2015 IEEE International Conference on Robotics and Automation (ICRA)
Washington State Convention Center
Seattle, Washington, May 26-30, 2015

# Adapting Human Motions to Humanoid Robots Through Time Warping Based on a General Motion Feasibility Index

Yu Zheng, Katsu Yamane

*Abstract*— **Having human-like motions will make humanoid robots more predictable and safer for the people around them. An effective way to realize this would be to use human motions as reference. Due to different kinematic and dynamic properties between humans and humanoid robots, however, a human motion could be physically infeasible for a robot and cause the robot to fall over. Therefore, it is necessary to modify and adapt an infeasible human motion to the robot. This paper presents a method for adapting human motions to humanoid robots based on a technique called time warping, which modifies the time line of a reference motion to speed up or slow down the motion. By doing this, the velocity and acceleration profiles of the motion are changed and it is possible to turn an infeasible motion into a feasible one. The optimal time warping is obtained through a generalized motion feasibility index that quantifies the feasibility of a motion considering the friction and center-of-pressure constraints. Thanks to the generality of the index, the proposed motion adaptation method can be applied to motions on arbitrary terrains or number of links in contact with the environment. Through dynamics simulation, we demonstrate that the method facilitates the reproduction of human motions on a humanoid robot.**

## I. Introduction

Humanoid robots are built to resemble humans and expected to assist and live with people in the future. Having human-like movements together with human-like appearances helps people infer their future motion and intention, which is of great importance for safe and smooth human-robot interactions. However, it is difficult to generate human-like motions through optimization or control because no clear criterion is known for coordinating all the joints such that the resulting motions look human-like. For example, many locomotion controllers for humanoid robots are based on simple models that cannot describe their full-body motion, so the motions usually do not look like a human [1], [2]. In computer graphics, some methods have successfully generated human-like walking and running motions on virtual characters [3], [4], but it is hard to extend them to stylized motions.

Since humanoid robot body is designed to resemble that of humans, mimicking human motions seems a natural and promising way to endow humanoid robots with human-like movements. Due to kinematic/dynamic differences between humans and robots, however, a human motion can be infeasible for a robot and cause the robot to fall over. To solve this issue, human motions are usually adapted to the robot's kinematics and dynamics. A potential issue with this approach is that the modified motion may not satisfy other constraints, such as footstep location in locomotion.

In this paper, we adapt a reference human motion to a humanoid robot by modifying the time line of the motion, called time warping, which implies that the robot will realize the same poses as in the reference motion at different moments. By doing this, the velocity and acceleration profiles of the motion are changed and the reference motion can be turned into a feasible motion for the robot. To ensure that the motion will not be disordered or prolonged by time warping, we choose a strictly increasing function that preserves the period of time of the motion to map the original time line to a new one. A general motion feasibility index is proposed to guide the choice of a time warping function such that the modified motion has a relatively larger feasibility margin for the robot. Thanks to the generality of the index, the proposed motion adaptation method can be applied to motions on arbitrary terrains or contact situations.

The rest of this paper is organized as follows. Section II reviews the related work. Section III goes over the dynamics of a humanoid robot. Section IV discusses the motion feasibility index, followed by the motion adaptation through time warping in Section V. Section VI shows simulation results. Conclusion and future work are given in Section VII.

## II. Related Work

As humanoid robots and humans have similar structures, programming humanoid robots by referring to human motions seems to be an effective way to let robots have natural human-like movements. Many efforts have been made to this end. Some methods calculate the joint angles of a robot to fit a human motion based purely on the kinematics of a robot [5], [6]. Some methods parameterize the joint trajectory of a robot and formulate an optimization problem to determine the values of parameters, respecting the fitting to the reference trajectory as well as robot's own dynamics and physical limitations [7], [8], [9]. Many methods modify the joint trajectory extracted from a human motion according to a replanned zero moment point (ZMP) or center of mass (CoM) trajectory, which is generated based on a simple model of the robot and ensures the equilibrium of the robot in the motion [10], [11], [12], [13]. Rather than offline planning, other methods can realize online motion tracking incorporated with balance control [14], [15], [16], [17], [18].

All the aforementioned methods modify the pose of the robot in a motion in order to make the motion feasible for the robot. In fact, the velocity and acceleration profiles of a

Yu Zheng is with the Department of Electrical and Computer Engineering, University of Michigan-Dearborn, MI 48128, USA. He was at Disney Research when the research was conducted. yuzheng001@gmail.com
Katsu Yamane is with Disney Research, Pittsburgh, PA 15213, USA. kyamane@disneyresearch.com

motion can be changed without changing the pose by scaling or warping the time line of the motion. By this means, it is possible to obtain a feasible motion in which the robot will realize the same pose as specified in the original reference motion. This idea was used for modifying the trajectory of manipulators subject to the joint torque limit [19].

In order to adapt human motions to humanoid robots, a criterion for quantifying motion feasibility is needed. The most widely used criterion is the Zero-Moment Point (ZMP) [20] that represents the center of the ground reaction force required to perform a given motion. The motion is feasible if the ZMP is always within the contact convex hull. Kanehiro et al. [21] used the ZMP criterion to determine the time progress to generate collision-free and dynamically feasible motions. Goswami [22] proposed the foot rotation indicator (FRI) that indicates the center of ground reaction force required to keep the foot stationary. However, they do not consider the friction constraint at contact and assume that all contacts occur on a flat surface. More general criteria based on feasible contact forces have been used in walking pattern generation [23], balance control [24], and motion tracking [25] of humanoid robots. In this paper, we present a general motion feasibility index that considers friction constraints and allows complex contact conditions and apply the index to human motion adaptation.

## III. Motion Tracking Control

To better understand the issue, we first review the dynamics and motion tracking of a humanoid robot in this section.

### A. Equation of Motion

We consider a humanoid robot with $N_J$ actuated joints and $N_G = N_J + 6$ degrees of freedom (DoF) including the six unactuated DoF of the floating base. Let $\boldsymbol{q} \in \mathbb{R}^{N_G}$ denote the generalized coordinates that define the configuration of the robot and the first six components of $\boldsymbol{q}$ correspond to the translation and rotation of the floating base of the robot. Let $N_C$ be the number of links in contact with the environment and $\boldsymbol{w}_i \in \mathbb{R}^6$ $(i = 1, 2, \ldots, N_C)$ the contact wrench (force and moment) applied to contact link $i$ by the environment.

The equation of motion of the robot with respect to the generalized coordinate space can be written as

$$\boldsymbol{M}\ddot{\boldsymbol{q}} + \boldsymbol{c} = \boldsymbol{N}^T\boldsymbol{\tau} + \boldsymbol{J}^T\boldsymbol{w} \tag{1}$$

where $\boldsymbol{M} \in \mathbb{R}^{N_G \times N_G}$ is the inertia matrix of the robot, $\boldsymbol{c} \in \mathbb{R}^{N_G}$ is the sum of Coriolis, centrifugal and gravity forces, and $\boldsymbol{w} = \begin{bmatrix} \boldsymbol{w}_1^T & \boldsymbol{w}_2^T & \ldots & \boldsymbol{w}_{N_C}^T \end{bmatrix}^T \in \mathbb{R}^{6N_C}$. Matrix $\boldsymbol{N} \in \mathbb{R}^{N_J \times N_G}$ maps the joint torques into the generalized forces. Since the floating base is unactuated, the first six rows of $\boldsymbol{N}^T$ are zero and $\boldsymbol{N} = [\boldsymbol{0}_{N_J \times 6} \quad \boldsymbol{I}_{N_J \times N_J}]$. Matrix $\boldsymbol{J} = \begin{bmatrix} \boldsymbol{J}_1^T & \boldsymbol{J}_2^T & \ldots & \boldsymbol{J}_{N_C}^T \end{bmatrix}^T \in \mathbb{R}^{6N_C \times N_G}$ is the contact Jacobian and its transpose maps the contact wrenches into the generalized forces, where $\boldsymbol{J}_i \in \mathbb{R}^{6 \times N_G}$ is the Jacobian of contact link $i$ with respect to the generalized coordinates.

Let $\boldsymbol{p}_{i,j} \in \mathbb{R}^3$ $(j = 1, 2, \ldots, N_i)$ be the vertices of the contact area between contact link $i$ and the environment, and $\boldsymbol{f}_{i,j} \in \mathbb{R}^3$ the contact force at vertex $j$ on contact link $i$, as depicted in Fig. 1. The number $N_i$ of contact vertices for every contact link can be 0, 1, 2, and 3 or more in the cases of no contact, point contact, edge contact, and face contact, respectively. We assume that all the contacts are rigid subject to Coulomb friction model. Therefore, $\boldsymbol{f}_{i,j}$ is a pure force and can be decomposed into three components $f_{i,j1}, f_{i,j2}, f_{i,j3}$ along the normal and two tangential vectors at the contact vertex, and it must satisfy the friction constraint as below to avoid undesired slippage at the contact link,

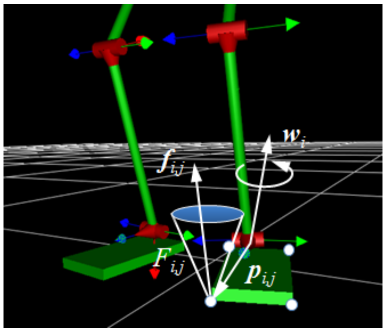


Fig. 1. Illustration of the relation between the wrench and the contact forces applied to a contact link by the environment.

$$F_{i,j} = \left\{ \boldsymbol{f}_{i,j} \in \mathbb{R}^3 \mid f_{i,j1} \geq 0, \ \sqrt{f_{i,j2}^2 + f_{i,j3}^2} \leq \mu_i f_{i,j1} \right\}. \tag{2}$$

The wrench $\boldsymbol{w}_i$ applied to contact link $i$ is the resultant force and moment from all contact forces $\boldsymbol{f}_{i,j}$ $(j = 1, 2, \ldots, N_i)$ and can be written as

$$\boldsymbol{w}_i = \sum_{j=1}^{N_i} \boldsymbol{R}_{i,j}\boldsymbol{f}_{i,j} = \boldsymbol{R}_i\boldsymbol{f}_i \tag{3}$$

where $\boldsymbol{R}_i = [\boldsymbol{R}_{i,1} \quad \boldsymbol{R}_{i,2} \quad \ldots \quad \boldsymbol{R}_{i,N_i}] \in \mathbb{R}^{6 \times 3N_i}$, $\boldsymbol{f}_i = \begin{bmatrix} \boldsymbol{f}_{i,1}^T & \boldsymbol{f}_{i,2}^T & \cdots & \boldsymbol{f}_{i,N_i}^T \end{bmatrix}^T \in \mathbb{R}^{3N_i}$, and $\boldsymbol{R}_{i,j} \in \mathbb{R}^{6 \times 3}$ is the matrix that maps $\boldsymbol{f}_{i,j}$ into the force and moment around the local frame of contact link $i$ in which $\boldsymbol{w}_i$ is expressed.

Expressing the contact wrench $\boldsymbol{w}_i$ as (3) with the contact forces $\boldsymbol{f}_{i,j}$ at the vertices of contact link $i$ has two advantages. First, it is guaranteed that the center of pressure of each contact link is within its contact area and the center of pressure of the robot is within the support convex hull. Second, the friction constraint can be explicitly considered.

### B. Inverse Dynamics

In order for a humanoid robot to realize a reference human motion, right joint torques need be determined for actuating the robot. To do this, the desired joint accelerations are first calculated based on the reference accelerations as well as the reference and current positions and velocities of joints:

$$\hat{\ddot{q}} = \ddot{\boldsymbol{q}}^{ref} + k_d(\dot{\boldsymbol{q}}^{ref} - \dot{\boldsymbol{q}}) + k_p(\boldsymbol{q}^{ref} - \boldsymbol{q}) \tag{4}$$

where $\boldsymbol{q}$, $\dot{\boldsymbol{q}}$ are the current joint angle and velocity, $\boldsymbol{q}^{ref}$, $\dot{\boldsymbol{q}}^{ref}$, $\ddot{\boldsymbol{q}}^{ref}$ are the reference joint angle, velocity, and acceleration, and $k_p$, $k_d$ are proportional and derivative gains, which can be different for different joints.

Once the desired joint accelerations are determined, joint torques and corresponding contact forces are computed based on (1) and (3) subject to the friction constraint (2). This problem is often known as inverse dynamics and can be reduced to a quadratic programming problem [15], [25], [18]. For more details, readers are referred to relevant references.

It is quite common that a reference motion is not always feasible for the robot. In other words, there may not exist feasible contact forces under the friction constraint to support the robot and enable the robot to realize the desired joint accelerations. In this case, the reference motion will not be exactly reproduced, and what motion the robot will generate depends on the property of the motion tracking controller and the extent of infeasibility of the reference motion. Sometimes a similar motion may be generated, but an unexpected motion may also occur and cause the robot to fall over. Therefore, it is necessary to modify the motion and make it feasible for the robot. To this end, we will first introduce a quantitative motion feasibility index and then a motion adaptation method based on the index in the following sections.

## IV. General Motion Feasibility Index

In this section, we present a general quantitative index of motion feasibility. The index not only tells whether or not a given motion is feasible for the robot but also quantifies how far the motion is from being infeasible/feasible.

### A. Definition

The first six equations in the full-body dynamics equation (1) describe the motion of the floating base and do not involve joint torques, which corresponds to the fact that the total linear and angular momenta are affected only by external forces. Extracting the first six equations of (1) yields

$$\boldsymbol{M}_1\ddot{\boldsymbol{q}} + \boldsymbol{c}_1 = \boldsymbol{J}_1^T\boldsymbol{w} \tag{5}$$

where $\boldsymbol{M}_1 \in \mathbb{R}^{6\times N_G}$ and $\boldsymbol{J}_1^T \in \mathbb{R}^{6\times 6N_C}$ consist of the first six rows of $\boldsymbol{M}$ and $\boldsymbol{J}^T$, respectively, and $\boldsymbol{c}_1$ comprises the first six components of $\boldsymbol{c}$. Substituting (3) into (5) yields

$$\boldsymbol{M}_1\ddot{\boldsymbol{q}} + \boldsymbol{c}_1 = \boldsymbol{G}\boldsymbol{f} \tag{6}$$

where $\boldsymbol{f} = \begin{bmatrix}\boldsymbol{f}_1^T & \boldsymbol{f}_2^T & \cdots & \boldsymbol{f}_{N_C}^T\end{bmatrix}^T \in \mathbb{R}^{3N_f}$ consists of the contact forces applied to all contact links and is called the total contact force, $\boldsymbol{G} = [\boldsymbol{G}_1 \;\; \boldsymbol{G}_2 \;\; \cdots \;\; \boldsymbol{G}_{N_C}] \in \mathbb{R}^{6\times 3N_f}$ with $\boldsymbol{G}_i = \boldsymbol{J}_{1i}^T\boldsymbol{R}_i \in \mathbb{R}^{6\times 3N_i}$ maps all contact forces to the wrench around the floating base and is called the total contact mapping, and $N_f = \sum_{i=1}^{N_C} N_i$ is the number of contact forces/points over all contact links.

Given a reference motion for the robot, the left-hand side of (6) can be calculated and it is the required wrench at the floating base for the robot to realize the reference motion. The right-hand side of (6) is the wrench that can be generated on the floating base by contact forces. Matrix $\boldsymbol{G}$ can be calculated based on the state and the position of every contact link specified in the reference motion. Then, whether the reference motion is feasible for the robot is equivalent to whether there are feasible contact forces $\boldsymbol{f}$, namely contact forces satisfying the friction constraint (2), to generate the required wrench and make (6) hold.

To define the motion feasibility index, we look closely at the geometric property of (6). The left side of (6) is a single point $\boldsymbol{w}$ in the 6-D wrench space. The friction constraint (2) defines a convex cone that restricts the contact force at every contact point, as depicted in Fig. 1. Through the linear mapping $\boldsymbol{G}$, the image of the convex cones from all contact points is a convex cone $V$ with apex at the origin in the 6-D wrench space, which consists of all wrenches that can be generated on the floating base with feasible contact forces. Then, the reference motion is feasible for the robot if and only if the single point $\boldsymbol{w}$ is contained in the convex cone $V$. If $\boldsymbol{w} \notin V$ or the motion is infeasible, we use the minimum Euclidean distance $d^+$ from $\boldsymbol{w}$ to $V$, defined as below, to measure how far the motion is from being feasible:

$$d^+ \triangleq \min_{\boldsymbol{v}\in V} \|\boldsymbol{w} - \boldsymbol{v}\| \tag{7}$$

where $\|\cdot\|$ denotes the Euclidean norm. If $\boldsymbol{w} \in V$ or the motion is feasible, we use the minimum Euclidean distance $d^-$ from $\boldsymbol{w}$ to the boundary of $V$ to measure how far the motion is from becoming infeasible:

$$d^- \triangleq \min_{\boldsymbol{v}\in \partial V} \|\boldsymbol{w} - \boldsymbol{v}\| \tag{8}$$

where $\partial V$ denotes the boundary of $V$. Finally, the motion feasibility index $d$ is taken to be $d^+$ if $\boldsymbol{w} \notin V$ or $-d^-$ if $\boldsymbol{w} \in V$. Note that $d$ is positive for an infeasible motion and non-positive for a feasible motion.

### B. Computation

Now we discuss the algorithms to compute $d^+$ and $d^-$. We first assume that the reference motion is infeasible and calculate $d^+$. If $d^+ = 0$, which implies that $\boldsymbol{w} \in V$ and the reference motion is feasible, then we continue to calculate $d^-$. In the following discussion, we denote by $V_k$ a finite set of vectors in $V$ and $\mathrm{CO}(V_k)$ the convex cone with apex at the origin formed by vectors in $V_k$ as its edges.

*1) Computation of $d^+$:* The value of $d^+$ can be calculated by the algorithm [26], which iteratively changes a simplicial cone in $V$ such that its minimum distance from the point $\boldsymbol{w}$ converges to $d^+$, as illustrated in Fig. 2a.

The algorithm [26] starts with any linearly independent set $V_k$ of vectors in $V$ and iterates as follows. Assume that $\boldsymbol{w}$ is not contained in $\mathrm{CO}(V_k)$. Let $d_k$ be the minimum distance from $\boldsymbol{w}$ to $\mathrm{CO}(V_k)$ and $\boldsymbol{v}_k$ the point on $\mathrm{CO}(V_k)$ closest to $\boldsymbol{w}$, which can be calculated by projecting $\boldsymbol{w}$ onto every face of $\mathrm{CO}(V_k)$. Let $\hat{V}_k$ be the minimal subset of $V_k$ forming a face of $\mathrm{CO}(V_k)$ that contains $\boldsymbol{v}_k$. If $\boldsymbol{v}_k \neq \boldsymbol{w}$, then the hyperplane with normal $\boldsymbol{w} - \boldsymbol{v}_k$ passing through $\boldsymbol{v}_k$ bounds $\mathrm{CO}(V_k)$ to one side and cuts the convex cone $V$, which allows us to find a vector on the boundary of $V$ that falls on the other side of the hyperplane. Adding the vector to $\hat{V}_k$, we obtain $V_{k+1}$ and can prove that the minimum distance $d_{k+1}$ from $\boldsymbol{w}$ to $\mathrm{CO}(V_{k+1})$ is strictly smaller than $d_k$. Therefore, $d_k$ is strictly decreasing along with the iteration and guaranteed to converge to $d^+$, which is the greatest lower bound on $d_k$.

If $\boldsymbol{w} \notin V$ or the reference motion is infeasible, then $d^+ > 0$ and the computing of the motion feasibility index $d$ stops; otherwise, $d^+ = 0$ and we continue to compute $d^-$.

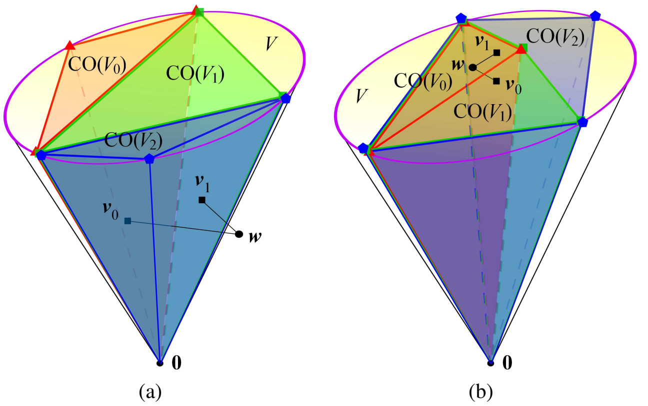


Fig. 2. Computation of the motion feasibility index. (a) Computing $d^+$: A simplicial cone $\mathrm{CO}(V_k)$ in the convex cone $V$ is changed iteratively such that its minimum distance from the point $\boldsymbol{w}$ converges to the minimum distance $d^+$ from $\boldsymbol{w}$ to $V$. (b) Computing $d^-$: A polyhedral cone $\mathrm{CO}(V_k)$ in the convex cone $V$ is expanded iteratively such that the minimum distance from the point $\boldsymbol{w}$ to its boundary converges to the minimum distance $d^-$ from $\boldsymbol{w}$ to the boundary of $V$.

*2) Computation of* $d^-$*:* If $\boldsymbol{w} \in V$ or the reference motion is feasible, then the algorithm [26] yields a finite set $V_k$ of vectors in $V$ such that $\boldsymbol{w}$ is contained in the polyhedral cone $\mathrm{CO}(V_k)$. We rewrite the set as $V_0$ here and assume that $V_0$ contains six linearly independent vectors such that the formed polyhedral cone is 6-D in the 6-D wrench space. In what follows, we present an algorithm to compute $d^-$ by iteratively adding vectors to $V_k$ and expanding $\mathrm{CO}(V_k)$ such that the minimum distance from $\boldsymbol{w}$ to the boundary of $\mathrm{CO}(V_k)$ converges to $d^-$, as illustrated in Fig. 2b.

Let $d_k$ be the minimum distance from $\boldsymbol{w}$ to the facets of $\mathrm{CO}(V_k)$ and $\boldsymbol{v}_k$ the orthogonal projection of $\boldsymbol{w}$ on the facet whose distance from $\boldsymbol{w}$ is $d_k$. Then, $d_k$ is the minimum distance from $\boldsymbol{w}$ to the boundary of $\mathrm{CO}(V_k)$. Moreover, the hyperplane with normal $\boldsymbol{v}_k - \boldsymbol{w}$ passing through $\boldsymbol{v}_k$ bounds $\mathrm{CO}(V_k)$ to one side. If the hyperplane cuts the convex cone $V$, then we can find a vector on the boundary of $V$ that is on the other side of the hyperplane. Adding such a vector to $V_k$ yields $V_{k+1}$, which forms a bigger polyhedral cone $\mathrm{CO}(V_{k+1})$ that contains facet $j_k$ of $\mathrm{CO}(V_k)$ in its interior. Then, by the iteration, $\mathrm{CO}(V_k)$ keeps expanding and $d_k$ is increasing. On the other hand, $d^-$ is the least upper bound on $d_k$, since $\mathrm{CO}(V_k)$ is always contained in $V$. Therefore, $d_k$ converges to $d^-$ as the algorithm iterates.

## V. Time Warping

In this section, we introduce a method for adapting a reference human motion to a humanoid robot based on the proposed motion feasibility index. Unlike most methods that change the pose of the robot in the motion, our method warps the time line of the motion and is called time warping. By doing this, the robot will need to realize the same pose as in the reference motion, which is particularly important for some locomotion tasks that do not allow the robot to deviate from the desired motion, such as following a preplanned footstep sequence or a collision-free trajectory to walk over a rough terrain or through obstacles.

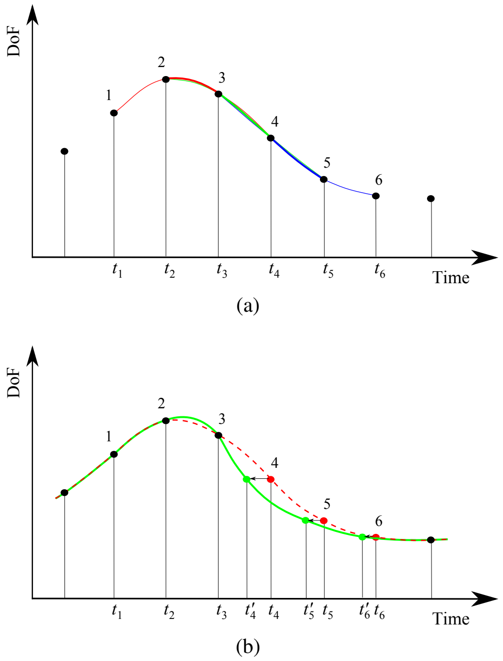


Fig. 3. Processing of reference motion. (a) Interpolation of key frames (black dots). To interpolate the motion trajectory between frames 2 and 3 (or 3 and 4 or 4 and 5), frames 1–4 (or 2–5 or 3–6) are used. (b) Time warping of frames 4–6. The motion trajectory is changed since frame 2.

### A. Basic Idea

To better understand how time warping works, we first introduce the constitution of a reference motion. A reference motion mainly consists of key motion frames for every DoF of the robot including the floating base and joints. It also contains the time of contact state change for every contact link. The time interval between two frames (typically $\frac{1}{30}$ second) is usually much longer than the control cycle (1 or 2 milliseconds). Then, the trajectory between two frames is determined through interpolation, as depicted in Fig. 3a, which involves not only the two but also several adjacent frames depending on the interpolation method.

Fig. 3b depicts the principle of time warping. Time warping is to alter the times of motion frames, which means that the robot will realize the same pose specified by a motion frame at a different time. Then, the motion trajectory over time, particularly the velocity and the acceleration, will be changed. By doing this, an infeasible reference motion can be turned into a feasible one for the robot.

### B. Canonical Time Warping Functions

We first define a canonical time warping function as $t' = f_0(t)$: $t \in [0, 1] \rightarrow t' \in [0, 1]$ with the following properties:

a) $f_0(0) = 0$;
b) $f_0(1) = 1$;
c) $f_0$ is strictly increasing in $[0, 1]$.

The first two properties ensure that the time warping function does not increase or decrease the motion period, while the third property ensures that time warping does not change the order of motion frames over time.

The canonical time warping function can be chosen at will as long as it possesses the above three properties. Here we write it as a combination of a set of basis functions

$$f_0(t) = \sum_{k=1}^{K} c_k f_k(t) \tag{9}$$

where $f_k(t)$'s are user-specified basis functions and $c_k$'s are unknown coefficients that need to be determined.

From (9), properties a) and b) can be expanded as

$$\begin{bmatrix} f_1(0) & f_2(0) & \cdots & f_K(0) \\ f_1(1) & f_2(1) & \cdots & f_K(1) \end{bmatrix} \boldsymbol{c} = \begin{bmatrix} 0 \\ 1 \end{bmatrix} \tag{10}$$

where $\boldsymbol{c} = [c_1 \quad c_2 \quad \cdots \quad c_K]^T$. Then, $\boldsymbol{c}$ can be written as

$$\boldsymbol{c} = \boldsymbol{c}_0 + \boldsymbol{N\lambda} \tag{11}$$

where $\boldsymbol{c}_0$ is a particular solution and $\boldsymbol{N\lambda}$ is the homogeneous solution to (10).

Property c) is equivalent to the condition $f_0'(t) > 0$ for $\forall t \in [0, 1]$, where $f_0'(t)$ denotes the derivative of $f_0(t)$. We approximate this condition by taking a finite number of sample points $t_l$ ($l = 1, 2, \ldots, L$) in $[0, 1]$ and rewrite it as

$$\boldsymbol{Ac} > \boldsymbol{0} \tag{12}$$

where

$$\boldsymbol{A} = \begin{bmatrix} f_1'(t_1) & f_2'(t_1) & \cdots & f_K'(t_1) \\ f_1'(t_2) & f_2'(t_2) & \cdots & f_K'(t_2) \\ \vdots & \vdots & \ddots & \vdots \\ f_1'(t_L) & f_2'(t_L) & \cdots & f_K'(t_L) \end{bmatrix}.$$

Substituting (11) into (12) yields

$$\boldsymbol{AN\lambda} > -\boldsymbol{Ac}_0. \tag{13}$$

Equation (13) defines a convex feasible region of $\boldsymbol{\lambda}$, which through (11) gives a convex feasible region of $\boldsymbol{c}$, as depicted in Fig. 4a. Sampling the feasible region of $\boldsymbol{\lambda}$ and substituting obtained feasible $\boldsymbol{\lambda}$'s into (11), from (9) we then obtain valid canonical time warping functions that possess all the three properties. Fig. 4b plots the derived valid canonical time warping functions using basis functions $t$, $t^2$, and $t^3$.

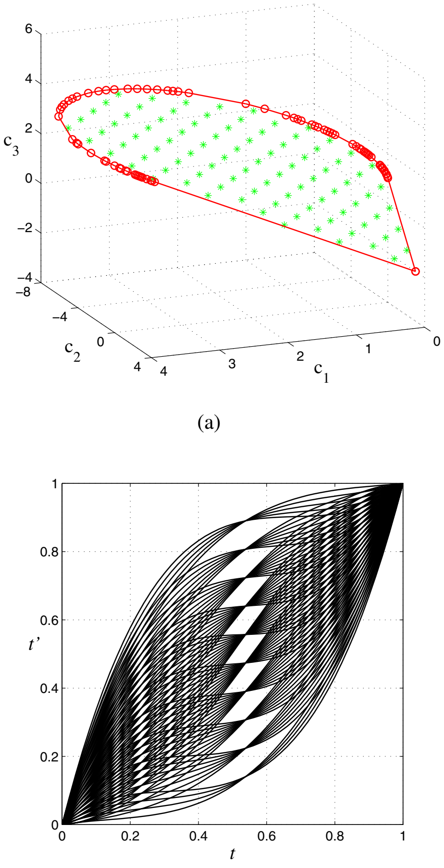
### C. Optimization for a Time Warping Function

Assume that a reference motion is infeasible between time $t_s$ and $t_e$ ($t_e > t_s$) and we use a time warping function to modify the segment. Based on a valid canonical time warping function $f_0(t)$, we can write the time warping function $f(t)$ for the segment between $t_s$ and $t_e$ as

$$f(t) = f_0\left(\frac{t - t_s}{t_e - t_s}\right)(t_e - t_s) + t_s. \tag{14}$$

Thanks to the three properties of $f_0(t)$ given in the previous subsection, $f(t)$ possesses the following properties:

a) $f(t_s) = t_s$;
b) $f(t_e) = t_e$;
c) $f(t)$ is strictly increasing in $[t_s, t_e]$.



(a)

(b)

Fig. 4. Example of canonical time warping function: $f_0(t) = c_1 t + c_2 t^2 + c_3 t^3$. (a) Feasible region (outlined by the red line) and samples (green asterisks) of coefficients such that $f_0(t)$ is strictly increasing in $[0, 1]$. (b) Plots of corresponding valid canonical time warping functions.

After choosing the time warping function $f(t)$, we verify and quantify the feasibility of the modified motion between $t_s - \Delta t$ and $t_e + \Delta t$ ($\Delta t > 0$) by the maximum value of the motion feasibility index $d$ over the motion segment, i.e.,

$$Q = \max_{t_s - \Delta t \le t \le t_e + \Delta t} d(t). \tag{15}$$

It should be noted that, though the time warping function changes only the times of motion frames between $t_s$ and $t_e$, the velocity and acceleration of the motion before $t_s$ and after $t_e$ may also be changed because motion frames between $t_s$ and $t_e$ are also used in motion interpolation before $t_s$ and after $t_e$, as illustrated in Fig. 3. The value of $\Delta t$ depends on how many motion frames are used in the interpolation, which determines how long of the motion before $t_s$ or after $t_e$ is affected by changing the times between $t_s$ and $t_e$.

For a chosen set of basis functions, we may have many valid canonical time warping functions, as shown in Fig. 4b, which provide multiple choices of the time warping function for an infeasible motion segment. Among all the time warping functions we seek the one such that the cost function $Q$ defined by (15) is minimal. Hopefully, the minimum value is non-positive, which means that the modified motion becomes feasible; otherwise, we may try different basis functions for time warping.

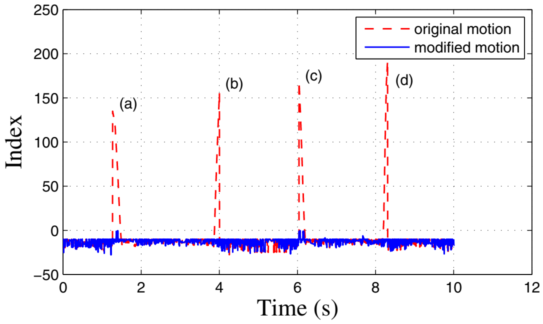


Fig. 5. Values of the motion feasibility index for the original and modified side stepping motion. The original motion becomes infeasible (index being positive) when the right foot of the robot (a) starts to lift off or (b) is about to touch down or the left foot (c) starts to lift off or (d) is about to touch down. By time warping, the modified motion is completely feasible for the robot (index being negative all the time).

## VI. Simulation Results

Here we verify the effectiveness of the proposed method through simulation with different motions. The dynamics simulator with rigid-contact model developed by University of Tokyo [27], [28] is used to conduct our experiments. The humanoid robot model used in the simulations has 25 joints (7 in each leg, 4 in each arm, and 3 in the torso) and 31 DoF including the translation and rotation of the floating base. The motion tracking controller proposed in [25] is used for the robot to reproduce a reference motion.

### A. Side Stepping

We first use a side stepping motion to demonstrate our method. Fig. 5 plots the value of the motion feasibility index for the motion. It can be seen that the index is positive at some moments, which implies that the motion is not entirely feasible for the robot. We take the time warping function to be a cubic function for every infeasible motion segment. Then, the index for the modified motion becomes all negative (Fig. 5), which implies that the motion becomes feasible after time warping. Fig. 6a shows the snapshots comparing the original and modified motions. It can be observed that time warping accelerates the robot's body motion towards one supporting foot side while the other foot is about to lift off. This ensures that the weight of the robot can be supported by only one foot after the other foot lifts off and the robot will not lose balance. Similarly, while the liftoff foot is about to touch down, the robot's body motion is slowed down, which wins the time for the liftoff foot to touch down. Fig. 6b shows the snapshots of the simulated motion when the robot tries to reproduce the modified motion. It can be seen that there is no significant difference between the two motions and the robot is able to reproduce the stepping motion well. The original, modified, and simulated motions are also shown in the accompanying video.

### B. Walking Upstairs

Thanks to the general motion feasibility index, the proposed motion adaptation method can be applied to motions on any terrain. In this example, we let the robot mimic a human walking upstairs, where each stair is about 0.083 m high. Fig. 7 shows the values of the motion feasibility index for the original and modified motions. Cubic functions are selected as the time warping functions, which successfully turn all the infeasible motion segments into feasible ones. The comparison between the original and modified motions and between the modified and simulated motions is exhibited in Fig. 8 and the accompanying video.

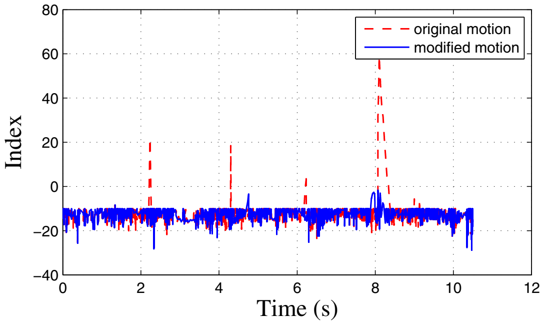


Fig. 7. Values of the motion feasibility index for the original and modified walking upstairs motion.

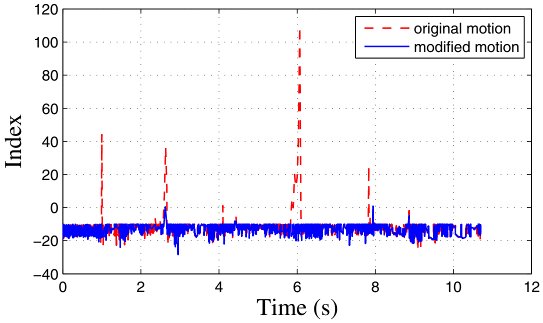


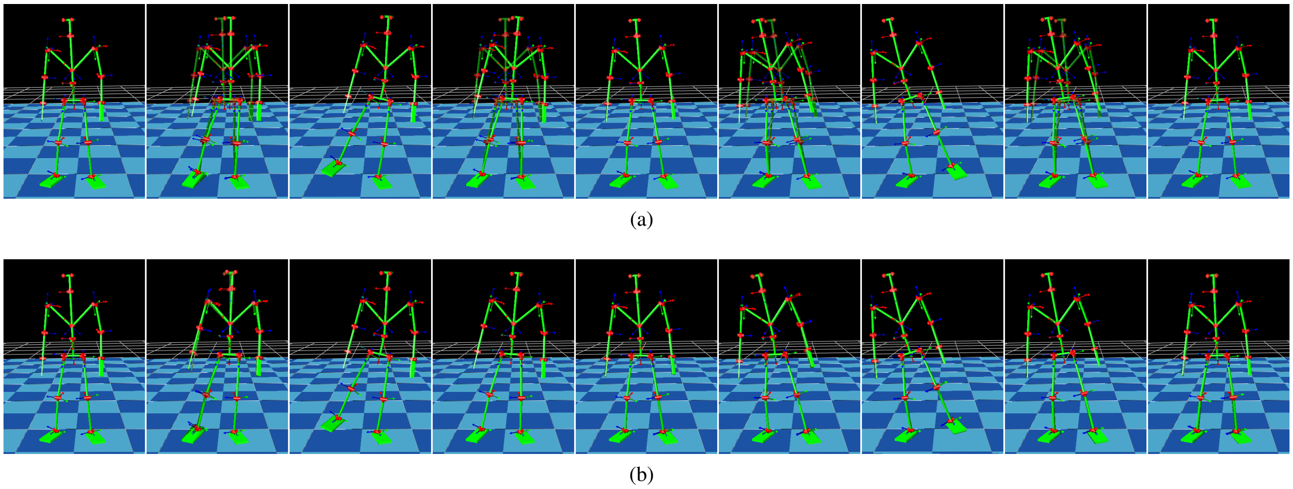
Fig. 9. Values of the motion feasibility index for the original and modified walking downstairs motion.

### C. Walking Downstairs

Similarly to the previous example, the robot is required to walk downstairs in this example. The value of the motion feasibility index is plotted in Fig. 9. The motions are shown in Fig. 10 and the video. Though the modified motion is still infeasible at some moments, its index value is just slightly above zero and the motion is very close to a feasible motion. This small extent of infeasibility for such a short period can be well handled by the motion tracking controller and the robot can get back on track before going too far. That is why the simulated motion has no big deviation from the modified motion, as shown in Fig. 10b and the accompanying video.

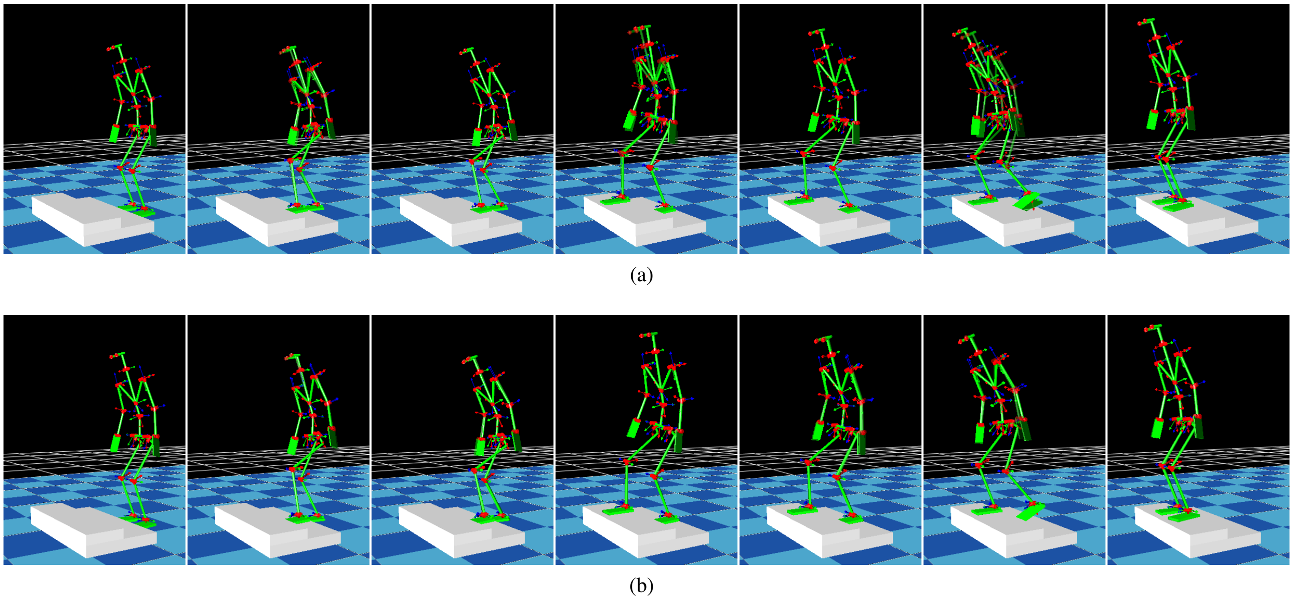
## VII. Conclusion and Future Work

This paper presents a method based on time warping for adapting human motions to humanoid robots by changing the time line of the motion. An advantage of this method is that it only changes the velocity and acceleration profiles of the motion and allows the robot to realize the same poses as in the reference motion. A general motion feasibility index


(a)

(b)

Fig. 6. Snapshots of side stepping. (a) Comparison of modified (solid) and original (transparent) motions. (b) Comparison of simulated (solid) and modified (transparent) motions.


(a)

(b)

Fig. 8. Snapshots of walking upstairs. (a) Comparison of modified (solid) and original (transparent) motions. (b) Comparison of simulated (solid) and modified (transparent) motions.

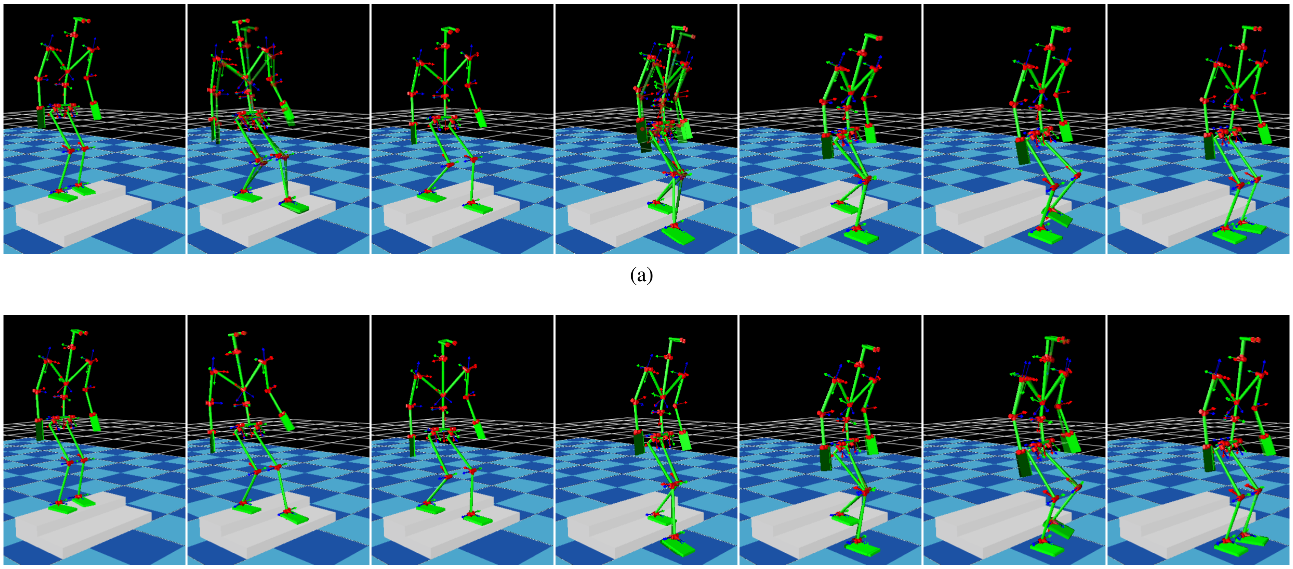
is proposed to identify infeasible motion segments for the robot in the reference motion. Then, each infeasible motion segment is modified through time warping. The proposed method has been tested on different human motions and is able to find feasible motions in most cases or at least bring an infeasible motion close to a feasible one, which makes it easier and safer for the robot to reproduce those motions.

This work can be extended in many directions. First, the choice of a time warping function can be arbitrary. However, it can be seen from Fig. 4b that a time warping function can either first speed up and then slow down the motion or do the reverse. It might be possible to identify whether a motion needs to be sped up or slowed down first to make it feasible, which will reduce the search scope of time warping functions and facilitate the use of this method. Second, the proposed motion feasibility index is general and applicable to motions happening on any terrains or the case that a humanoid robot needs to contact the environment with links other than feet. Hence, the method will be tested on more challenging human motions. Third, we currently modify a motion in the time domain. We can modify a motion in the joint or operation space as well based on the general motion feasibility index, which will also be investigated in the future.

## References

[1] S. Kajita, F. Kanehiro, K. Kaneko, K. Yokoi, and H. Hirukawa, “The 3D linear inverted pendulum mode: a simple modeling for a biped walking pattern generation,” in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, Maui, Hawaii, 2001, pp. 239–246.

[2] S. Kajita, F. Kanehiro, K. Kaneko, K. Fujiwara, K. Harada, K. Yokoi, and H. Hirukawa, “Biped walking pattern generation by using preview control of zero-moment point,” in *Proc. IEEE Int. Conf. Robot. Automat.*, Taipei, Taiwan, 2003, pp. 1620–1626.



(a)

(b)

Fig. 10. Snapshots of walking downstairs. (a) Comparison of modified (solid) and original (transparent) motions. (b) Comparison of simulated (solid) and modified (transparent) motions.

[3] K. K. Yin, K. Loken, and M. van de Panne, "SIMBICON: Simple biped locomotion control," *ACM Transactions on Graphics*, vol. 26, no. 3, p. 105, 2007.

[4] U. Muico, Y. J. Lee, J. Popović, and Z. Popović, "Contact-aware nonlinear control of dynamic characters," *ACM Transactions on Graphics*, vol. 28, no. 3, p. 81, 2009.

[5] A. Billard, S. Calinon, R. Dillman, and S. Schaal, "Robot programming by demonstration," in *Springer Handbook of Robotics*, B. Siciliano and O. Khatib, Eds. Springer, 2008, pp. 1371–1394.

[6] S. Schaal, A. Ijspeert, and A. Billard, "Computational approaches to motor learning by imitation," *Phylosophical Transactions of the Royal Society of London B: Biological Sciences*, vol. 358, pp. 537–547, 2003.

[7] A. Ude, C. Atkeson, and M. Riley, "Planning of joint trajectories for humanoid robots using b-spline wavelets," in *Proc. IEEE Int. Conf. Robot. Automat.*, San Francisco, CA, 2000, pp. 2223–2228.

[8] M. Ruchanurucks, S. Nakaoka, S. Kudoh, and K. Ikeuchi, "Generation of humanoid robot motions with physical constraints using hierarchical b-spline," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2005, pp. 1869–1874.

[9] W. Suleiman, E. Yoshida, F. Kanehiro, J.-P. Laumond, and A. Monin, "On human motion imitation by humanoid robot," in *Proc. IEEE Int. Conf. Robot. Automat.*, Pasadena, CA, 2008, pp. 2697– 2704.

[10] A. Dasgupta and Y. Nakamura, "Making feasible walking motion of humanoid robots from human motion captured data," in *Proc. IEEE Int. Conf. Robot. Automat.*, Detroit, MI, 1999, pp. 1044–1049.

[11] S. Nakaoka, A. Nakazawa, K. Yokoi, H. Hirukawa, and K. Ikeuchi, "Generating whole body motions for a biped humanoid robot from captured human dances," in *Proc. IEEE Int. Conf. Robot. Automat.*, Taipei, Taiwan, 2003, pp. 3905–3910.

[12] S. Kim, C. H. Kim, B. You, and S. Oh, "Stable whole-body motion generation for humanoid robots to imitate human motions," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, St. Louis, MO, 2009, pp. 2518–2524.

[13] L. Boutin, A. Eon, S. Zeghloul, and P. Lacouture, "An auto-adaptable algorithm to generate human-like locomotion for different humanoid robots based on motion capture data," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, Taipei, Taiwan, 2010, pp. 634–639.

[14] C. Ott, D. Lee, and Y. Nakamura, "Motion capture based human motion recognition and imitation by direct marker control," in *Proc. IEEE-RAS Int. Conf. Humanoid Robots*, Daejeon, Korea, 2008, pp. 399–405.

[15] K. Yamane and J. Hodgins, "Simultaneous tracking and balancing of humanoid robots for imitating human motion capture data," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, St. Louis, 2009, pp. 2510–2517.

[16] ——, "Control-aware mappping of human motion data with stepping for humanoid robots," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, Taipei, Taiwan, 2010, pp. 726–733.

[17] R. Vuga, M. Ogrinc, A. Gams, T. Petric, N. Sugimoto, A. Ude, and J. Morimoto, "Motion capture and reinforcement learning of dynamically stable humanoid movement primitives," in *Proc. IEEE Int. Conf. Robot. Automat.*, Karlsruhe, Germany, 2013, pp. 5284–5290.

[18] S. Kuindersma, F. Permenter, and R. Tedrake, "An efficiently solvable quadratic program for stabilizing dynamic locomotion," in *Proc. IEEE Int. Conf. Robot. Automat.*, Hong Kong, China, 2014, accepted.

[19] J. M. Hollerbach, "Dynamic scaling of manipulator trajectories," in *Proceedings of American Control Conference*, San Francisco, CA, 1983, pp. 752–756.

[20] M. Vukobratović and B. Borovac, "Zero-moment point: thirty five years of its life," *International Journal of Humanoid Robotics*, vol. 1, no. 1, pp. 157–173, 2004.

[21] F. Kanehiro, W. Suleiman, F. Lamiraux, E. Yoshida, and J.-P. Laumond, "Integrating dynamics into motion planning for humanoid robots," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, Nice, France, 2008, pp. 660–667.

[22] A. Goswami, "Postural stability of biped robots and the foot-rotation indicator (FRI) point," *International Journal of Robotics Research*, vol. 18, no. 6, pp. 523–533, 1999.

[23] H. Hirukawa, S. Hattori, K. Harada, S. Kajita, K. Kaneko, F. Kanehiro, K. Fujiwara, and M. Morisawa, "A universal stability criterion of the foot contact of legged robots - adios zmp," in *Proc. IEEE Int. Conf. Robot. Automat.*, Orlando, Florida, 2006, pp. 1976–1983.

[24] C. Ott, M. A. Roa, and G. Hirzinger, "Posture and balance control for biped robots based on contact force optimization," in *Proc. IEEE-RAS Int. Conf. Humanoid Robots*, Bled, Slovenia, 2011, pp. 26–33.

[25] Y. Zheng and K. Yamane, "Human motion tracking control with strict contact force constraints for floating-base humanoid robots," in *Proc. IEEE-RAS Int. Conf. Humanoid Robots*, Atlanta, GA, 2013, accepted.

[26] Y. Zheng and C.-M. Chew, "Distance between a point and a convex cone in *n*-dimensional space: computation and applications," *IEEE Transactions on Robotics*, vol. 25, no. 6, pp. 1397–1412, 2009.

[27] K. Yamane and Y. Nakamura, "A numerical robust LCP solver for simulating articulated rigid bodies in contact," in *Robotics: Science and Systems*, 2008.

[28] ——, "Dynamics simulation of humanoid robots: forward dynamics, contact, and experiments," in *The 17th CISM-IFToMM Symposium on Robot Design, Dynamics, and Control*, 2008.