represents the pseudo inverse matrix of $(\cdot)$. We can also utilize MATLAB's arithmetic operator of the matrix left division to solve Eq. (31) easily.

Here let us summarize the proposed learning procedure.

Step 0: Set the initial condition x^0 and $k = 1$. Then, go to Step k.

Step $k(1 \leq k \leq 5)$: Execute k-th of 5 preliminary experiments in order to obtain data sets for the first control input under appropriate initial conditions around x^0 and zero control input. If $k \leq 4$, set $k = k+1$ and go to Step k. Otherwise set $i = 3$ and go to Step 6.

Step $k = 2i(i \geq 3)$: Utilizing data sets $\Delta Y^-_{(2i)}$ and $\Delta\dot{Y}^+_{(2i)}$ in Eq. (30) and Eq. (32), estimate the Jacobian $\frac{\partial f_Q}{\partial(y,\dot{y})}$. With the initial condition x^0, executes the $2i$-th laboratory experiment via the iteration law of (27). and go to Step $2i$+1.

Step $k = 2i+1(i \geq 3)$: With the initial condition x^0, executes the $2i$+1-th laboratory experiment via the iteration law of (28). Set $i = i+1$ and go to Step $2i$.

5. SIMULATION

We apply the proposed algorithm in the previous section to the compass gait biped depicted in Fig. 1 to generate an optimal gait trajectory on the level ground. We proceed 80 steps of the learning procedure which means 160 simulations with the initial condition $(\theta_1^0, \theta_2^0, \dot{\theta}_1^0, \dot{\theta}_2^0)^{\mathrm{T}} = (-0.23, 0.25, 1.5, -1)^{\mathrm{T}}$. The design parameters of the cost function (20) are $(k_{y_1}, k_{y_2}, k_{\dot{y}_1}, k_{\dot{y}_2}) = (10, 10, 1\times 10^{-2}, 1\times 1^{-2})$ and $\Lambda_{\bar{u}} = \mathrm{diag}(1\times 10^{-5}, 1\times 10^{-5})$. Parameters of the learning procedures are $\epsilon_{(\cdot)} = 1$ and $K_{(\cdot)} = \mathrm{diag}(60, 5)$.

Fig. 2 shows the history of the cost function (20) along the iteration decreasing monotonically. It implies that the output trajectory converges to an optimal one smoothly. Figs. 3 and 4 show responses of θ and $\dot{\theta}$ at the last step in the proposed method in solid lines and those at the 1,40,80 and 160th steps in dotted lines. Fig. 5 shows the control inputs generated in the last iteration. Fig. 6 shows the phase portrait of $\theta^1 - \dot{\theta}^1$. It exhibits that a limit cycle which implies a periodic motion generated.

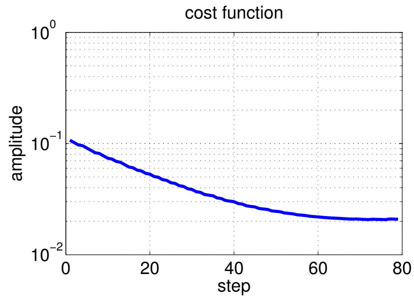

Fig. 2. Cost function

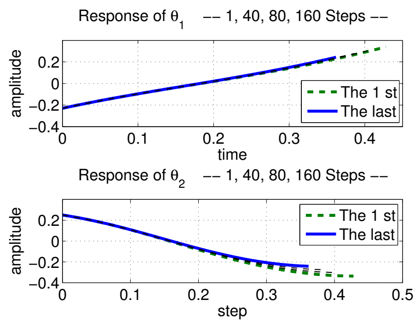

Fig. 3. Responses of θ_1 and θ_2

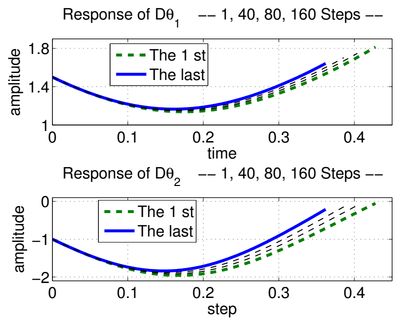

Fig. 4. Responses of $\dot{\theta}_1$ and $\dot{\theta}_2$

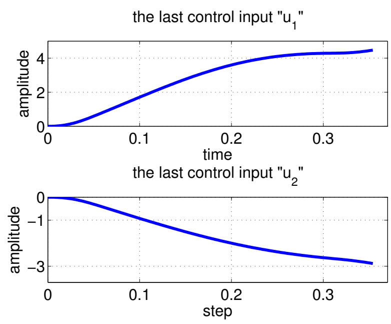

Fig. 5. Generated control inputs

6. CONCLUSION

In this paper, we have proposed a gait generation framework including discrete state transitions. The proposed method can generate optimal feedforward control input for energy efficient gait by combining iterative learning control based on a symmetric property of Hamiltonian systems and the state transition mapping estimation based on the least-squares. It does not require the precise knowledge

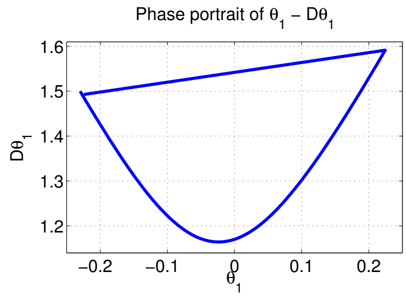

Fig. 6. Phase portrait of $\theta_1 - \dot{\theta}^1$

of the plant system nor the numerical transition model which is not accurate for real robots. Applying this method to the compass gait biped, we generate an optimal gait trajectories on the level ground. Numerical simulations demonstrate the effectiveness of the proposed framework. Because our proposed method does not require the information intrinsic the robot itself, it is expected that it is valid for general walking robots.

REFERENCES

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.

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.

A. Goswami, B. Thuilot, and B. Espiau. Compass-like biped robot part i: Stability and bifurcation of passive gaits. *INRIA Research Report*, (2996), 1996.

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.

T. McGeer. Passive dynamic walking. *Int. J. Robotics Research*, 9(2):62–82, 1990.

K. Osuka and K. Kirihara. Motion analysis and experiments of passive walking robot quartet ii. In *Proc. the 2000 IEEE Int. Conf. on Robotics and Automation*, pages 3052–3056, 2000.

A. Sano, Y. Ikemata, and H. Fujimoto. Analysis of dynamics of passive walking from storage energy and supply rate. In *Proc. the 2003 IEEE Int. Conf. on Robotics and Automation*, pages 2478–2483, 2003.

S. Satoh, K. Fujimoto, and S. Hyon. Optimal gait generation for the compass gait biped based on variational symmetry. In *Proc. 35th SICE Symposium on Control Theory (in Japanese)*, pages 357–360, 2006a.

S. Satoh, K. Fujimoto, and S. Hyon. Gait generation for a hopping robot via iterative learning control based on variational symmetry. In *Proc. IFAC 3rd Workshop on Lagrangian and Hamiltonian Methods for Nonlinear Control*, pages 125–130, 2006b.

S. Satoh, K. Fujimoto, and S. Hyon. Gait generation for passive running via iterative learning control. In *Proc. IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 5907–5912, 2006c.

M. W. Spong. Passivity-base control of the compass gait biped. In *Proc. of IFAC World Congress*, pages 19–23, 1999.