A framework for optimal gait generation via learning optimal control using virtual constraint

Satoshi Satoh, Kenji Fujimoto and Sang-Ho Hyon

***Abstract*— This paper proposes an optimal gait generation framework using virtual constraint and learning optimal control. In this method, firstly, we add a constraint by a virtual potential energy to prevent the robot from falling. Secondly, we execute iterative learning control (ILC) to generate an optimal feedforward input. Thirdly, we execute iterative feedback tuning (IFT) to mitigate the strength of the virtual constraint automatically according to the progress of learning control. Consequently, it is expected to generate an optimal gait without constraint eventually. Although existing ILC frameworks require a lot of experimental data under the same initial condition, the proposed method does not need to repeat experiments under the same initial condition because the virtual constraint restricts the motion of the robot to a symmetric trajectory. Furthermore, it does not require the precise knowledge of the plant system. Finally, some numerical simulations demonstrate the effectiveness of the proposed method.**

I. Introduction

In the recent active research and development regarding humanoid robots, a lot of techniques to realize dynamic bipedal walking have been proposed. Many conventional frameworks for bipedal walking control are classified as motion planning based on the zero moment point (ZMP) based control. Dynamic walking control based on *passive dynamic walking* [1] attracts attention, e.g., [2], [3], [4], as is antithetical to the ZMP based control with respect to the energy consumption. As an alternative, walking control methods using virtual constraint based on the output zeroing control are proposed [5], [6].

In [7], [8], we have studied optimal gait generation in terms of the energy efficiency via iterative learning control (ILC) proposed in [9] which utilizes a property of Hamiltonian systems. Thanks to the advantage of the method, our technique does not require precise information about the plant system. Instead, existing ILC frameworks require a lot of laboratory experiments under the same initial condition. It is sometimes difficult to repeat experiments under the same initial condition, because it is difficult to realize the desired initial velocity of the mechanical systems including walking robots.

To solve the problem, in this paper, we propose an optimal gait generation framework using virtual constraint and learning optimal control. The proposed method is summarized as follows. Firstly, we add a constraint by adding a virtual potential energy to prevent the robot from falling. Secondly, we execute the learning procedure as is proposed in our previous works [7], [8]. The proposed technique restricts the motion of the robot to a symmetric trajectory by the virtual constraint. Due to this, it does not need to repeat experiments under the same initial condition. Thirdly, by regarding the potential gain for the constraint as a tuning parameter, we execute iterative feedback tuning (IFT) to mitigate the strength of the virtual constraint automatically according to the progress of learning control. Consequently, it is expected to generate an optimal gait without constraint eventually. Let us note that the proposed method differs from the existing techniques using virtual constraint in that our method automatically optimizes the strength of the constraint. Finally, some numerical simulations demonstrate the effectiveness of the proposed method.

S. Satoh is with Department of Mechanical Science and Engineering, Graduate School of Engineering, Nagoya University, Furo-cho, Chikusa-ku, Nagoya-shi, Aichi 464-8603, Japan s.satoh@nagoya-u.jp

K. Fujimoto is with Department of Mechanical Science and Engineering, Graduate School of Engineering, Nagoya University, Furo-cho, Chikusa-ku, Nagoya-shi, Aichi 464-8603, Japan k.fujimoto@ieee.org

S. Hyon is with JST, ICORP, Computational Brain Project, Saitama 332-0012, Japan and ATR Computational Neuroscience Laboratories, Kyoto 619-0288, Japan sangho@atr.jp

II. Learning optimal control of Hamiltonian systems

This section refers to some of the basic results about learning optimal control of Hamiltonian systems, particularly about **iterative learning control (ILC)** and **iterative feedback tuning (IFT)** based on variational symmetry originally proposed in [9] and [10].

A. Variational symmetry of Hamiltonian systems

Consider a Hamiltonian system with dissipation and a controlled Hamiltonian $H(x,u,t)$ described by

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

Here $x(t) \in \mathbb{R}^n$, $u(t), y(t) \in \mathbb{R}^m$ 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 consider behaviors of the system (1) at a finite time interval $[t^0, t^1]$ and often describe the system as $\Sigma : L_2^m[t^0,t^1] \rightarrow L_2^m[t^0,t^1] : u \mapsto y$. The variational system $\mathrm{d}\Sigma$ of the system Σ represents the Fréchet derivative of Σ. The following theorem with respect to $\mathrm{d}\Sigma$ holds. This property is called **variational symmetry** of Hamiltonian control systems.

Theorem 1: [9] 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

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

Suppose moreover that $J-R$ is nonsingular. Then the adjoint of the variational system $(\mathrm{d}\Sigma)^*$ has the time-reversal state-space realization of the variational system $\mathrm{d}\Sigma$.

Remark 1: Suppose the Hessian of the Hamiltonian with respect to (x,u) is satisfying

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

Then, under the appropriate initial condition of Σ, Equation (3) holds

$$(\mathrm{d}\Sigma(u))^*(v) \approx \frac{1}{\epsilon}\mathcal{R}\circ(\Sigma(u+\epsilon\mathcal{R}(v)) - \Sigma(u)), \tag{3}$$

where ϵ represents sufficiently small positive constant and $\mathcal{R}$ is a time-reversal operator defined by $\mathcal{R}(u)(t-t^0) = u(t^1-t)$ for $\forall t \in [t^0,t^1]$.

Equation (3) implies that one can calculate the input-output mapping of the adjoint by only using the input-output data of the original system. The literature [9] says that one can utilize the property (3) in considering general mechanical systems.

B. Iterative learning control and iterative feedback tuning

We review some results of ILC in [9] and IFT in [10]. They have common feature that they take advantage of variational symmetry of Hamiltonian systems. The objective of ILC is to find an optimal feedforward input which minimizes a given cost function, while that of IFT is to find optimal parameters of a given feedback controller. Figure 1 illustrates the difference of the concept between them.

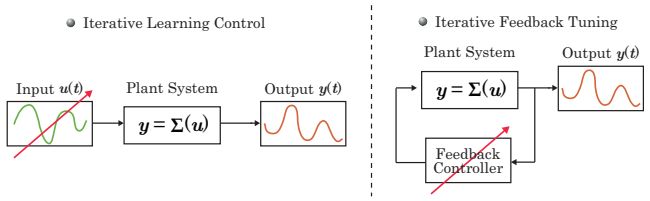

Fig. 1. Illustrations of ILC and IFT

Firstly, let us mention ILC in more detail. Consider the system Σ in (1) and a cost function $\Gamma : L_2^m[t^0,t^1] \times L_2^m[t^0,t^1] \to \mathbb{R}$. This technique is based on the steepest descent method. The gradient of the cost function Γ with respect to the control input u is calculated as

$$\begin{aligned} \mathrm{d}\Gamma(u,y)(\mathrm{d}u,\mathrm{d}y) &= \langle \nabla_u\Gamma(u,y),\mathrm{d}u\rangle + \langle \nabla_y\Gamma(u,y),\mathrm{d}y\rangle \\ &= \langle \nabla_u\Gamma(u,y) + (\mathrm{d}\Sigma(u))^*\nabla_y\Gamma(u,y),\mathrm{d}u\rangle =: \langle \Gamma'_u,\mathrm{d}u\rangle. \end{aligned} \tag{4}$$

Here $\mathrm{d}\Gamma(u,y)$ represents the Fréchet derivative of Γ and note that $\mathrm{d}y = \mathrm{d}\Sigma(u)(\mathrm{d}u)$. It follows from well-known Riesz's representation theorem that there exists operators $\nabla_u\Gamma(u,y)$ and $\nabla_y\Gamma(u,y)$ as above. The steepest descent method implies that we should update the input u such that

$$u_{(i+1)} = u_{(i)} - K_{(i)}\Gamma'_u\Big|_{u=u_{(i)},y=y_{(i)}}, i = 0,1,2,\cdots, \tag{5}$$

where K is an appropriate positive gain and i denotes the i-th iteration in laboratory experiment. In calculating Γ'_u, one can calculate $\nabla_u\Gamma(u,y)$ and $\nabla_y\Gamma(u,y)$ in (4) by information of u and y. However, the precise knowledge of the system is generally required to calculate $(\mathrm{d}\Sigma(u))^*$. This method utilizes variational symmetry (3) to solve this problem as mentioned in Remark 1.

Secondly, IFT proposed in [10] is mentioned in more detail. We consider feedback controllers $u = C(\rho,x)$ under which Hamiltonian structure is preserved, i.e., the closed loop system is again Hamiltonian system. Here $\rho \in \mathbb{R}^k$ represents gain parameter. For example, a class of such feedback controllers is the generalized canonical transformation [11]. In this method, gain parameters of a given feedback controller are considered to be virtual inputs for a Hamiltonian system to utilize variational symmetry. The algorithm is similar to that of ILC. We define the zeroth-order hold operator which maps the parameter $\rho \in \mathbb{R}^k$ to the L_2^k space in order to define the virtual input.

Definition 1: Consider $\xi \in \mathbb{R}^k$ and $w \in L_2^k$. We define an operator $\mathfrak{h}$ satisfying the following equation as the zeroth-order hold operator.

$$\begin{aligned} \mathfrak{h} : \mathbb{R}^k &\to L_2^k : \xi \mapsto w \\ w(t) &= \begin{cases} \xi & (t^0 \le t \le t^1) \\ 0 & (t < t^0, t > t^1) \end{cases} \end{aligned} \tag{6}$$

When we define the virtual input as $u_\rho := \mathfrak{h}(\rho)$. Then the corresponding output which induces variational symmetry (3) is given by

$$y_\rho := -\frac{\partial \bar{H}(x,t,\rho)}{\partial \rho}^{\mathrm{T}},$$

where $\bar{H}$ denotes the Hamiltonian of the closed loop system. Utilizing u_ρ and y_ρ, one can update parameters in the similar manner (5) as in the case of ILC.

Remark 2: A typical mechanical system can be described by a Hamiltonian system

$$\Sigma : \begin{cases} \begin{pmatrix} \dot{q} \\ \dot{p} \end{pmatrix} = \begin{pmatrix} 0 & I \\ -I & -D \end{pmatrix} \begin{pmatrix} \frac{\partial H(q,p,u)}{\partial q}^{\mathrm{T}} \\ \frac{\partial H(q,p,u)}{\partial p}^{\mathrm{T}} \end{pmatrix} \\ y = -\frac{\partial H(q,p,u)}{\partial u}^{\mathrm{T}} = q \end{cases} \tag{7}$$

with the Hamiltonian

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

where $q,p \in \mathbb{R}^m$ and a positive matrix $M(q)$ denotes the inertia matrix, a positive semidefinite matrix D denotes the friction coefficients and a scalar function $V(q)$ denotes the potential energy of the system. It is proven that we can always apply ILC and IFT methods mentioned in this section to the system described in (7). See [9], for the detail.

TABLE I

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	$\mathrm{m/s^2}$
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

III. OPTIMAL GAIT GENERATION VIA LEARNING OPTIMAL CONTROL USING VIRTUAL CONSTRAINT

This section proposes our framework. Firstly, we refer to a constraint by adding virtual potential energies, which plays an important role in our method. Then, we define a cost function and derive the learning iteration laws.

A. Description of the plant

We consider a fully actuated planar biped robot called the **compass gait biped** [12] depicted in Fig. 2 and afterward we also consider the one with a torso depicted in Fig. 6 as a more general walking robot model. Table I shows physical parameters and variables. Assumptions on this robot conforms [12] and they are omitted here. We use number of notations with respect to the state. Table II summaries them. 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 (7) with the Hamiltonian

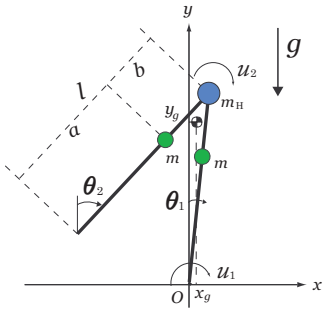

Fig. 2. The compass gait biped

TABLE II

SOME NOTATIONS

Notation	Meaning
$q := (q_1, q_2)^{\mathrm{T}} = (\theta_1, \theta_2)^{\mathrm{T}}$	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
$.^{-(+)}$	just before (after) transfer

$$H(q, p, \bar{u}) = \frac{1}{2} p^{\mathrm{T}} M(q)^{-1} p + V(q) - \bar{u}^{\mathrm{T}} q \tag{8}$$

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

$$M(q) = \begin{pmatrix} m_H l^2 + ma^2 + ml^2 & -mbl\cos(q_1 - q_2) \\ -mbl\cos(q_1 - q_2) & mb^2 \end{pmatrix}$$
$$V(q) = \{(m_H l + ma + ml)\cos q_1 - mb\cos q_2\} g. \tag{9}$$

Note that using (9), the generalized momentum is described as $p = M(q)\dot{q}$. Following the law of conservation of the angular momentum, a transition equation is obtained. The detail of the equation conforms [12], so it is omitted here.

B. Constraint by adding virtual potential energies

In the literatures [5], [6], walking control methods using virtual constraint based on the output zeroing control are proposed. In [6], especially, they can achieve stable symmetric walking gaits using another property of Hamiltonian systems than that used in this paper. They set the output function $y = h(x) = q_1 + q_2$ to zero by the output zeroing control and keep the leg angles bounded by a leg exchange scheme. As a consequence, they guarantee that the robot does not fall and obtain symmetric walking gaits satisfying $q_1 + q_2 = 0$.

On the other hand, we use a similar concept of the virtual constraint to prevent the robot from falling, but do not use the output zeroing control. There are two reasons: one is that the output zeroing control requires the precise knowledge of the plant system and the other is that such constraints consume a lot of control energy. We add a virtual potential energy such as Equation (10) to produce a similar effect to [6]

$$P_c := \frac{K_c}{2}(q_1 + q_2)^2. \tag{10}$$

Here, the gain parameter K_c represents the constraint strength. We make K_c sufficiently large at the beginning of the learning steps so that the trajectory of the robot is restricted to symmetric one, i.e. $q_1 + q_2 = 0$ holds. Due to [6], it is expected that the robot does not fall. The advantages of this method instead of the output zeroing are as follows. Firstly, it does not require the model parameters of the plant system, since the potential energy (10) can be generated by a simple proportional feedback $-K_c(q_1 + q_2)$. Secondly, after adding the potential energy, the plant system preserve the Hamiltonian structure and the constraint parameter K_c is explicitly contained in the Hamiltonian. By regarding K_c as a tuning parameter, we try to adjust the constraint strength by IFT mentioned in Section II-B. The idea of the proposed framework is the following.

- Step1 : Add a virtual energy to restrict the motion of the robot to a symmetric trajectory. Then, make constraint parameter K_c sufficiently large to expect that the robot does not fall.
- Step2 : Execute ILC procedure to generate an optimal walking gait as our previous work in [7].
- Step3 : Mitigate the constraint parameter by IFT automatically according to the progress of learning control.

Step4 : As a result, it is expected that an optimal gait is generated without a constraint eventually.

The feature of the proposed method is that the robot improves his walk keeping on walking, because the robot does not fall due to Step1. Our method also differs from the past proposed ones using virtual constraint in that it automatically optimizes the strength of the constraint.

C. Derivation of the iteration laws

Let us consider a Hamiltonian $\bar{H}$ of the closed loop system adding the virtual potential energy (10) by the proportional feedback $-K_c(q_1+q_2)$ as bellow

$$\bar{H}(q,p,\bar{u},K_c) = \frac{1}{2}p^{\mathrm{T}}M(q)^{-1}p + V(q) \\ -\bar{u}^{\mathrm{T}}q + \frac{K_c}{2}(q_1+q_2)^2.$$

The output which induces variational symmetry (3) with respect to $\bar{u}$ for the closed loop Hamiltonian system is given by

$$y := -\frac{\partial \bar{H}(q,p,\bar{u},K_c)}{\partial \bar{u}} = q.$$

In what follows, the operator $\bar{u} \mapsto y$ is described as $\bar{\Sigma}$ and each elements of the output y is described y_1 and y_2, respectively, that is, $y_1 = q_1$ and $y_2 = q_2$. We define the virtual input u_c to mitigate the constraint gain parameter K_c as $u_c := \mathfrak{h}(K_c)$. Then the corresponding output which induces variational symmetry is given by

$$y_c := -\frac{\partial \bar{H}(q,p,\bar{u},K_c)}{\partial K_c} = -\frac{(q_1+q_2)^2}{2}. \tag{11}$$

In what follows, the operator $u_c \mapsto y_c$ is described as Σ_c.

We propose a cost function as follows

$$\Gamma(y,\bar{u},u_c) := \\ \frac{1}{2}\int_{t^0}^{t^1} \big(y(\tau) - C\mathcal{R}(y)(\tau)\big)^{\mathrm{T}} \Lambda_y(\tau)\big(y(\tau) - C\mathcal{R}(y)(\tau)\big)\mathrm{d}\tau \\ + \frac{\gamma_{\bar{u}}}{2}\|\bar{u}\|_{L_2}^2 + \frac{\gamma_{u_c}}{2}\|u_c(y_1+y_2)\|_{L_2}^2 + \frac{\gamma_c}{2}\|u_c\|_{L_2}^2, \tag{12}$$

where the matrix C which exchanges the support and the swing leg angles is given by

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

Positive constants $\gamma_{\bar{u}}, \gamma_{u_c}$ and γ_c represent weighting coefficients for each terms, respectively. A symmetric positive definite matrix $\Lambda_y(t)$ represents the weighting function defined by

$$\Lambda_y(t) := \mathrm{diag}\left(\gamma_{y_1}\Lambda_t(t), \gamma_{y_2}\Lambda_t(t)\right), \\ \Lambda_t(t) := \begin{cases} \frac{1}{2}\left(1 - \cos\left(\frac{t^0+\Delta t - t}{\Delta t}\pi\right)\right) & (t^0 \le t \le t^0+\Delta t) \\ 0 & (t^0+\Delta t < t \le t^1), \end{cases} \tag{13}$$

where positive constants $\gamma_{y_1}, \gamma_{y_2}$ represents weighting coefficients for y_1 and y_2, respectively. Δt denotes a sufficiently small positive constant. Equation (13) implies that $\Lambda_t(t)$ plays a role of a weight function with respect to time to evaluate $y(t) - C\mathcal{R}(y)(t)$ at only around $t = t^0$. Fig. 3 illustrates $\Lambda_t(t)$. The physical meanings of each term of the cost function (12) are as follows. The first term is a restraint condition to satisfy a necessary condition for 1 period-periodic trajectories such that $\theta_1(t^1) \equiv \theta_2(t^0)$ and $\theta_2(t^1) \equiv \theta_1(t^0)$. Note that just after the collision between a leg and the ground, leg angles do not change and leg exchange between the support leg and the swing leg arises instantaneously, i.e. $\theta_1^+ \equiv \theta_2(t^1)$ and $\theta_2^+ \equiv \theta_1(t^1)$, so the following relation holds for 1 period-periodic trajectories $\theta_1(t^1) \equiv \theta_2(t^0)$ and $\theta_2(t^1) \equiv \theta_1(t^0)$ (see also Fig. 4). The second and the third terms are to minimize feedforward input $\bar{u}$ and the feedback input $K_c(y_1+y_2)$ which is applied for the virtual potential energy, respectively. The last term is to optimize the strength of the virtual constraint.

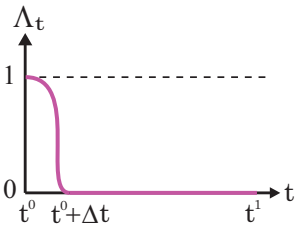

Fig. 3. Illustration of Λ_t

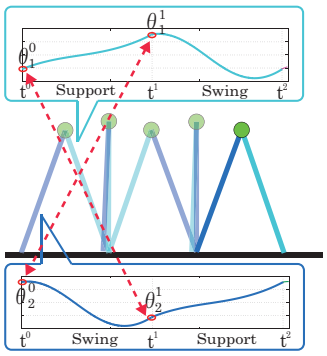

Fig. 4. Illustration of the restraint condition of the cost function

In our previous result [8], we also considered a necessary condition with respect to angular velocities $\dot{q}$ in the cost function. The condition is that initial angular velocities are equivalent to velocities just after touch down. We do not consider the condition in this paper, because it makes iteration law more complex. It is our future work. For another choice of cost function, we can deal with functionals with respect to the joint angle θ, its velocity $\dot{\theta}$ and the control input u.

We derive the updating law for the feedforward inputs for Steps 2 in the summary mentioned in the Section III-B. Let us calculate the Fréchet derivative of the cost function (12) as follows

$$\mathrm{d}\Gamma(y,\bar{u},u_c)(\mathrm{d}y,\mathrm{d}\bar{u}) = \langle \Lambda_y(y - C\mathcal{R}(y)), \mathrm{d}y - C\mathcal{R}(\mathrm{d}y)\rangle \\ + \langle \gamma_{\bar{u}}, \mathrm{d}\bar{u}\rangle + \langle \gamma_{u_c}u_c^2(1,1)y, (1,1)\mathrm{d}y\rangle.$$

We have the gradient of the cost function (12) with respect to $\bar{u}$ corresponding to Eq. (4) as follows

$$\begin{aligned}\Gamma'_{\bar{u}} &:= \nabla_{\bar{u}}\Gamma(\bar{u},y)+(\mathrm{d}\bar{\Sigma}(\bar{u}))^*\nabla_y\Gamma(\bar{u},y), \qquad (14)\\ \nabla_{\bar{u}}\Gamma &= \gamma_{\bar{u}}\bar{u},\\ \nabla_y\Gamma &= (\mathrm{id}-\mathcal{R}C)\Lambda_y(\mathrm{id}-\mathcal{R}C)(y)+\gamma_{u_c}u_c^2\begin{pmatrix}1 & 1\\ 1 & 1\end{pmatrix}y\end{aligned}$$

By using Eqs. (3) and (14), the updating law for the feed-forward input is given by

$$\bar{u}_{(2i+1)} = \bar{u}_{(2i)} + \epsilon_{1(2i)}\mathcal{R}\big(\nabla_y\Gamma_{(2i)}\big) \qquad (15)$$

$$\bar{u}_{(2i+2)} = \bar{u}_{(2i)} - K_{1(2i)}\Big(\nabla_{\bar{u}}\Gamma_{(2i)} + \frac{1}{\epsilon_{1(2i)}}\mathcal{R}(y_{(2i+1)}-y_{(2i)})\Big) \qquad (16)$$

provided that the initial input $\bar{u}_{(0)} \equiv 0$ and the first initial condition $x^0_{(0)}$ is appropriately chosen. Here $\epsilon_{1(\cdot)}$ denotes a sufficiently small positive constant and an appropriate positive definite matrix $K_{1(\cdot)}$ represents a gain. The pair of iteration laws (15) and (16) implies that this learning procedure needs two experiments to execute a single update step in the steepest decent method. In the $(2i$+1)-th iteration, we can get the output signal of $\Sigma\big(\bar{u}+\epsilon\mathcal{R}(v)\big)$ in Eq. (3) and then we can calculate the input and output signals of $(\mathrm{d}\Sigma)^*$ by using variational symmetry (3). The input for the $(2i$+2)-th iteration is generated by Eq. (5) with these signals.

Here we derive the updating law for the tuning parameter. Note that the third term of the cost function (12) is calculated as $\frac{\gamma_{u_c}}{2}\|u_c(y_1+y_2)\|^2_{L_2} = \gamma_{u_c}\|u_c\|^2_{L_2}\|y_c\|_{L_2}$, we have Fréchet derivative of the cost function as follows

$$\begin{aligned}&\mathrm{d}\Gamma(y,\bar{u},u_c)(\mathrm{d}y_c,\mathrm{d}u_c) =\\ &\langle\gamma_c u_c,\mathrm{d}u_c\rangle + \langle 2\gamma_{u_c}|y_c|u_c,\mathrm{d}u_c\rangle + \langle\gamma_{u_c}u_c^2\mathrm{sign}(y_c),\mathrm{d}y_c\rangle\\ &+\underline{\langle\gamma_{y_1}\Lambda_t(y_1-\mathcal{R}(y_2))+\gamma_{y_2}\Lambda_t(y_2-\mathcal{R}(y_1))}\\ &\qquad\underline{,\mathrm{d}y_1+\mathrm{d}y_2-\mathcal{R}(\mathrm{d}y_1)-\mathcal{R}(\mathrm{d}y_2)\rangle}_{\circledast} \qquad (17)\end{aligned}$$

Using $\mathrm{d}y_c = -(y_1+y_2)(\mathrm{d}y_1+\mathrm{d}y_2)$ from Eq. (11), let us calculate the gradient with respect to y_c from the rest part $\circledast$ in Eq. (17). For simplicity, we consider $\gamma_{y_1} = \gamma_{y_2}$ in Eq. (13) in what follows. Then, we have

$$\circledast = \langle(\mathrm{id}-\mathcal{R})\gamma_{y_1}\Lambda_t(\mathrm{id}-\mathcal{R})(y_1+y_2),\mathrm{d}y_1+\mathrm{d}y_2\rangle. \qquad (18)$$

Since Δt in Eq. (13) is sufficiently small, we approximate the left term in the inner product in Eq. (18) by

$$(\mathrm{id}-\mathcal{R})\gamma_{y_1}\Lambda_t(\mathrm{id}-\mathcal{R})(y_1+y_2)(t) \approx \begin{cases}2(y_1(t^0)+y_2(t^0)) & t=t^0\\ 0 & t^0<t<t^1\\ -2(y_1(t^1)+y_2(t^1)) & t=t^1\end{cases}. \qquad (19)$$

Note that $\theta_1+\theta_2 = 0$ holds when the swing leg of the compass gait biped touches down the ground, i.e. $y_1(t^0)+y_2(t^0) = y_1(t^1)+y_2(t^1) = 0$ holds. As a consequence, we have the gradient of the cost function (12) with respect to the constraint parameter as

$$\begin{aligned}&\mathrm{d}\Gamma(y,\bar{u},u_c)(\mathrm{d}y_c,\mathrm{d}u_c) \qquad (20)\\ &= \langle\gamma_c u_c - 2\gamma_{u_c}u_c y_c,\mathrm{d}u_c\rangle + \langle-\gamma_{u_c}u_c^2,\mathrm{d}\Sigma_c(u_c)(\mathrm{d}u_c)\rangle\\ &= \langle\mathfrak{h}^*\Big(\gamma_c u_c - 2\gamma_{u_c}u_c y_c - (\mathrm{d}\Sigma_c(u_c))^*(\gamma_{u_c}u_c^2)\Big),\mathrm{d}K_c\rangle,\end{aligned}$$

using $|y_c| = -y_c$ and $\mathrm{sign}(y_c) = -1$ by Eq. (11) and $u_c = \mathfrak{h}(K_c)$. We refer to the fact showed in [10] that for any $v \in L_2[t^0,t^1]$, the following equation holds

$$\mathfrak{h}^*v = \int_{t^0}^{t^1} v(\tau)\mathrm{d}\tau. \qquad (21)$$

By Eqs (3), (20) and (21), the updating law for the constraint parameter is given by

$$K_{c(2i+1)} = K_{c(2i)} - \epsilon_{2(2i)}\gamma_{u_c}K^{\,2}_{c(2i)} \qquad (22)$$

$$\begin{aligned}K_{c(2i+2)} = K_{c(2i)} &- K_{2(2i)}\int_{t^0}^{t^1}\gamma_c K_{c(2i)} - 2\gamma_{u_c}K_{c(2i)}y_{c(2i)}\mathrm{d}\tau\\ &-\frac{K_{2(2i)}}{\epsilon_{2(2i)}}\int_{t^0}^{t^1} y_{c(2i+1)} - y_{c(2i)}\mathrm{d}\tau. \qquad (23)\end{aligned}$$

The derivation is similar to that of the iteration laws (15) and (16), so it is omitted here.

Here let us summarize the proposed learning procedure.

Step 0: Set the initial control input as $\bar{u}_{(0)} \equiv 0$ and make the constraint parameter $K_{c(0)}$ sufficiently large. Let the robot start walking under an appropriate initial condition x^0.

Step $2i$+1:While $2i$+1 period, one utilizes the control input derived by (15) under the constraint gain derived by (22).

Step $2i$+2:While $2i$+2 period, one utilizes the control input derived by (16) under the constraint gain derived by (23).

In our framework, there are some open parameters, that is, ϵ_1, ϵ_2, K_1, K_2, γ_{y_1}, γ_{y_2}, $\gamma_{\bar{u}}$, γ_{u_c} and γ_c. First of all, all of them have to be positive. We should choose ϵ_1 and ϵ_2 small enough so that approximation in Eq.(3) holds. The parameters K_1 and K_2 are the step parameters in the steepest decent method. Generally we set K_1 and K_2 large in the beginning of learning, then we make them small gradually according to the leaning steps. γ_{y_1} and γ_{y_2} represent a constraint for periodic trajectories, so we choose them larger than other coefficients $\gamma_{\bar{u}}$, γ_{u_c} and γ_c.

Since it is only guaranteed that we can achieve a local minimum of a cost function because our framework is based on the steepest decent method as mentioned in Section II-B, a generated trajectory is not always physically valid, that is, it is not always a walking gait. Therefore we have to choose its initial condition in the neighborhood of a walking gait. Though virtual potential energy does not always converge to 0, it plays a role of stabilization against disturbance during walking.

IV. Simulation

A. Application to the compass gait biped

We apply the proposed algorithm in the previous section to the compass gait biped depicted in Fig. 2.

We proceed 250 steps of the learning procedure, which implies that the robot continues to walk 500 steps totally. The simulation is executed with the initial condition $(\theta_1^0, \theta_2^0, \dot{\theta}_1^0, \dot{\theta}_2^0) = (-0.4, 0.3, 1.5, -1.0)$. The design parameters of the cost function (12) are $(\gamma_{y_1}, \gamma_{y_2}, \gamma_{\bar{u}}, \gamma_{u_c}, \gamma_c, \Delta t) = (30, 30, 0.30, 1.0, 1.0, 3.0\times10^{-3})$. Parameters of the learning procedures are $K_{c(0)} = 100$, $\epsilon_{1(\cdot)} = \epsilon_{2(\cdot)} = 0.10$, $K_{1(\cdot)} = \mathrm{diag}(2.0 \times 10^{-2}, 1.0 \times 10^{-2})$ and $K_{2(\cdot)} = 5.0 \times 10^{-2}$.

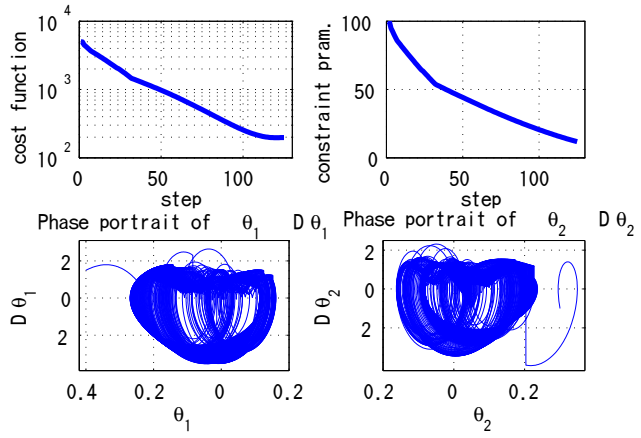

Fig. 5. History of cost function (upper left), that of constraint parameter K_c (upper right) and phase portrait (bottom)

Fig. 5 shows the simulation results. The history of the cost function (12) along the iteration decreasing monotonically. It implies that the output trajectory and the strength of the virtual constraint converges to an optimal ones smoothly. The figure shows the phase portrait of $\theta - \dot{\theta}$. It exhibits that a limit cycle which implies a periodic motion is generated as the robot continues to walk.

B. Application to the compass gait biped with a torso

We observe that a limit cycle is generated in the previous section, but the walking speed does not designed in the case of the compass gait biped. In this section, the additional degree of freedom is installed as in Fig. 6. We consider the compass gait biped with a torso, which is a more general walking robot model. The generalized coordinate is defined as $q := (q_1, q_2, q_3)^{\mathrm{T}} = (\theta_1, \theta_2, \theta_3)^{\mathrm{T}}$, where θ_3 denotes the torso angle and the control input defined as $\bar{u} := (\bar{u}_1, \bar{u}_2, \bar{u}_3)^{\mathrm{T}} = (u_1 - u_3, -u_2, u_2 + u_3)^{\mathrm{T}}$. Then, the dynamics of the robot is described by a Hamiltonian system (7) with the Hamiltonian (8). Here the inertia matrix and the potential energy are as follows

$$M(q) = \begin{pmatrix} m_T l^2 + ml^2 + ma^2 & -mbl\cos(q_1 - q_2) & m_T cl\cos(q_1 - q_3) \\ -mbl\cos(q_1 - q_2) & mb^2 & 0 \\ m_T cl\cos(q_1 - q_3) & 0 & m_T c^2 \end{pmatrix}$$

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

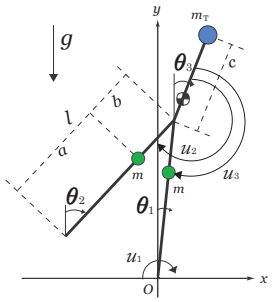

Fig. 6. The compass gait biped with a torso

The detail of the transition equation of the robot is omitted here. See [5].

The control objective is as follows. We control the legs by $\bar{u}_1$ and $\bar{u}_2$ as in the same manner of the compass gait biped. For the torso control, we use the following controller

$$\bar{u}_3 = -K_{31}q_3 - K_{32}(v_x^d - l\dot{q}_1\cos q_1), \qquad (24)$$

where K_{31} and K_{32} are appropriate positive constants and a design parameter v_x^d represents a desired horizontal velocity and $l\dot{q}_1\cos q_1$ denotes the velocity of the hip joint of the robot. In this method, one can design the desired velocity of the hip joint instead of the center of mass, because one can obtain the hip velocity only using information of the total leg length l, while the usage of the center of mass requires the precise knowledge of the robot parameter.

We proceed 240 steps of the learning procedure. The simulation is executed with the initial condition $(\theta_1^0, \theta_2^0, \theta_3^0, \dot{\theta}_1^0, \dot{\theta}_2^0, \dot{\theta}_3^0) = (-0.2, 0.2, 0.0, 1.5, -0.1, 0.0)$. The design parameters of the cost function (12) are $(\gamma_{y_1}, \gamma_{y_2}, \gamma_{\bar{u}}, \gamma_{u_c}, \gamma_c, \Delta t) = (3, 3, 0.30, 3.0 \times 10^{-3}, 5.0 \times 10^{-3}, 3.0 \times 10^{-2})$. Parameters of the learning procedures are $K_{c(0)} = 90$, $\epsilon_{1(\cdot)} = 1.0 \times 10^{-2}$, $\epsilon_{2(\cdot)} = 1.0 \times 10^{-3}$, $K_{1(\cdot)} = \mathrm{diag}(1.0 \times 10^{-2}, 1.0 \times 10^{-2})$, $K_{2(\cdot)} = 2.0 \times 10^{-2}$ and $v_x^d = 0.50\mathrm{m/s}$.

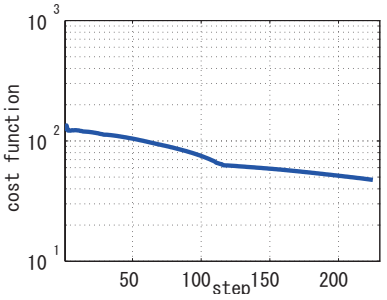

Fig. 7. History of cost function

Fig. 7 shows the cost function (12) decreasing monotonically along the iteration. It implies that the output trajectory converges to an optimal one smoothly. Fig. 8 implies that the strength of the virtual constraint is mitigated and Fig. 9 shows the sum of the norms of all inputs, i.e. the feedback input for potential energy, the feedforward inputs $\bar{u}_1, \bar{u}_2$ and

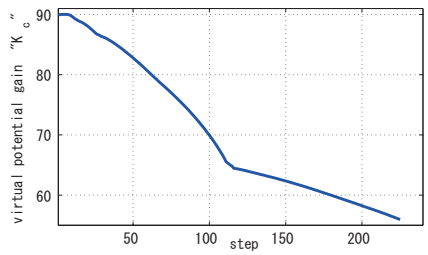

Fig. 8. History of virtual potential gain K_c

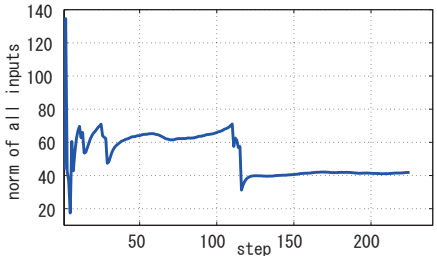

Fig. 9. The norm of all inputs

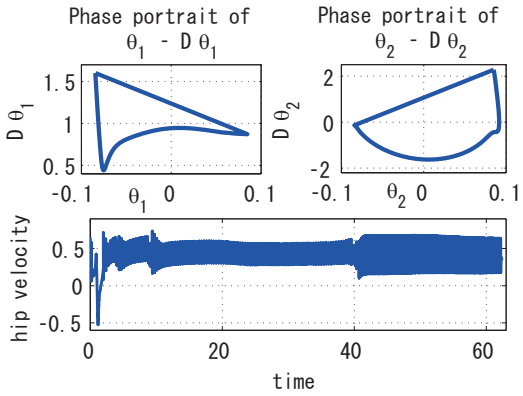

Fig. 10. Phase portrait and horizontal velocity ($v_x^d = 0.5$m/s)

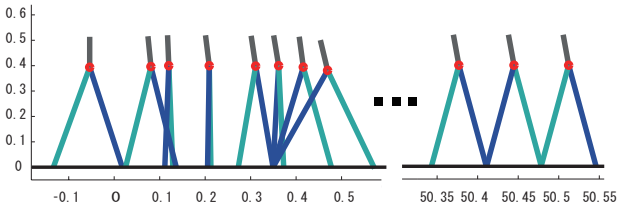

Fig. 11. Animation of the robot (from the first 5 steps to the last 3steps)

the feedback input $\bar{u}_3$ is also optimized. Fig. 10 shows that a limit cycle which implies a periodic motion is generated consequently and the robot achieves the desired horizontal velocity $v_x^d = 0.5$m/s. Finally, Fig. 11 shows the animation of the robot and it implies that the robot improves his walk as it continues to walk.

We observe that some limit cycles are generated by changing the desired velocities.

V. Conclusion

In this paper, we have proposed an optimal gait generation framework using virtual constraint and learning optimal control. The proposed method does not require the precise knowledge of the plant system. Due to the constraint, it does not need to repeat laboratory experiments under the same initial condition which is necessary for existing ILC frameworks. The proposed technique also differs from the past proposed ones using virtual constraint in that it automatically mitigates the strength of the constraint according to the progress of learning control. Finally, numerical simulations demonstrate the effectiveness of the proposed framework.

We try to apply the proposed method to more complex robots which have many degrees of freedom as a future work.

References

[1] T. McGeer, "Passive dynamic walking," *Int. J. Robotics Research*, vol. 9, no. 2, pp. 62–82, 1990.

[2] A. Goswami, B. Espiau, and A. Keramane, "Limit cycles in a passive compass gait biped and passivity-mimicking control laws," *Autonomous Robots*, vol. 4, no. 3, pp. 273–286, 1997.

[3] M. W. Spong, "Passivity-base control of the compass gait biped," in *Proc. of IFAC World Congress*, 1999, pp. 19–23.

[4] 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*, vol. 20, no. 3, pp. 565–573, 2004.

[5] J. W. Grizzle, G. Abba, and F. Plestan, "Asymptotically stable walking for biped robots: analysis via systems with impulse effects," *IEEE Trans. Autom. Contr.*, vol. 46, no. 1, pp. 51–64, 2001.

[6] S. Hyon and T. Emura, "Symmetric walking control: Invariance and global stability," in *Proc. IEEE ICRA*, 2005, pp. 1455–1462.

[7] 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*, 2006, pp. 5907–5912.

[8] ——, "Biped gait generation via iterative learning control including discrete state transitions," 2008, to appear in *Proc. of IFAC World Congress*.

[9] K. Fujimoto and T. Sugie, "Iterative learning control of Hamiltonian systems: I/O based optimal control approach," *IEEE Trans. Autom. Contr.*, vol. 48, no. 10, pp. 1756–1761, 2003.

[10] K. Fujimoto and I. Koyama, "Iterative feedback tuning for Hamiltonian systems," 2008, to appear in *Proc. of IFAC World Congress*.

[11] K. Fujimoto and T. Sugie, "Canonical transformation and stabilization of generalized Hamiltonian systems," *Systems & Control Letters*, vol. 42, no. 3, pp. 217–227, 2001.

[12] A. Goswami, B. Thuilot, and B. Espiau, "Compass-like biped robot part i: Stability and bifurcation of passive gaits," *INRIA Research Report*, no. 2996, 1996.