Biped Walking Control Using Offline and Online Optimization

Chenggang Liu, Jianbo Su
Department of Automation, Shanghai Jiao Tong University, Shanghai 200240
E-mail: cgliu2008@gmail.com

Abstract: The paper is aimed at an energetically efficient control method for biped walking. The walking cycle is composed of successive single-support phases and passive impacts. A parametric trajectory optimization method is implemented that finds symmetric, periodic, low torque walking gaits for a planar 5-link biped robot offline. The robot's configuration is then regulated by RHC (receding horizon control) with the same optimization criterion online. Only the geometric evolution of the robot's configuration is controlled, but not the temporal evolution. The effectiveness of the proposed method is evaluated using simulated walking control. The results show lower torques and more robustness from the proposed controller compared to a hand-tuned PD servo based walking controller.

Key Words: biped walking, gait optimization, receding horizon control

1 INTRODUCTION

The control of biped walking remains a difficult problem due to high dimensionality, nonlinearity, the intermittent contact between the feet and the ground, and constraints on kinematics and dynamics, such as joint limitations, the foot clearance requirement, and the foot-ground contact conditions [1].

By far, the most common approach to biped walking control is through tracking pre-computed reference trajectories. The most popular technique used to generate reference trajectories is based on the concept of ZMP (Zero Moment Point) [1]. Emphases are placed on enlarging the stability margin during gait planning [3, 4]. Another most popular technique is based on LIPM (Linear Inverted Pendulum Model) [5]. Because of the simplification of the whole dynamics as a linear system, modern control techniques can be used [6]. However, these techniques make use of a simplified dynamic model and do not take internal motions into account, the resultant walking control is not energetically efficient in general.

Walking control with optimal gaits is desirable for longer operation time and more human-like motions. The most frequently used approach for gait optimization is based on parametric optimization, in which the joint coordinates and/or the control variables are taken as optimization variables [2, 7] or they are approximated using an appropriate function approximation (e.g., polynomial approximation [8–11], Fourier extensions [12]). The coefficients of these functions are then treated as optimization variables. Then, the gait optimization problem can be solved by general nonlinear programming methods, such as sequential quadratic programming (SQP) [13].

With the context of tracking, many feedback control methods have been explored, such as PID controllers, computed torque and sliding model control [14]. Unfortunately, for an underactuated system, even high feedback gains sometimes cannot stabilize the system around a reference trajectory in the presence of external perturbations or modeling errors.

RHC (receding horizon control), also known as model predictive control, was first proposed for linear systems, and then applied to general nonlinear systems [15, 16]. With the improvement of computation speed of current controllers, RHC is now applicable to articulated robot control [17]. The use of RHC for biped walking control is motivated by its ability to handle constrained nonlinear systems. It is suitable to the problem of biped walking control, which is subject to joint limitations, the requirements of ground-foot contact, and disturbances. By introducing artificial constraints such as those on the body's posture and the swing limb trajectory, RHC was used to generate walking gaits without reference trajectories [18]. In [19], RHC using a simplified model was used to generate the reference ZMP trajectory online. However, RHC using a simplified model cannot generate walking gaits with the same efficiency as those generated by gait optimization using a full dynamic model and with an infinite time horizon.

Long-term stability and efficiency of walking gaits can be achieved by offline optimization with an infinite time horizon, while short-term stability and efficiency of gait control can be achieved by online optimization with a finite time horizon. In this paper, we propose an implementation of gait optimization with an infinite time horizon using a nonlinear parametric optimization method. The finite set of unknowns consists of the duration of the single-support phase, the joint coordinate values and the control values at uniformly distributed time intervals during the motion. In order to make these optimal gaits stable walking cycles, we take advantage of receding horizon control using a full dynamic model.

This article is organized as following: in Section 2, the robot model for our simulation study is proposed. In Section 3, a parametric optimization method for gait optimization with an infinite time horizon is proposed, which provides desired geometric evolutions of the robot's configuration for specified walking speeds. Gait control with receding horizon control using a full dynamic model is then proposed, which regulates the geometric evolution of the robot's configuration and makes an optimal gait a stable walking cycle. In Section 4, simulation and comparison with a hand-tuned PD servo controller results are presented, which demonstrate the validity, robustness, and performance of the proposed method. Conclusions and future work are discussed in Section 5.

This work is supported by National Nature Science Foundation Key Project under Grant 60935001.

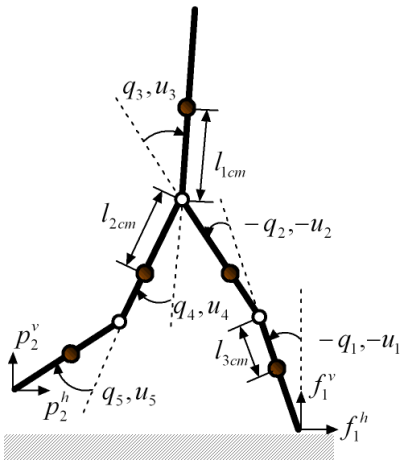


Fig. 1: Simplified structure of the planar 5-link biped robot used for our study.

Tab. 1: Physical parameters of the simulated robot

	calf	thigh	torso
mass $[kg]$	6.90	5.68	50.00
inertia $[kg \cdot m^2]$	0.15	0.10	1.50
length $[m]$	0.38	0.39	0.80
l_{cm}[m]	0.24	0.19	0.29

2 Robot Model and Model Assumptions

As shown in Fig. 1, a planar 5-link biped robot is used in our current study. The robot has a torso and two identical legs with knees. Although we do not model the feet, the ankle joint of stance leg can apply torques. A planar biped robot is used currently, but the proposed method can be easily extended to a 3-D biped robot. Kinematic and dynamic parameters of the simulated robot are according to a real hydraulic humanoid robot and listed in Table 1. The hydraulic humanoid robot allows us to control its joint torques directly. As in [14], we do not consider the double-support phase, either. Therefore, all walking cycles consist of successive phases of single support and passive impacts, while the robot progresses from left to right.

The walking cycle has two mathematical models: ordinary differential equations, describing the dynamics during the single-support phase, and an impact model, describing the instantaneous change in the velocities during impact. The biped model is then hybrid in nature, consisting of a continuous dynamics and a discrete impact effect [14].

2.1 Single Support Model

In the single-support phase, a biped robot is modeled as a rotational joint open-chain manipulator with five links. The dynamic model of the robot during this phase has five degrees of freedom. Let $\mathbf{q} = (q_1, \ldots, q_5)^T$ be the generalized coordinates describing the configuration of the robot depicted in Fig. 1. Since only symmetric gaits are considered, the same dynamic model is used no matter which leg is the stance leg, and the coordinates are relabeled after impact. The dynamics equations can be derived using the method of Lagrange. The result is a standard second order system

$$\mathbf{M}(\mathbf{q})\ddot{\mathbf{q}} + \mathbf{h}(\mathbf{q}, \dot{\mathbf{q}}) = \mathbf{u} \quad 0 < t < T \tag{1}$$

where $\mathbf{M}(\mathbf{q}) \in \mathbb{R}^{5\times5}$ is the inertia matrix, $\mathbf{h}(\mathbf{q}, \dot{\mathbf{q}}) \in \mathbb{R}^5$ is the vector of centrifugal, Coriolis, and gravity forces, $\mathbf{u} = (u_1, \ldots, u_5)^T$ is the actuated torque vector, and T is the duration of single-support phase. The second order system of (1) can be written in state space form by defining

$$\begin{aligned} \dot{\mathbf{x}} &= \begin{bmatrix} \dot{\mathbf{q}} \\ \mathbf{M}^{-1}(\mathbf{q})\big(-\mathbf{h}(\mathbf{q}, \dot{\mathbf{q}}) + \mathbf{u}\big) \end{bmatrix} \\ &= f(\mathbf{x}) + g(\mathbf{x})\mathbf{u} \quad 0 < t < T \end{aligned} \tag{2}$$

where $\mathbf{x} = (\mathbf{q}^T, \dot{\mathbf{q}}^T)^T$.

2.2 Impact Model

The impact between the swing leg and the ground is modeled as a contact between two rigid bodies. The assumptions are

1. The revolute joints connecting each two links will be assumed to be ideal, that is, perfect elastic and no mechanical tolerance [20].

2. The impact is instantaneous. The impulsive forces due to the impact may result in an instantaneous change in the velocities, but there is no instantaneous change in the positions.

3. Centripetal torques are assumed to be smaller than the impulsive external forces and are neglected.

4. The contact of the swing leg end with the ground results in no rebound and no slipping of the swing leg, and the stance leg lifting from the ground without interaction.

The impact model results in a smooth map [14]:

$$\mathbf{x}^+ = \Delta(\mathbf{x}^-), \tag{3}$$

where $\mathbf{x}^- = \lim_{\tau \to T^-} \mathbf{x}(\tau)$ is the value of the state 'just prior impact' at time T^- and $\mathbf{x}^+ = \lim_{\tau \to T^+} \mathbf{x}(\tau)$ is the value of the state 'just after impact' at time T^+. Function $\Delta(\cdot)$ is

$$\Delta(\mathbf{x}^-) = \begin{bmatrix} \mathbf{R} & 0 \\ 0 & \mathbf{R} \end{bmatrix} \begin{bmatrix} \mathbf{q}^- \\ \Delta_{\dot{\mathbf{q}}}(\mathbf{q}^-)\dot{\mathbf{q}}^- \end{bmatrix}, \tag{4}$$

where $\Delta_{\dot{\mathbf{q}}}(\mathbf{q})$ is a 5×5 matrix of smooth function of $\mathbf{q}$ and R is a constant matrix such that $R\mathbf{q}$ accounts for relabeling of the robot's coordinates which makes the swing leg become the stance leg.

2.3 Hybrid Dynamic Model

The overall biped robot dynamics can be expressed by a hybrid dynamic system consisting of ordinary differential equations and a discrete map:

$$\dot{\mathbf{x}} = f(\mathbf{x}) + g(\mathbf{x})\mathbf{u} \quad \mathbf{x}^- \notin \mathcal{S} \tag{5}$$

$$\mathbf{x}^+ = \Delta(\mathbf{x}^-) \quad \mathbf{x}^- \in \mathcal{S} \tag{6}$$

where $\mathcal{S} = \{\mathbf{x} | p_2^v(\mathbf{x}) = 0, \dot{p}_2^v(\mathbf{x}) < 0\}$ (see Fig. 1 for the definition of p_2^v). As shown in Fig. 2, the robot mechanics system evolves according to the ordinary differential equations (5) until the state in set S when impact occurs. The

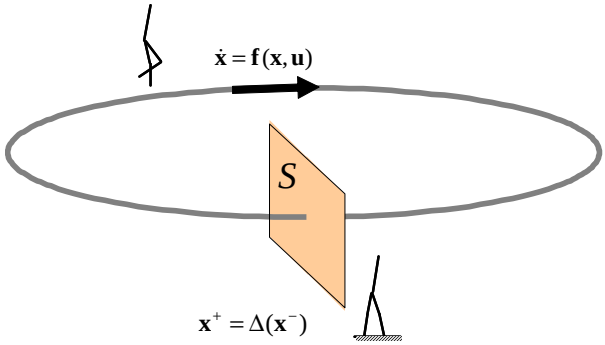


Fig. 2: A hybrid dynamic system for biped walking

impact with the ground results a rapid change in the velocity components of the state. The impact model also relabels the state components. The ultimate result of the impact model (6) is a new initial state from which the robot model evolves until the next occurrence of impact.

3 Walking Control with Optimal Gaits

3.1 Gait Optimization with An Infinite Time Horizon

The cost functional of one walking cycle is defined as

$$J'(\mathbf{x}(\cdot), \mathbf{u}(\cdot)) = \int_{0^+}^{T^-} \epsilon^t L\big(\mathbf{x}(t), \mathbf{u}(t)\big) \mathrm{d}t \tag{7}$$

where $\epsilon \in (0, 1)$ is a discount factor and L is Lagrangian, which will be defined later. For symmetric and periodic walking gaits, the following equations should hold

$$\begin{aligned} \mathbf{x}(T^+) &= \mathbf{x}(0^+) && (8)\\ \mathbf{x}(t+T) &= \mathbf{x}(t) && (9)\\ \mathbf{u}(t+T) &= \mathbf{u}(t) && (10)\\ \forall t > 0 \quad &\text{and} \quad t \neq kT \quad k = 1, \ldots, \infty. \end{aligned}$$

Thus the cost functional of an infinite time horizon

$$\begin{aligned} J_1 &= \int_{0^+}^{T^-} \epsilon^t L(\mathbf{x}, \mathbf{u}) \mathrm{d}t + \int_{T^+}^{2T^-} \epsilon^t L(\mathbf{x}, \mathbf{u}) \mathrm{d}t + \ldots \\ &= (1 + \epsilon^T + \epsilon^{2T} + \ldots) J' \\ &= \frac{1}{1-\epsilon^T} \int_{0^+}^{T^-} \epsilon^t L\big(\mathbf{x}(t), \mathbf{u}(t)\big) \mathrm{d}t. \end{aligned} \tag{11}$$

Gait optimization is to find the duration of single-support phase, T, the state trajectory, $\mathbf{x}(t)$, and the corresponding control trajectory, $\mathbf{u}(t)$, for $0 < t < T$ such that the cost functional J_1 is minimized, subject to equations of motion, joint limitations, and periodic conditions.

This problem can be cast as a nonlinear optimization problem by discretizing the time interval $(0, T)$ into N small time intervals and writing discrete approximations for the derivatives that appear in the robot model. By treating the state and the control variables at discrete-time intervals as optimization parameters, the continuous-time optimal control problem can be transformed to a nonlinear programming problem. Letting $\mathbf{q}(j)$ denote the value of $\mathbf{q}$ at time $j(T/N), j = 0, 1, \ldots, N$, we define the discrete approximations to the first-order derivatives at the midpoint of each time interval as follows:

$$\dot{\mathbf{q}}(j - 0.5) = \frac{\mathbf{q}(j) - \mathbf{q}(j-1)}{(T/N)} \quad j = 0, \ldots, N. \tag{12}$$

Therefore, the discrete approximations to the first-order and the second-order derivatives of $\mathbf{q}$ at time interval $j = 0, \ldots, N-1$ are given by midpoint discretization [21] as

$$\dot{\mathbf{q}}(j) = \frac{\dot{\mathbf{q}}(j - 0.5) + \dot{\mathbf{q}}(j + 0.5)}{2} \tag{13}$$

and

$$\ddot{\mathbf{q}}(j) = \frac{\dot{\mathbf{q}}(j + 0.5) - \dot{\mathbf{q}}(j - 0.5)}{(T/N)}. \tag{14}$$

The cost functional J_1 now becomes a cost function in form of

$$J_1 = \frac{1}{1-\epsilon^T} \sum_{j=0}^{N-1} \epsilon^{j(T/N)} L\big(\mathbf{x}(j), \mathbf{u}(j)\big)(T/N), \tag{15}$$

where $\mathbf{x}(j) = (\mathbf{q}(j)^T, \dot{\mathbf{q}}(j)^T)^T$ and $\mathbf{u}(j)$ is the value of $\mathbf{u}$ at time $j(T/N)$.

In order to generate an efficient gait of a specified walking speed, L is chosen as

$$L(\mathbf{x}, \mathbf{u}) = w_u \sum_{i=1}^{5} u_i^2 + w_v (v(\mathbf{x}) - v_d)^2 + w_r f_1^h(\mathbf{x})^2, \tag{16}$$

where w_u, w_v, and w_r are weighted factors, v_d and $v(\mathbf{x})$ are respectively the desired walking speed and the horizontal velocity of the hip, $f_1^h(\mathbf{x})$ is the horizontal component force of the ground reaction forces (as depicted in Fig. 1), which can be calculated by the inverse dynamics of the robot. The last term on the right-hand side of (16) is for reducing f_1^h to satisfy the limitations of the friction between the stance foot and the ground.

In summary, gait optimization for a specified walking speed is to find

$$\min_{\mathbf{q}(\cdot), \mathbf{u}(\cdot), T} J_1(\mathbf{q}(\cdot), \mathbf{u}(\cdot), T) \tag{17}$$

subject to

$$\begin{aligned} \ddot{\mathbf{q}}(j) &= \mathbf{M}^{-1}\big(\mathbf{q}(j)\big)\Big(- \mathbf{h}\big(\mathbf{q}(j), \dot{\mathbf{q}}(j)\big) + \mathbf{u}(j)\Big) \\ &\quad \forall j \in [0, N) && (18)\\ T &> 0 && (19)\\ \mathbf{q}(N) &= \mathbf{q}_f && (20)\\ \mathbf{x}(0) &= \Delta(\mathbf{x}(N)) && (21)\\ \mathbf{x}(j) &\in \mathcal{X} \quad \forall j = [0, N] && (22)\\ \mathbf{u}(j) &\in \mathcal{U} \quad \forall j = [0, N) && (23)\\ p_2^v(N/2) &> 0.05, && (24) \end{aligned}$$

with N, ϵ, $\mathbf{q}_f$ and v_d are specified. $\mathcal{X}$ is the set of feasible state and $\mathcal{U}$ is the set of feasible control. $\mathcal{X}$ and $\mathcal{U}$ are given by

$$\begin{aligned} \mathcal{X} &= \{\mathbf{x} \in \mathbb{R}^{10} | \mathbf{x}_{min} \leq \mathbf{x} \leq \mathbf{x}_{max}\} && (25)\\ \mathcal{U} &= \{\mathbf{u} \in \mathbb{R}^{5} | \mathbf{u}_{min} \leq \mathbf{u} \leq \mathbf{u}_{max}\}. && (26) \end{aligned}$$

The foot clearance constraint of (24) is to keep the swing foot high enough from touching the ground. We solve this optimization problem by an optimization software package, SNOPT [13].

3.2 Gait Control with Receding Horizon Control

Receding horizon control (RHC) is used to stabilize the robot and regulate the geometric evolution of the robot's configuration. The cost functional of RHC is in the form of

$$J_2 = \phi\big(\bar{\mathbf{x}}(T_c)\big) + \int_0^{T_c} \epsilon^{\tau} L\big(\bar{\mathbf{x}}(\tau), \bar{\mathbf{u}}(\tau)\big)\mathrm{d}\tau, \qquad (27)$$

where $\phi(\cdot)$ is the endpoint cost function, $\bar{\mathbf{x}}$ and $\bar{\mathbf{u}}$ are the state and the control vectors on τ axis, T_c is the time horizon of RHC, and L is the same as (16). Let $\mathbf{x}^o(t')$ denote the state of an optimal gait, which is the closest state to current state $\mathbf{x}(t)$ according to the distance, $\big(p^h_{hip}(\mathbf{x}) - p^h_{hip}(\mathbf{x}^o)\big)^2$ (p^h_{hip} is the horizontal position of the hip). Then the endpoint cost function $\phi(\cdot)$ is chosen as

$$\phi(\bar{\mathbf{x}}(T_c)) = \Delta\mathbf{x}(T_c)^T S_f \Delta\mathbf{x}(T_c) \qquad (28)$$

where

$$\Delta\mathbf{x}(T_c) = \bar{\mathbf{x}}(T_c) - \mathbf{x}^o(t' + T_c), \qquad (29)$$

$\mathbf{x}^o(t' + T_c)$ is the state of the optimal gait at time $t' + T_c$, and S_f is a diagonal weighted matrix.

RHC needs to find the control trajectory, $\bar{\mathbf{u}}(\tau)$, for $0 \le \tau \le T_c$ such that J_2 is minimized, subject to equations of motion and joint limitations. As in Section 3.1, this problem is also cast as a nonlinear optimization problem by discretizing the time interval $[0, T_c]$ into h small time intervals. Let $\bar{\mathbf{q}}(j)$ and $\bar{\mathbf{u}}(j)$ denote the values of $\mathbf{q}$ and $\mathbf{u}$ on τ axis at time $j(T_c/h), j = 0, 1, \ldots, h$. Then the discrete approximations to the first-order and the second-order derivatives are given by midpoint discretization as (13) and (14), where $\mathbf{q}$ is now replaced with $\bar{\mathbf{q}}$. The cost functional is now a cost function

$$J_2 = \phi(\bar{\mathbf{x}}(h)) + \sum_{j=0}^{h-1} \epsilon^{j(T_c/h)} L(\bar{\mathbf{x}}(j), \bar{\mathbf{u}}(j))(T_c/h), \qquad (30)$$

where $\bar{\mathbf{x}}(j) = (\bar{\mathbf{q}}^T(j), \dot{\bar{\mathbf{q}}}(j)^T)^T$. At each time step of simulation, find

$$\min_{\bar{\mathbf{q}}(\cdot), \bar{\mathbf{u}}(\cdot)} J_2\big(\mathbf{x}(t), \bar{\mathbf{q}}(\cdot), \bar{\mathbf{u}}(\cdot)\big) \qquad (31)$$

subject to

$$\bar{\mathbf{x}}(0) = \mathbf{x}(t) \qquad (32)$$

and (18), (22), and (23) where $\mathbf{x}$ and $\mathbf{u}$ are respectively replaced with $\bar{\mathbf{x}}$ and $\bar{\mathbf{u}}$, N with h, T with T_c and with h and T_c are specified. Only the first step of the resultant control sequence, $\bar{\mathbf{u}}(0)$, is applied as the actuated torques.

4 Simulation Results

We use the proposed gait optimization method to generate the optimal gait at walking speed $v_d = 0.5$ m/s. The parameters used by gait optimization are listed in Table 2. The stick diagram of one walking cycle and the actuated torques versus time of the optimal gait are shown in Fig. 3. This optimal gait was used by the following simulations.

For RHC, T_c is chosen as 10 ms, h as 1, S_f as $\mathrm{diag}(1, 1, 1, 1, 1, 10^{-3}, 10^{-3}, 10^{-3}, 10^{-3}, 10^{-3})$. $\mathbf{u}_{max}$ and $\mathbf{u}_{min}$ are the same as listed in Table 2, except for the range of actuated ankle torque u_1, which is $[-50, 50]$. The use of ankle torque provides additional robustness of gait control.

Tab. 2: Parameters used by gait optimization

w_v	10^2	$\mathbf{q}_f$	$[0.32, 0.25, 0.09, -0.32, -0.25]^T$
w_r	10^{-6}	$\mathbf{x}_{min}$	$[-0.8, -0.8, -0.8, -0.8, 0.0, -5.0, -5.0, -5.0, -5.0, -5.0]^T$
w_u	10^{-4}	$\mathbf{x}_{max}$	$[0.8, 0.0, 0.8, 0.8, 0.8, 5.0, 5.0, 5.0, 5.0, 5.0]^T$
ϵ	e^{-1}	$\mathbf{u}_{min}$	$-[0.0, 500, 500, 500, 500]^T$
N	50	$\mathbf{u}_{max}$	$[0.0, 500, 500, 500, 500]^T$

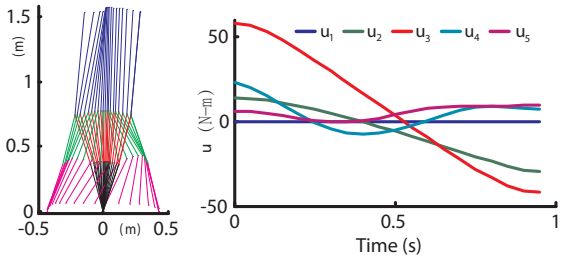


Fig. 3: The optimal gait at 0.5 m/s walking speed. a) The stick diagram. The configuration of the robot is drawn at every 50 ms. b) The actuated torques versus time.

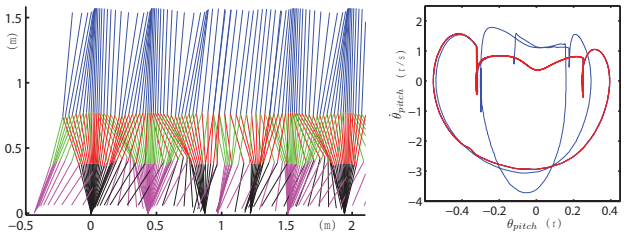
In the following simulations, pitch angle θ_{pitch} and pitch velocity $\dot{\theta}_{pitch}$ of the left thigh are calculated by

$$\theta_{pitch} = \begin{cases} q_1 + q_2 & \text{if the left leg is the stance leg} \\ q_1 + q_2 + q_3 + q_4 & \text{otherwise} \end{cases} \qquad (33)$$

and $\dot{\theta}_{pitch} = \frac{\mathrm{d}}{\mathrm{d}t}\theta_{pitch}$. The horizontal position of the center of pressure (CoP) with respect to the ankle joint of stance leg, x_{cop}, is calculated by $x_{cop} = -u_1/f_1^v$, where f_1^v is the vertical component force of the ground reaction forces (as depicted in Fig. 1).

4.1 Response to a Perturbation With a Perfect Model

The proposed controller was evaluated under a perturbation, which was a horizontal force (4000 Newtons) applied for 0.01 seconds at 2.21 seconds or 40 Newton-seconds impulse at the hip. The state of the robot was initialized on the optimal gait. A perfect model was used, that is, the same physical parameters were used by gait optimization, RHC, and the simulated robot. The stick diagram and the phase portrait of the left thigh's pitch motion are shown in Fig. 4. Convergence to the optimal gait is obtained after the perturbation. Since the proposed method regulates the internal state of the robot instead of tracking the trajectories of the optimal gait, the foot placement (as shown in Fig. 4(a)) and the walking speed (as shown in Fig. 5) are changed under perturbations. The walking speed is close to the desired walking speed (0.5 m/s) after the convergence. Also, as a result of limiting the actuated ankle torque by RHC, x_{cop} keeps inside the region, $[-0.1, 0.1]$. This means the stance foot will rest on the ground and the robot will walk stably if the stance foot can cover this region. After the convergence to the optimal gait, x_{cop} tends to zero because of zero actuated ankle torque of the optimal gait.



(a) (b)

Fig. 4: Response to 40 Newton-seconds perturbation with a perfect model. a) The stick diagram. The configuration of the robot is drawn at every 50 ms. b) Phase portrait of the left thigh's pitch motion. It traverses clockwise.

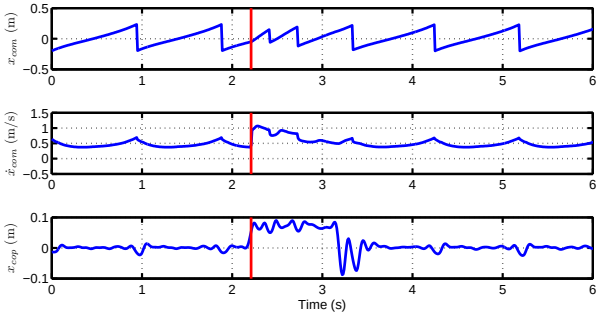


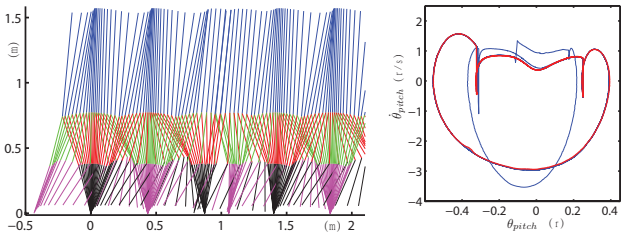
Fig. 5: Response to 40 Newton-seconds perturbation with a perfect model. The red lines show the time when the perturbation happens. a) The horizontal position of the hip versus time. b) The horizontal velocity of the hip versus time. c) The horizontal position of CoP with respect to the ankle joint of the stance leg versus time.

4.2 Response to a Perturbation With Modeling Errors

In practice, the robot's parameters are not perfectly known. We assume that we have some errors on the mass and the inertia of the torso. The proposed controller was evaluated under the same perturbation (40 Newton-seconds impulse at the hip) as before but with imprecise model data, that is, the physical parameters listed in Table 1 were used by gait optimization and RHC, while 50% more mass and inertia by the simulated robot. The state of the robot was also initialized on the optimal gait. As shown in Figs. 6 and 7, convergence to the optimal gait is obtained. The walking speed is close to the desired walking speed (0.5 m/s) after the convergence. Because of the existence of modeling errors, x_{cop} does not tend to zero even after the convergence.

4.3 Starting Step Control

The robustness of the proposed controller was further demonstrated by starting step control. The robot was initialized with a zero state or in a static up right posture. As shown in Fig. 8, it starts to walk and then walks with the optimal gait after applying the proposed controller without any changes. It is not necessary to have a special control for gait initialization as usual [4, 8]. This result shows the robustness of the proposed controller in that it can converge to the optimal gait starting from the origin of the state space.



(a) (b)

Fig. 6: Response to 40 Newton-seconds perturbation with modeling errors. a) The stick diagram. The configuration of the robot is drawn at every 50 ms. b) Phase portrait of the left thigh's pitch motion. It traverses clockwise.

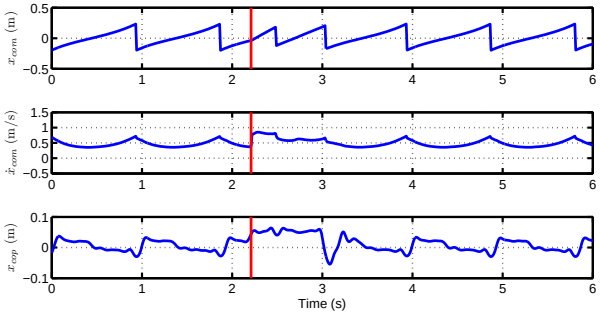
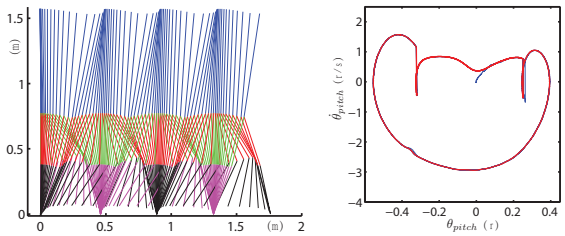


Fig. 7: Response to 40 Newton-seconds perturbation with modeling errors. The red lines show the time when the perturbation happens. a) The horizontal position of the hip versus time. b) The horizontal velocity of the hip versus time. c) The horizontal position of CoP with respect to the ankle joint of the stance leg versus time.

4.4 Comparison with a PD Servo Controller

We compared the proposed controller with a PD servo controller, in which each joint has a stiff PD controller to track the optimal gait. We use 1000 as the proportional gains and 10 as the derivative gains for all joints. The PD servo controller falls down after a impulsive perturbation of lager than 17 Newton-seconds or with a mass/inertia error of the torso larger then $+20\%$. In contrast, the proposed controller is able to handle an impulsive perturbation of up to 40 Newton-seconds and a mass/inertia error of the torso up to $+50\%$. We measured the sum of the squared torques, $\sum \mathbf{u}^{\mathsf{T}}\mathbf{u}T$, over 6 seconds starting in a state on the optimal gait. For walking in the presence of an impulsive perturbation of 17



(a) (b)

Fig. 8: Starting step control by the proposed method. a) The stick diagram. The configuration of the robot is drawn at every 50 ms. b) Phase portrait of the left thigh's pitch motion. It traverses clockwise.

Newton-seconds, the cost for the proposed controller was 9343.23, the corresponding cost for the PD servo controller was 19360.6. For walking with a $+20\%$ mass/inertia error of the torso, the cost for the former was 7638.78, compared to 8313.02 for the later.

5 Conclusions and Future work

In this paper, we proposed an efficient control method for planar biped walking. The combination of offline gait optimization and online optimization of RHC balances the requirements of efficiency, control robustness, and real-time computation. It shows that RHC's robustness and ability to handle constraints results in a large stable region of state space around these optimal gaits. Since the proposed control method regulates the internal state of the robot instead of tracking the optimal gaits, it may change foot placement and walking speed under perturbations, which results in more robust gait control. On a workstation with Intel(R) Xeon(TM) 3.20GHz dual-core CPU and 2G memory, it takes about 10 ms to solve the optimization problem in RHC without further code optimization. Therefore, the proposed controller is applicable to real-time applications.

In our future work, double-support phases will be addressed, which may help to reduce impulsive forces and increase robustness and efficiency of gait control. If the endpoint cost function of RHC returns the future cost, then RHC becomes trajectory optimization with an infinite horizon. We are exploring methods to approximate the future cost. Furthermore, we would like to extend our method to a full 3D humanoid robot model and implement this algorithm on real robots.

REFERENCES

[1] M. Vukobratovic and B. Borovac, Zero-moment point thirty five years of its life[J], *International Journal of Humanoid Robotics*, 2004, 1(1):157–173.

[2] M. Hardt, K. Kreutz-Delgado, and J. Helton, Optimal biped walking with a complete dynamical model[C]// *Proceedings of 38th IEEE Conference on Decision and Control*. 1999: 2999–3004.

[3] K. Hirai, M. Hirose, Y. Haikawa, and T. Takenaka, The development of Honda humanoid robot[C]// *Proceedings of IEEE International Conference on Robotics and Automation*. 1998:1321–1326.

[4] Q. Huang, K. Yokoi, S. Kajita, K. Kaneko, H. Arai, N. Koyachi, and K. Tanie, Planning walking patterns for a biped robot[J]. *IEEE Transactions on Robotics and Automation*, 2001, 17(3):280–289.

[5] S. Kajita and K. Tani, Experimental study of biped dynamic walking[J]. *IEEE Control Systems Magazine*, 1996, 16(1):13–19.

[6] S. Kajita, O. Matsumoto, and M. Saigo, Real-time 3D walking pattern generation for a biped robot with telescopic legs[C]// *Proceedings of IEEE International Conference on Robotics and Automation*. 2001: 2299–2306.

[7] L. Roussel, C. Canudas, and A. Goswami, Generation of energy optimal complete gait cycles for biped robots[C]// *Proceedings of IEEE International Conference on Robotics and Automation*. 1998: 2036–2041.

[8] D. Djoudi, C. Chevallereau, and Y. Aoustin, Optimal reference motions for walking of a biped robot[C]// *Proceedings of IEEE International Conference on Robotics and Automation*. 2005: 2002–2007.

[9] G. Bessonnet, P. Seguin, and P. Sardain, A parametric optimization approach to walking pattern synthesis[J]. *International Journal of Robotics Research*, 2005, 24(7): 523–536.

[10] T. Saidouni and G. Bessonnet, Generating globally optimised sagittal gait cycles of a biped robot[J]. *Robotica*, 2003, 21(2): 199–210.

[11] C. Chevallereau and Y. Aoustin, Optimal reference trajectories for walking and running of a biped robot[J]. *Robotica*, 2001, 19(5): 557–569.

[12] L. Roussel, C. Canudas, and A. Goswami, Comparative study of methods for energy-optimal gait generation for biped robots[C]// *Proceedings of International Conference on Informatics and Control*. 1997: 1205–1212.

[13] P. E. Gill, W. Murray, M. Saunders, SNOPT: an SQP algorithm for large-scale constrained optimization[J]. *SIAM Journal on Optimization*, 1997, 12: 979-1006.

[14] J. Grizzle, G. Abba, and F. Plestan, Asymptotically stable walking for biped robots: analysis via systems with impulse effects[J]. *IEEE Transactions on Automatic Control*, 2001, 46(1): 51–64.

[15] C. E. Garcia, D. M. Prett, and M. Morari, Model predictive control: theory and practice — a survey[J]. *Automatica*, 1989, 25: 335–348.

[16] D. Q. Mayne and H. Michalska, Receding horizon control of nonlinear systems[J]. *IEEE Transactions on Automatic Control*, 1990, 35(7): 814–824.

[17] D. Dimitrov, P.-B. Wieber, H. J. Ferreau, and M. Diehl, On the implementation of model predictive control for on-line walking pattern generation[C]// *Proceedings of IEEE International Conference on Robotics and Automation*. 2008: 2685–2690.

[18] C. Azevedo, P. Poignet, and B. Espiau, Moving horizon control for biped robots without reference trajectory[C]// *Proceedings of IEEE International Conference on Robotics and Automation*. 2002: 2762 –2767.

[19] H. Takeuchi, Real time optimization for robot control using receding horizon control with equal constraint[J]. *Journal of Robotic Systems*, 2003, 20(1): 3–13.

[20] M. Moore, J. Wilhelms, C. Graphics, I. S. Board, and S. Cruz, Collision detection and response for computer animation[J]. *Computer Graphics*, 1988, 22(4): 289–298.

[21] R. J. Vanderbei, Case studies in trajectory optimization: trains, planes, and other pastimes[J]. *Optimization and Engineering*, 2001, 2(2): 215–243.