

Full paper

Enhancing Zero Moment Point-Based Control Model: System Identification Approach

Wael Suleiman *, Fumio Kanehiro, Kanako Miura and Eiichi Yoshida

CNRS-AIST JRL (Joint Robotics Laboratory), UMI3218/CRT, Intelligent Systems Research
Institute, National Institute of Advanced Industrial Science and Technology (AIST), Tsukuba
Central 2, 1-1-1 Umezono, Tsukuba, Ibaraki 305-8568, Japan

Received 24 August 2010; accepted 20 October 2010

Abstract

The approximation of a humanoid robot by an inverted pendulum is one of the most frequently used models to generate a stable walking pattern using a planned zero moment point (ZMP) trajectory. However, on account of the difference between the multibody model of the humanoid robot and the simple inverted pendulum model, the ZMP error might be bigger than the polygon of support and the robot falls down. To overcome this limitation, we propose to improve the accuracy of the inverted pendulum model using system identification techniques. The candidate model is a quadratic in the state space representation. To identify this system, we propose an identification method that is the result of the comprehensive application of system identification to dynamic systems. Based on the quadratic system, we also propose controlling algorithms for on-line and off-line walking pattern generation for humanoid robots. The efficiency of the quadratic system and the walking pattern generation methods has been successfully shown using dynamical simulation and conducting real experiments on the cybernetic human HRP-4C.

© Koninklijke Brill NV, Leiden and The Robotics Society of Japan, 2011

Keywords

Humanoid robot, zero moment point control, system identification, optimization, nonlinear system control

1. Introduction

Research on biped humanoid robots is nowadays one of the most active research fields in robotics. Making humanoid robots walk has been the subject of intensive investigation over recent years. Many researchers have proposed different methods to generate a stable walking motion for a humanoid robot [1–8]. Most of these methods use a simplified model based on the approximation of the humanoid robot

* To whom correspondence should be addressed. E-mail: wael.suleiman@aist.go.jp

by an inverted pendulum model. The mass of the inverted pendulum coincides with the centre of mass (CoM) of the humanoid robot.

In order to generate a stable motion (dynamically balanced), those methods use the zero moment point (ZMP) principle [9]. The ZMP is a point of the support polygon (i.e., the convex hull of all points of contact between the support foot (feet) and the ground) at which the horizontal moments are vanishing.

The cart-table model proposed by Kajita *et al.* [3] is one of these methods. The efficiency of this method has been proved by generating on-line stable walking motions using a planned trajectory of the ZMP.

Some methods have been proposed to compensate for the error between the simple inverted pendulum model and the multibody model of the humanoid robot [10, 11]. To achieve the same goal, however, the purpose in this paper is to show that the linear model related to the cart-table model can be improved by capturing the dynamic difference between the the multibody model of the humanoid robot and that of the cart-table model, then identifying a quadratic system [12] that is able to approximate this difference. Furthermore, we aim to give insights into the problem of controller design of the quadratic system in order to generate dynamically stable walking patterns. The dynamic balance of the humanoid robot during the execution of the ZMP trajectory is subject to several perturbations, such as inaccuracy of the dynamic and kinematics parameters, and the flexibility in some joints of the humanoid robot. However, by guaranteeing that the ZMP trajectory is always inside a safety margin of stability, the on-line stabilizer can ensure the dynamic balance of the humanoid robot. Therefore, the dynamic stability is guaranteed by tracking a predefined ZMP trajectory that is always inside the polygon of support while respecting a safety margin.

The proposed controlling algorithms are classified into two categories:

- (i) Off-line walking pattern generation algorithms: in this category the computational time is not considered and the objective is to track the desired ZMP trajectory precisely.
- (ii) On-line walking pattern generation algorithms: in this category the computational complexity of the controller should respect the real-time constraints. In this case, the desired ZMP trajectory is tracked as much as possible.

This paper is organized as follows. An overview of the cart-table model and the proposed model to improve the reliability of the cart-table model is discussed in Section 2. An adapted identification method for the identification of the quadratic system is developed in Section 3. Section 4 points out the problem of designing a controller to track a desired ZMP trajectory. The efficiency of the proposed quadratic system and the walking generation algorithms is shown in Section 5 through real experiments on the cybernetic human HRP-4C [13].

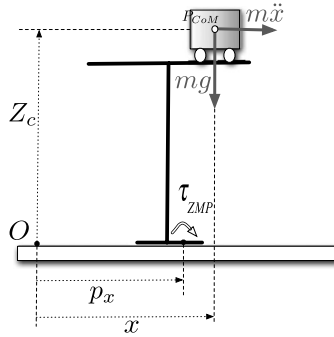


Figure 1. Cart-table model.

2. Cart-Table Model and Quadratic System

The main idea of the cart-table model [3] is to approximate the humanoid robot by a mass located at its CoM and it is equal to the total mass of the humanoid robot. Therefore, the complex problem of controlling the humanoid robot is transformed to the control of an inverted pendulum.

Let us define the Cartesian position of the CoM (P_{CoM}) by:

$$P_{\text{CoM}} = \begin{bmatrix} x \\ y \\ z_c \end{bmatrix}. \quad (1)$$

We suppose that the vertical position of the mass z_c is constant.

The ZMP is a particular point of the horizontal plane at which the horizontal moments vanish. For the inverted pendulum, it is defined as:

$$\text{ZMP} = \begin{bmatrix} p_x \\ p_y \end{bmatrix} = \begin{bmatrix} x - \frac{z_c}{g} \ddot{x} \\ y - \frac{z_c}{g} \ddot{y} \end{bmatrix}, \quad (2)$$

where g is the absolute value of gravity acceleration.

It is clear from (2) that the two elements of the ZMP are very similar. For that, in the following, we only consider how to build the controller for p_x .

To control p_x , Kajita *et al.* [3] proposed to consider the cart-table model that is shown in Fig. 1. It depicts a running cart of mass m on a pedestal table whose mass is negligible.

2.1. Controlling the ZMP

Let us define a new variable u_x as the time derivative of $\ddot{x}$:

$$u_x = \frac{d\ddot{x}}{dt}. \quad (3)$$

The variable u_x is usually called the jerk.

Regarding u_x as the input of p_x , we can translate the equation of p_x into the

following dynamic system:

$$\begin{aligned} \frac{d}{dt} \begin{bmatrix} x \\ \dot{x} \\ \ddot{x} \end{bmatrix} &= \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} x \\ \dot{x} \\ \ddot{x} \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} u_x \\ p_x &= \begin{bmatrix} 1 & 0 & -\frac{z_c}{g} \end{bmatrix} \begin{bmatrix} x \\ \dot{x} \\ \ddot{x} \end{bmatrix}. \end{aligned} \quad (4)$$

As the humanoid robot is controlled in discrete time, we discretize the system (4) with sampling time T_s ; the discrete control model for p_x can be expressed by:

$$\begin{aligned} z_{k+1} &= Az_k + B\bar{u}_k \\ p_k &= Cz_k, \end{aligned} \quad (5)$$

where:

$$\begin{aligned} z_k &\triangleq [x(kT_s) \quad \dot{x}(kT_s) \quad \ddot{x}(kT_s)]^T \\ \bar{u}_k &\triangleq u_x(kT_s) \\ p_k &\triangleq p_x(kT_s) \\ A &= \begin{bmatrix} 1 & T_s & T_s^2/2 \\ 0 & 1 & T_s \\ 0 & 0 & 1 \end{bmatrix}, \quad B = \begin{bmatrix} T_s^3/6 \\ T_s^2/2 \\ T_s \end{bmatrix}, \quad C = \begin{bmatrix} 1 & 0 & -\frac{z_c}{g} \end{bmatrix}. \end{aligned} \quad (6)$$

Regarding the similarity between the equations of p_x and p_y , the above equation can be extended to obtain the ZMP-based control model as:

$$\begin{aligned} X_{k+1} &= A_1 X_k + B_1 u_k \\ p_k &= C_1 X_k, \end{aligned} \quad (7)$$

where:

$$\begin{aligned} X_k &\triangleq \begin{bmatrix} x(kT_s) \\ \dot{x}(kT_s) \\ \ddot{x}(kT_s) \\ y(kT_s) \\ \dot{y}(kT_s) \\ \ddot{y}(kT_s) \end{bmatrix}, \quad u_k \triangleq \begin{bmatrix} u_x(kT_s) \\ u_y(kT_s) \end{bmatrix}, \quad p_k \triangleq \begin{bmatrix} p_x(kT_s) \\ p_y(kT_s) \end{bmatrix} \\ A_1 &= I_2 \otimes A, \quad B_1 = I_2 \otimes B, \quad C_1 = I_2 \otimes C. \end{aligned} \quad (8)$$

$I_2 \in \mathbb{R}^{2 \times 2}$ is the identity matrix and $\otimes$ is the operator of Kronecker product, which is defined as follows: $X \in \mathbb{R}^{m \times n}$, $Y \in \mathbb{R}^{p \times l}$:

$$X \otimes Y \in \mathbb{R}^{mp \times nl} \triangleq \begin{bmatrix} x_{11}Y & \cdots & x_{1n}Y \\ \vdots & \ddots & \vdots \\ x_{m1}Y & \cdots & x_{mn}Y \end{bmatrix}, \quad (9)$$

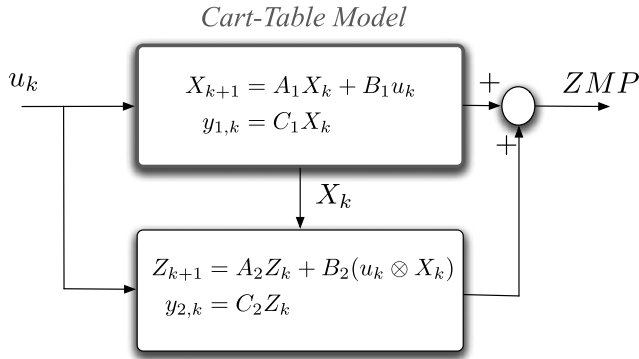


Figure 2. ZMP model of the humanoid robot regarded as a quadratic system.

where x_{ij} is the element of the i th line and the j th column of the matrix X .

The above model has been used to generate a stable biped walking pattern using a preview controller [3]. However, when the error between the output of this model and the real ZMP trajectory of the humanoid robot becomes bigger than the stability margin, then the robot falls down. The solution proposed by Kajita *et al.* [3] is to inject this error in the preview controller as a second stage in order to eliminate the dynamic error. However, this procedure requires the dynamic simulation of the multibody model of the humanoid robot; as a result, it is a time-consuming process.

To overcome this problem, we propose to model the ZMP of the humanoid robot by the model given in Fig. 2. This model consists of two blocks: (i) the previous cart-table model and (ii) a linear system with respect to the input $u_k \otimes X_k$.

The obtained model is a quadratic system in the state space representation. This realization of the quadratic system is given by:

$$\begin{aligned}
 X_{k+1} &= A_1 X_k + B_1 u_k \\
 Z_{k+1} &= A_2 Z_k + B_2(u_k \otimes X_k) \\
 p_k &= C_1 X_k + C_2 Z_k,
 \end{aligned} \tag{10}$$

where $u_k \in \mathbb{R}^2$ is the input signal of the system expressed as in (8). $p_k \in \mathbb{R}^2$ is the output of the system (ZMP).

$\{X_k, Z_k\}$ are the states, the dimension of X_k is equal to 6 and the dimension of Z_k is n , which will be determined by the identification algorithm.

We have proven [12, 14] that this class of nonlinear system is a special case of the Volterra series of degree 2. This class can play a useful role to model many real systems in practice.

3. System Identification

It is clear that the system is defined by the coefficient matrices of (10). The unknown matrices are A_2 , B_2 and C_2 . That means that the structure parameters can be given by

$$\theta = \begin{bmatrix} \text{vec}(A_2) \\ \text{vec}(B_2) \\ \text{vec}(C_2) \end{bmatrix}, \quad (11)$$

where $\text{vec}(\cdot)$ denotes the vectorization operator, which is defined as follows

$$\text{vec} : M \in \mathbb{R}^{m \times n} \rightarrow \mathbb{R}^{m \cdot n}$$

$$\text{vec}(M) = \text{vec} [m_1 \quad m_2 \quad \cdots \quad m_n] = [m_1^T \quad m_2^T \quad \cdots \quad m_n^T]^T.$$

Given the input u_k and the output p_k of the real system, our goal is achieved if the output of the following model:

$$\begin{aligned} \hat{X}_{k+1} &= A_1 \hat{X}_k + B_1 u_k \\ \hat{Z}_{k+1}(\theta) &= A_2(\theta) \hat{Z}_k(\theta) + B_2(\theta)(u_k \otimes \hat{X}_k) \\ \hat{p}_k(\theta) &= C_1 \hat{X}_k + C_2(\theta) \hat{Z}_k(\theta), \end{aligned} \quad (12)$$

approximates the output p_k of the real system accurately enough.

This criterion can be transformed into the minimization of the output error with respect to the parameter θ . Considering a data set of N samples, the output error cost function is given by:

$$J_N(\theta) = \frac{1}{N} \sum_{k=1}^N \|p_k - \hat{p}_k(\theta)\|_2^2 = \frac{1}{N} E_N(\theta)^T E_N(\theta), \quad (13)$$

where $E_N(\theta) = [e(1)^T e(2)^T \cdots e(N)^T]^T$ is the error vector in which $e(k) = p_k - \hat{p}_k(\theta)$. The minimization of (13) is a nonlinear, nonconvex optimization problem. The numerical solution of this problem can be calculated by different algorithms. For instance, the iterative gradient search method is a popular one. This iterative method is based on the updating of the system parameters $\hat{\theta}$ as follows:

$$\hat{\theta}^i = \hat{\theta}^{i-1} - (\psi_N^T(\hat{\theta}^{i-1}) \psi_N(\hat{\theta}^{i-1}) + \lambda^i I)^{-1} \psi_N^T(\hat{\theta}^{i-1}) E_N(\hat{\theta}^{i-1}), \quad (14)$$

where λ^i is the regularization parameter and:

$$\psi_N(\theta) \triangleq \frac{\partial E_N(\theta)}{\partial \theta^T}, \quad (15)$$

is the Jacobian of the error vector $E_N(\theta)$.

3.1. Multiple-Experiment Identification

In fact, the model obtained by considering a single experiment might be not reliable [15]. This is because the input signals are not perfectly excited. To overcome this problem in identifying practical applications, we apply various sets of input

signals. The objective of each set is exciting one or more modes of the system. The multiple experiments should be exploited simultaneously to obtain an accurate model of the system.

The optimization problem that considers K experiments simultaneously can be formulated as:

$$J_K(\theta) = \frac{1}{K} \sum_{i=1}^K \frac{1}{N_i} \sum_{k=1}^{N_i} \|p_k^i - \hat{p}_k^i(\theta)\|_2^2 = \frac{1}{K} E_K(\theta)^T E_K(\theta), \quad (16)$$

where:

$$E_K(\theta) = [E_{N_1}^1(\theta)^T E_{N_2}^2(\theta)^T \cdots E_{N_K}^K(\theta)^T]^T, \quad (17)$$

and:

$$E_{N_i}^i(\theta) = \frac{1}{\sqrt{N_i}} [e^i(1)^T e^i(2)^T \cdots e^i(N_i)^T]^T, \quad (18)$$

is the error vector in which $e^i(k) = p_k^i - \hat{p}_k^i(\theta)$. p_k^i is the output of the system according to the input u_k^i .

The estimated output $\hat{p}_k^i(\theta)$ of the data set number i is given by:

$$\begin{aligned} \hat{X}_{k+1}^i &= A_1 \hat{X}_k^i + B_1 u_k^i \\ \hat{Z}_{k+1}^i(\theta) &= A_2(\theta) \hat{Z}_k^i(\theta) + B_2(\theta)(u_k^i \otimes \hat{X}_k^i) \\ \hat{p}_k^i(\theta) &= C_1 \hat{X}_k^i + C_2(\theta) \hat{Z}_k^i(\theta). \end{aligned} \quad (19)$$

The minimization of (16) can be calculated, similarly to the case of a single experiment, by using the iterative gradient search method as:

$$\hat{\theta}^{l+1} = \hat{\theta}^l - (\psi_K^T(\hat{\theta}^l) \psi_K(\hat{\theta}^l) + \lambda^l I)^{-1} \psi_K^T(\hat{\theta}^l) E_K(\hat{\theta}^l), \quad (20)$$

where:

$$\psi_K(\theta) = \begin{bmatrix} \psi_{N_1}^1(\theta) \\ \psi_{N_2}^2(\theta) \\ \vdots \\ \psi_{N_K}^K(\theta) \end{bmatrix} \quad (21)$$

and:

$$\psi_{N_i}^i(\theta) \triangleq \frac{\partial E_{N_i}^i(\theta)}{\partial \theta^T}, \quad (22)$$

is the Jacobian of the error vector $E_{N_i}^i$.

In order to apply the identification algorithm, the input and output signals are supposed to be known. In our case, a sequence of walking patterns has been deployed to identify the quadratic system. The input data sets have been calculated using the method proposed by Kajita *et al.* [3] by defining the ZMP trajectories (the output signals).

3.2. Implementation Algorithm of the Identification Method

The first step in applying the identification algorithm is determining the optimal dimension (n) of the state Z_k . In order to achieve this task, a method based on oblique projection and the subspace identification method can be used (for more details, refer to Refs [12, 14]). Another strategy is to choose n arbitrarily and then change this value in order to obtain the quadratic system that fits accurately with the response of the real system.

Then, the implementation of the system identification algorithm can be summarized as:

Calculate the state $\hat{X}_k^i$, $\hat{Z}_k^i$ and $\hat{p}_k^i$ by simulating system (19) with $\theta = \hat{\theta}^{j-1}$.

Compute $E_{N_i}^i(\theta)$ using (18).

Calculate the matrix ψ_K using (21) and (22).

Calculate the update rule of the gradient search algorithm using (20) and obtain $\hat{\theta}^j$.

Perform the termination test for minimization. If true, the algorithm stops. Otherwise, return to Step (i), i.e., compute the values $J_L(\hat{\theta}^{j-1})$ and $J_L(\hat{\theta}^j)$ using (16), and test if $\|J_L(\hat{\theta}^j) - J_L(\hat{\theta}^{j-1})\|_2$ is small enough.

The above algorithm yields the matrices A_2 , B_2 and C_2 . Once the quadratic system is identified, not only the behavior of ZMP can be predicted using this model, but also the problem of ZMP servo tracking can be addressed as well.

4. Controller Design

The problem of designing a controller to follow a desired ZMP trajectory can be formulated as:

$$\begin{aligned} \min_{u_k} \sum_{k=0}^N u_k^T Q_u u_k + (p_k^{\text{ref}} - p_k)^T Q_e (p_k^{\text{ref}} - p_k) \quad \text{subject to} \\ X_{k+1} = A_1 X_k + B_1 u_k \\ Z_{k+1} = A_2 Z_k + B_2 (u_k \otimes X_k) \\ p_k = C_1 X_k + C_2 Z_k, \end{aligned} \quad (23)$$

where p_k^{ref} designs the desired ZMP trajectory and N is the number of last sampling of the trajectory.

In this section we propose one algorithm for off-line control and two algorithms for on-line control of the quadratic system. The off-line algorithm is useful, for instance, in the case of motion planning. In this case, the path of the humanoid robot is provided off-line. However, for reactive motions and human–robot interactions, the on-line control algorithms are crucial.

4.1. Off-Line Walking Pattern Generation Algorithm

It is well known that the space of the admissible solutions of the minimization problem (23) is very large. In order to transform this space to a smaller dimensional space, we can use a basis of shape functions.

Let us consider a basis of shape functions $\mathbf{S}_k$ that is defined as:

$$\mathbf{S}_k = [S_k^1 \quad S_k^2 \quad \cdots \quad S_k^l]^T, \quad (24)$$

where S_k^i denotes the value of the shape function number i at the instant k . The dimension of $\mathbf{S}_k$ is l , which defines the dimension of the basis of shape functions.

The projection of u_k into the basis of shape functions S_k can be given by:

$$u_k = (I_2 \otimes \mathbf{S}_k)^T \mathbf{u}_S, \quad (25)$$

where $I_2 \in \mathbb{R}^{2 \times 2}$ denotes the identity matrix and $\mathbf{u}_S \in \mathbb{R}^{2l}$ denotes the vector of B-spline functions control points.

Thus, the optimization problem (23) can be rewritten as:

$$\begin{aligned} \min_{\mathbf{u}_S} \sum_{k=0}^L \mathbf{u}_S^T (I_2 \otimes \mathbf{S}_k) Q_u (I_2 \otimes \mathbf{S}_k)^T \mathbf{u}_S + (p_k^{\text{ref}} - p_k)^T Q_e (p_k^{\text{ref}} - p_k) \quad \text{subject to} \\ X_{k+1} = A_1 X_k + B_1 (I_2 \otimes \mathbf{S}_k)^T \mathbf{u}_S \\ Z_{k+1} = A_2 Z_k + B_2 [(I_2 \otimes \mathbf{S}_k)^T \mathbf{u}_S] \otimes X_k \\ p_k = C_1 X_k + C_2 Z_k. \end{aligned} \quad (26)$$

By defining the following variables:

$$\begin{aligned} J = \sum_{k=0}^L \mathbf{u}_S^T (I_2 \otimes \mathbf{S}_k) Q_u (I_2 \otimes \mathbf{S}_k)^T \mathbf{u}_S + (p_k^{\text{ref}} - p_k)^T Q_e (p_k^{\text{ref}} - p_k) \\ H = \begin{bmatrix} X_{k+1} - A_1 X_k - B_1 (I_2 \otimes \mathbf{S}_k)^T \mathbf{u}_S \\ Z_{k+1} - A_2 Z_k - B_2 [(I_2 \otimes \mathbf{S}_k)^T \mathbf{u}_S] \otimes X_k \\ p_k - C_1 X_k - C_2 Z_k \end{bmatrix}, \end{aligned}$$

the optimization problem (26) is transformed into the following classical optimization problem:

$$\min_{\mathbf{u}_S} J(\mathbf{u}_S) \quad \text{subject to} \quad H(\mathbf{u}_S) = 0. \quad (27)$$

In other words, the optimization problem has been transformed into finding the vector $\mathbf{u}_S \in \mathbb{R}^{2l}$. Note that the trajectory of the CoM of the robot can be directly obtained from X_k by applying the input signal u_k to the quadratic system (10).

4.2. On-Line Walking Pattern Generation Algorithm I

The idea of this algorithm is to linearize the quadratic system and then control the linearized system. The linearized system of quadratic system (10) around the value

$X_k = X^*$ is given by:

$$\begin{aligned} X_{k+1} &= A_1 X_k + B_1 u_k \\ Z_{k+1} &= A_2 Z_k + B_2(u_k \otimes X^*) \\ p_k &= C_1 X_k + C_2 Z_k. \end{aligned} \quad (28)$$

Let us define the matrix $B_{2,j}$ as:

$$B_{2,j} = B_2(:, (j-1)n_1 + 1 : jn_1), \quad (29)$$

where n_1 denotes the dimension of X_k (in our case $n_1 = 6$) and the notation $M(:, i : j)$ designs the submatrix of matrix M that contains the columns from the i th to j th column.

Using (29), the variable $B_2(u_k \otimes X^*)$ can be formulated as:

$$\begin{aligned} B_2(u_k \otimes X^*) &= [B_{2,1}X^* \quad B_{2,2}X^* \quad \cdots \quad B_{2,m}X^*]u_k \\ &\triangleq B_2^*u_k. \end{aligned} \quad (30)$$

Using the above definitions, the linearized system (28) can be reformulated as:

$$\begin{aligned} \mathcal{X}_{k+1} &= \mathcal{A}\mathcal{X}_k + \mathcal{B}u_k \\ p_k &= \mathcal{C}\mathcal{X}_k, \end{aligned} \quad (31)$$

where:

$$\begin{aligned} \mathcal{X}_k &= \begin{bmatrix} X_k \\ Z_k \end{bmatrix}, \quad \mathcal{A} = \begin{bmatrix} A_1 & \mathbf{0} \\ \mathbf{0} & A_2 \end{bmatrix}, \\ \mathcal{B} &= \begin{bmatrix} B_1 & \mathbf{0} \\ \mathbf{0} & B_2^* \end{bmatrix}, \quad \mathcal{C} = [C_1 \quad C_2]. \end{aligned} \quad (32)$$

Therefore, the controller design problem (23) is transformed into the following problem:

$$\begin{aligned} \min_{u_k} \sum_{k=0}^L u_k^T Q_u u_k + (p_k^{\text{ref}} - p_k)^T Q_e (p_k^{\text{ref}} - p_k) \quad \text{subject to} \\ \mathcal{X}_{k+1} &= \mathcal{A}\mathcal{X}_k + \mathcal{B}u_k \\ p_k &= \mathcal{C}\mathcal{X}_k. \end{aligned} \quad (33)$$

The above optimization problem is well known in automatic control as the preview control of a linear system. The optimal input signal (u_k) can be obtained by applying the classical linear-quadratic state-feedback regulator (LQR) technique. Thus, when the ZMP reference is previewed for N_L future steps at every sampling time, the optimal controller that minimizes the objective function (33) is given by:

$$u_k = -K\mathcal{X}_k + [f_1 \quad f_2 \quad \cdots \quad f_{N_L}] \begin{bmatrix} p_{k+1}^{\text{ref}} \\ p_{k+2}^{\text{ref}} \\ \vdots \\ p_{k+N_L}^{\text{ref}} \end{bmatrix}, \quad (34)$$

where K and f_i are calculated as:

$$\begin{aligned} K &= (Q_u + \mathcal{B}^T \mathbf{P} \mathcal{B})^{-1} \mathcal{B}^T \mathbf{P} \mathcal{A} \\ f_i &= (Q_u + \mathcal{B}^T \mathbf{P} \mathcal{B})^{-1} \mathcal{B}^T (\mathcal{A}^T - K^T \mathcal{B}^T)^{(i-1)} \mathcal{C}^T Q_e. \end{aligned} \quad (35)$$

$\mathbf{P}$ is the solution of the following Riccati equation:

$$\mathbf{P} = \mathcal{A}^T \mathbf{P} \mathcal{A} + \mathcal{C}^T Q_e \mathcal{C} - \mathcal{A}^T \mathbf{P} \mathcal{B} (R + \mathcal{B}^T \mathbf{P} \mathcal{B})^{-1} \mathcal{B}^T \mathbf{P} \mathcal{A}. \quad (36)$$

Note that the linearized system (28) is valid for a region around the value $X_k = X^*$. Therefore, a threshold $\varepsilon > 0$ should be defined and the linearized system is updated with the new value of X_k if $\|X_k - X^*\| > \varepsilon$.

The trajectory of the CoM can be obtained directly from X_k by simulating the quadratic system (10) using the control signal u_k , which is the result of applying the above control algorithm.

4.3. On-Line Walking Pattern Generation Algorithm II

The key idea of this algorithm can be summarized as follows:

- (i) Controlling the quadratic system as a linear system. The approximated linear system is the linear subsystem (cart-table model) of the quadratic system.
- (ii) Calculating the error between the dynamic simulation of the linear system and the quadratic system.
- (iii) Correcting the error by considering it as perturbation and calculating the appropriate input signal to decrease its effect.

Note that this algorithm works well when the dominant part of the system's dynamic is the linear part. It is, however, the case of the walking motion.

Therefore, at first the optimization algorithm (23) is transformed into the following problem:

$$\begin{aligned} \min_{u_k} \quad & \sum_{k=0}^L u_k^T Q_u u_k + (p_k^{\text{ref}} - p_k)^T Q_e (p_k^{\text{ref}} - p_k) \quad \text{subject to} \\ & X_{k+1} = A_1 X_k + B_1 u_k \\ & p_k = C_1 X_k. \end{aligned} \quad (37)$$

Analogously to the previous control algorithm, the above optimization problem can be solved using the LQR technique.

The second step is to find the error between the outputs of the linear subsystem and the quadratic system by simulating the systems (37) and (10) using the optimal input signal u_k , and obtain:

$$\Delta p_k^{\text{ref}} = C_2 Z_k. \quad (38)$$

In order to decrease the effects of this error, one can consider it as an external perturbation and apply the LQR technique once more to calculate the variation of the input signal that should be added to the optimal u_k obtained from solving (37).

Thus, the new LQR problem can be formulated as:

$$\begin{aligned} \min_{u_k} \sum_{k=0}^L \Delta u_k^T Q_u \Delta u_k + (\Delta p_k^{\text{ref}} - \Delta p_k)^T Q_e (\Delta p_k^{\text{ref}} - \Delta p_k) \quad \text{subject to} \\ X_{k+1} = A_1 X_k + B_1 \Delta u_k \\ \Delta p_k = C_1 X_k. \end{aligned} \tag{39}$$

Analogously to (37) the variation of the input signal Δu_k can be obtained. As a result the input signal that should be applied to the humanoid robot becomes $u_k^* = u_k + \Delta u_k$.

5. Experimental Results

5.1. Humanoid Robot: Kinematic Structure

The proposed walking patterns generation algorithms have been validated using the cybernetic human HRP-4C[13]. HRP-4C is a life-size humanoid robot that was developed for entertainment use, such as a fashion model, master of ceremonies of various events, etc. For these applications, it has been decided to make it more human-like than the humanoid robots that have been developed so far [16, 17]. In order to realize a human-like shape, its dimensions are designed referring to a database of Japanese women of the 20th century (Table 1) [18]. The basic joint configurations are shown in Fig. 3 (right) (face and hand joints are not displayed).

5.2. Identification of the Quadratic System

In order to identify the quadratic system, we have simulated several walking patterns with different step length and walking speed values. We define the model

Table 1.
Specifications of the cybernetic human HRP-4C

Degrees of freedom	arm	6
	hand	2
	leg	6
	waist	3
	neck	3
	face	8
	total	42
Height	158 cm	
Weight (including batteries)	43 kg	
Sensors	force/torque sensor $\times$ 2, inertia sensor	

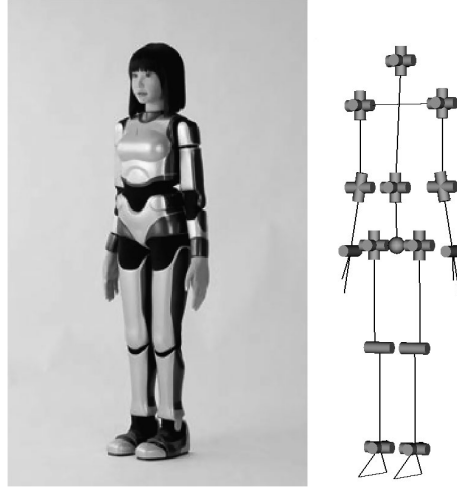


Figure 3. Exterior and joint configuration of cybernetic human HRP-4C (face and hand joints are not displayed).

accuracy as the percent variance accounted for (%VAF)

$$\%VAF \triangleq \left(1 - \frac{\sum_{k=1}^L (p_k - \hat{p}_k)^T (p_k - \hat{p}_k)}{\sum_{k=1}^L (p_k - \bar{p}_k)^T (p_k - \bar{p}_k)} \right) \times 100,$$

where $\hat{p}_k$ denotes the estimated output signal (the output of the quadratic system) and $\bar{p}_k$ is the mean value of the real output of the system p_k (ZMP trajectory of the multibody model; Fig. 4).

The optimal dimension of the state Z_k has been calculated according to the procedure explained in Ref. [14] and we obtained $n = 8$.

In order to validate the identified model, the number of conducted simulation walking patterns and the accuracy of the identified quadratic system model are given in Table 2.

Depending on the accuracy reported in Table 2, we have chosen the identified model that is the result of using 10 walking patterns. In Fig. 5, we observe that the quadratic system is perfectly able to capture the dynamic difference between the multibody model and the simple cart-table model. For more details on the identification procedure, refer to Ref. [19].

5.3. Results

5.3.1. Off-Line Walking Pattern Generation Algorithm

The application of the off-line walking pattern generation algorithm in order to track a desired trajectory of the ZMP is applied to the cybernetic human HRP4-C. We have chosen a basis (S_k) of 40 B-spline functions. The optimization algorithm stops when the norm of the error between the planned ZMP (p^{ref}) and the output of the quadratic system is less than 10^{-4} m. The ZMP error between the multibody

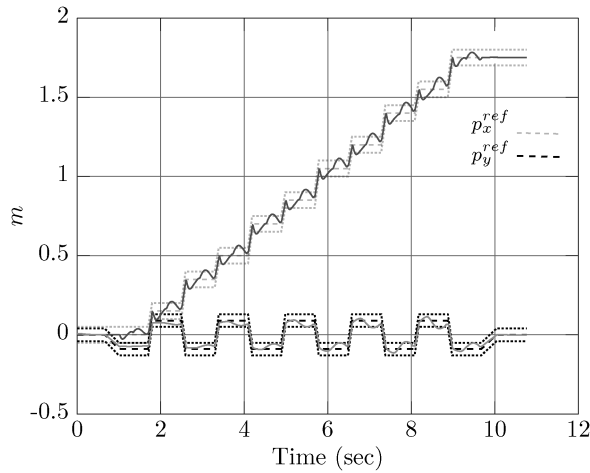


Figure 4. Reference ZMP trajectory (dashed lines), ZMP of the multibody model (solid lines) and safety region of stability (dotted lines).

Table 2.
Accuracy of the identified model

Number of experiments	Accuracy of the proposed model (%VAF)
2	89.7
5	93.1
10	97.2

model of the humanoid robot and the planned ZMP trajectory (p^{ref}) is given in Fig. 6.

From Fig. 6 we can conclude:

- (i) The planned ZMP trajectory is well tracked by the proposed controller.
- (ii) The error between the ZMP of the multibody model of the humanoid robot and the planned ZMP trajectory (p^{ref}) is small enough (less than 2 cm) and the ZMP stays always inside of the region of stability. Moreover, this proves that the quadratic system is an accurate model of the walking motion.

In order to validate the obtained results, we have simulated the generated motion using the dynamic simulator OpenHRP3 [20, 21]. Snapshots of the simulated walking pattern are given in Fig. 7.

5.3.2. On-Line Walking Pattern Generation Algorithm I

In this experiment, we apply the on-line walking pattern generation algorithm I to generate the CoM trajectory in order to track the desired trajectory of ZMP. The

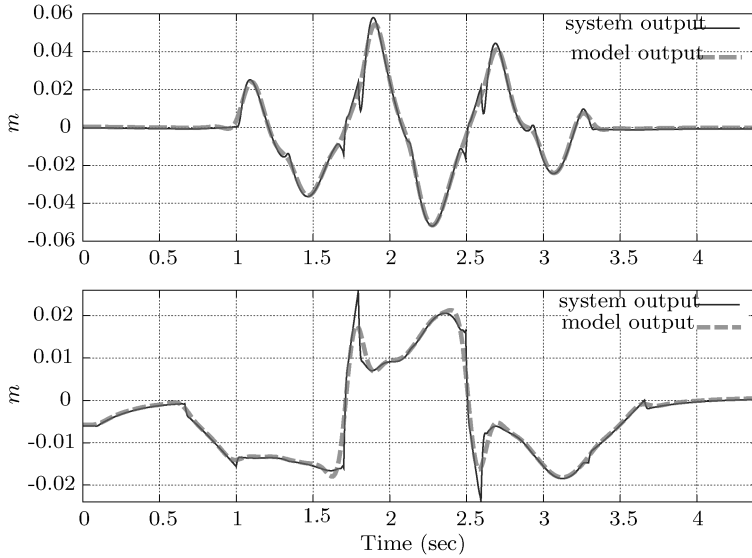


Figure 5. ZMP error between the multibody model of a humanoid robot and the simple cart-table model (solid line), and the output of the second block of the quadratic system (dashed line).

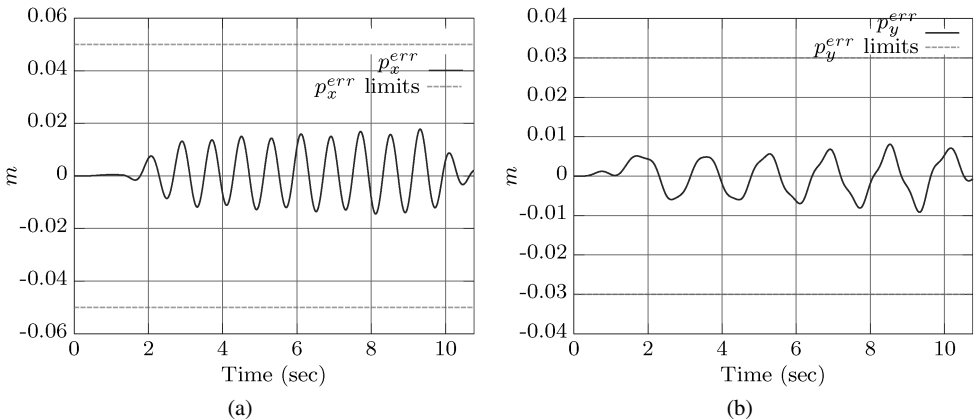


Figure 6. Off-line walking pattern generation algorithm: the ZMP error between the multibody model of the humanoid robot and the planned ZMP trajectory (p^{ref}) (solid line), and the safety margin of stability (dashed line). (a) The x element (p_x^{err}). (b) The y element (p_y^{err}).

ZMP error between the multibody model of the humanoid robot and the planned ZMP trajectory (p^{ref}) is given in Fig. 8.

From Fig. 8, we observe that the ZMP trajectory is always inside the safety region of stability. However, the ZMP tracking error is bigger than that of off-line walking pattern generation. On the other hand, this controller can be applied on-line on account of its low computational complexity.

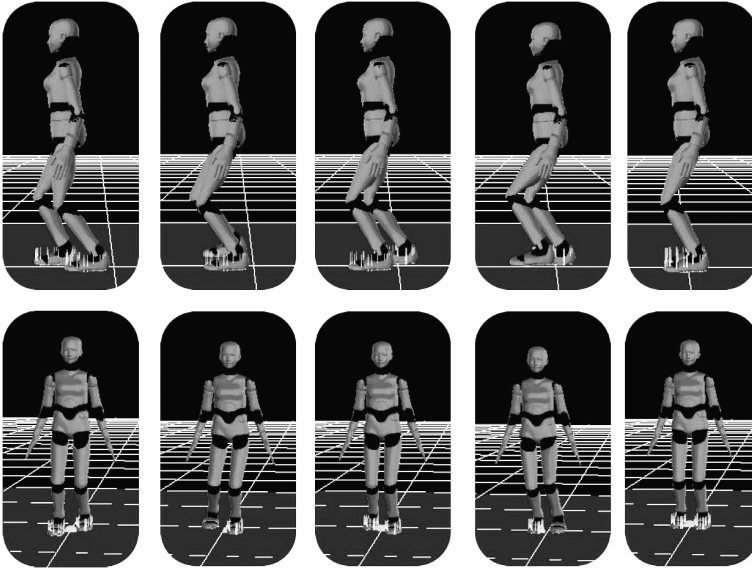


Figure 7. Off-line walking pattern generation algorithm: snapshots of the simulated walking motion.

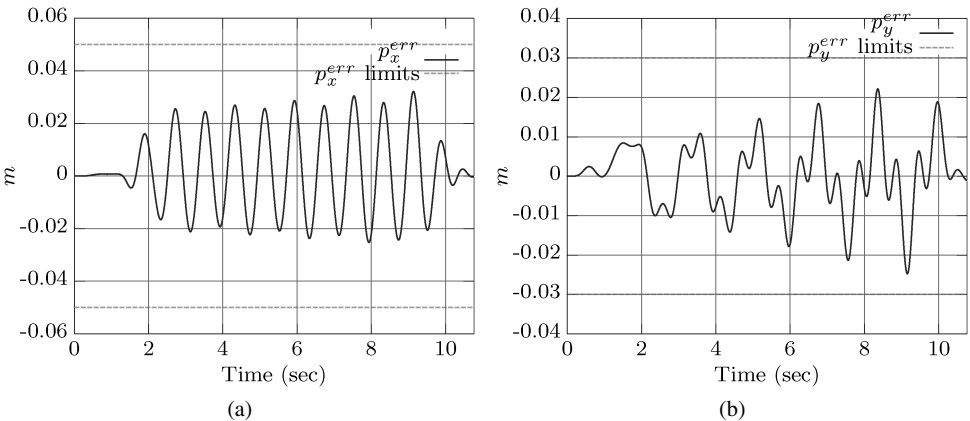


Figure 8. On-line walking pattern generation algorithm I: the ZMP error between the multibody model of the humanoid robot and the planned ZMP trajectory (p^{ref}) (solid line), and the safety margin of stability (dashed line). (a) The x element (p_x^{err}). (b) The y element (p_y^{err}).

Snapshots of the real experiment conducted on the cybernetic human HRP-4C are given in Fig. 9.

5.3.3. On-Line Walking Pattern Generation Algorithm II

Analogously to the previous case, we apply the on-line walking pattern generation algorithm II to generate the CoM trajectory in order to track the desired trajectory of the ZMP. The ZMP error between the multibody model of the humanoid robot and the planned ZMP trajectory (p^{ref}) is given in Fig. 10.

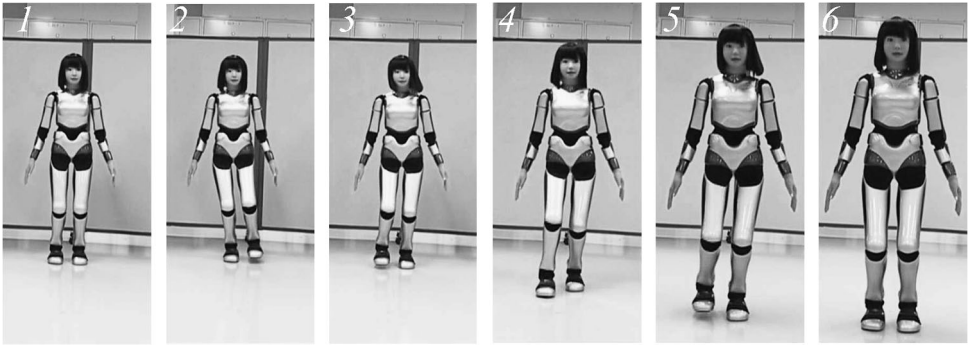


Figure 9. On-line walking pattern generation algorithm I: snapshots of the real experiment using the cybernetic human HRP-4C.

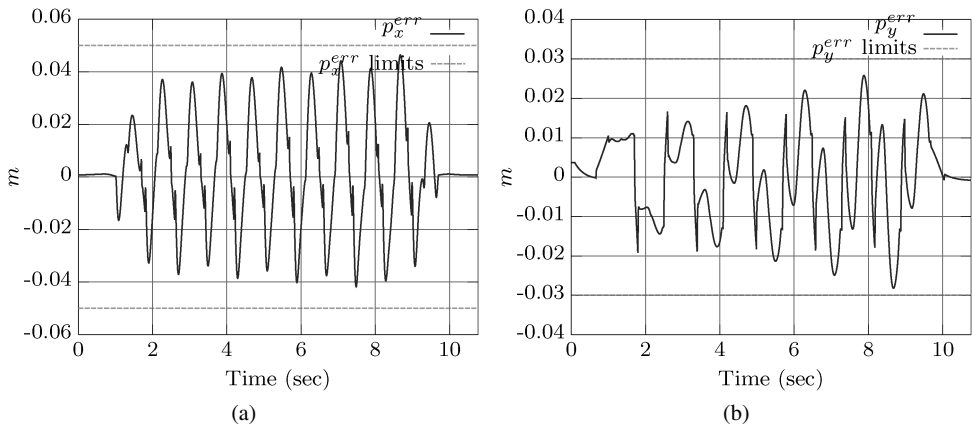


Figure 10. On-line walking pattern generation algorithm II: the ZMP error between the multibody model of the humanoid robot and the planned ZMP trajectory (p^{ref}) (solid line), and the safety margin of stability (dashed line). (a) The x element (p_x^{err}). (b) The y element (p_y^{err}).

From Fig. 10, we observe that the ZMP tracking error becomes bigger. However, the ZMP is always inside the safety region of stability and the dynamic stability of the walking patterns has been verified using the dynamical simulator OpenHRP3.

6. Discussion and Conclusions

In this paper, we proposed a new nonlinear model in order to approximate the dynamic behavior of humanoid robot walking motion. The linear part of the nonlinear model is the cart-table model proposed in Ref. [3]. The nonlinear model is a quadratic system that is a special case of the Volterra series. The main advantage of this model is not only predicting the ZMP trajectory of the humanoid robot without the calculation of the multibody dynamics, but also providing the trajectory of the CoM of the humanoid robot in order to track the planned ZMP trajectory.

To identify the quadratic system, we proposed an adapted identification algorithm. This algorithm is based on using multiple walking patterns in order to identify an accurate model. Moreover, the problem of controller design of the quadratic system has also been investigated. We proposed on-line and off-line walking pattern generation algorithms based on controlling the quadratic system. Even though the off-line walking pattern generation algorithm can track the planned ZMP trajectory precisely, the computational time is not adequate for real-time applications. On the other hand, the on-line walking pattern generation algorithms are devoted to real-time applications and the ZMP trajectory is tracked as much as possible. The conducted simulations using the dynamic simulator OpenHRP3 [20, 21] and the real experiments using the cybernetic human HRP4-C [13] have pointed out that the ZMP trajectory always stays inside the safety region of stability, and the generated walking patterns are dynamically stable.

The actual limitation of the proposed model is that when the walking trajectories are curved, the quadratic system is not accurate enough to capture the ZMP behavior of the humanoid robot. As a result, the generated CoM trajectory might yield an unstable walking motion and the robot might fall down. One reason for this phenomenon is that the input signal of the actual model includes only the jerk functions of the horizontal projections of the CoM. One might think that including the rotational velocities of the CoM in the input signal might yield a more accurate model. This point will be investigated in future work.

Finally, the first obtained results of this challenging problem are encouraging; thus, enhancing the results to be more robust and accurate is the intent of our future work.

Acknowledgements

This research was partially supported by a Grant-in-Aid for Scientific Research from the Japan Society for the Promotion of Science (20-08816).

References

1. S. Kajita and K. Tani, Experimental study of biped dynamic walking in the linear inverted pendulum mode, in: *Proc. IEEE Int. Conf. on Robotics and Automation*, Nagoya, vol. 3, pp. 2885–2891 (1995).
2. N. Naksuk, Y. Mei and C. S. G. Lee, Humanoid trajectory generation: an iterative approach based on movement and angular momentum criteria, in: *Proc. IEEE-RAS/RSJ Int. Conf. on Humanoid Robots*, Los Angeles, CA, vol. 2, pp. 576–591 (2004).
3. 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. on Robotics and Automation*, Taipei, pp. 1620–1626 (2003).
4. T. Sugihara, Y. Nakamura and H. Inoue, Realtime humanoid motion generation through ZMP manipulation based on inverted pendulum control, in: *Proc. IEEE Int. Conf. on Robotics and Automation*, Washington, DC, pp. 1404–1409 (2002).

5. S. Kagami, K. Nishiwaki, T. Kitagawa, T. Sughiara, M. Inaba and H. Inoue, A fast generation method of a dynamically stable humanoid robot trajectory with enhanced ZMP constraint, in: *Proc. IEEE Int. Conf. on Humanoid Robots*, Barcelona, pp. 306–311 (2000).
6. J.-I. Yamaguchi, A. Takanishi and I. Kato, Development of a biped walking robot compensating for three-axis moment by trunk motion, in: *Proc. IEEE Int. Conf. on Intelligent Robots and Systems*, Tokyo, pp. 561–566 (1993).
7. K. Harada, S. Kajita, K. Kaneko and H. Hirukawa, An analytical method on real-time gait planning for a humanoid robot, *J. Humanoid Robotics* **3**, 1–19 (2006).
8. K. Nagasaka, Y. Kuroki, S. Suzuki, Y. Itoh and J. Yamaguchi, Integrated motion control for walking, jumping and running on a small bipedal entertainment robot, in: *Proc. IEEE Int. Conf. on Robotics and Automation*, New Orleans, LA, pp. 3189–3194 (2004).
9. M. Vukobratović and B. Borovac, Zero-moment point—thirty five years of its life, *Int. J. of Humanoid Robotics* **1**, 157–173 (2004).
10. K. Nagasaka, M. Inaba and H. Inoue, Stabilization of dynamic walk on a humanoid using torso position compliance control, in: *Proc. 17th Ann. Conf. of the Robotics Society of Japan*, Hiratsuka, pp. 1193–1194 (1999).
11. T. Sugihara and Y. Nakamura, Precise ZMP manipulation on humanoid robots with acceleration offset, in: *Proc. 22nd Ann. Conf. of the Robotics Society of Japan*, Gifu, p. 3L21 (2004).
12. W. Suleiman and A. Monin, Identification of quadratic system by local gradient search, in: *Proc. IEEE Int. Conf. on Control Applications*, Munich, pp. 2565–2570 (2006).
13. K. Kaneko, F. Kanehiro, M. Morisawa, K. Miura and S. Kajita, Cybernetic human HRP-4C, in: *Proc. IEEE–RAS Int. Conf. on Humanoid Robots*, Paris, pp. 7–14 (2009).
14. W. Suleiman and A. Monin, New method for identifying finite degree Volterra series, *Automatica* **44**, 488–497 (2008).
15. W. Suleiman and A. Monin, Linear multivariable system identification: multi-experiments case, in: *Proc. Conf. on Systems and Control*, Marrakech (2007).
16. K. Kaneko, F. Kanehiro, S. Kajita, H. Hirukawa, T. Kawasaki, M. Hirata, K. Akachi and T. Isozumi, Humanoid robot HRP-2, in: *Proc. IEEE Int. Conf. on Robotics and Automation*, New Orleans, LA, pp. 1083–1090 (2004).
17. K. Kaneko, K. Harada, F. Kanehiro, G. Miyamori and K. Akachi, Humanoid robot HRP-3, in: *Proc. IEEE Int. Conf. on Intelligent Robots and Systems*, Nice, pp. 2471–2478 (2008).
18. Japanese Body Dimension Data, 1997–1998, <http://riodb.ibase.aist.go.jp/dhbodydb/97-98/index.html.en>.
19. W. Suleiman, F. Kanehiro, K. Miura and E. Yoshida, Improving ZMP-based control model using system identification techniques, in: *Proc. IEEE–RAS 9th Int. Conf. on Humanoid Robots*, Paris, pp. 74–80 (2009).
20. F. Kanehiro, H. Hirukawa and S. Kajita, OpenHRP: open architecture humanoid robotics platform, *Int. J. Robot. Res.* **23**, 155–165 (2004).
21. OpenHRP3 Official Site, <http://www.openrtp.jp/openhrp3/en/index.html>.

About the Authors



Wael Suleiman received his Master and PhD degrees in Automatic Control from Paul Sabatier University in Toulouse, France, in 2004 and 2008, respectively. He is now on a Postdoctoral Fellowship of the Japan Society for the Promotion of Science (JSPS) at CNRS-AIST JRL, UMI3218/CRT, National Institute of Advanced Industrial Science and Technology, Tsukuba, Japan. His research interests include nonlinear system identification and control, numerical optimization, and humanoid robots.



Fumio Kanehiro received the BE, ME and PhD degrees, in 1994, 1996 and 1999, respectively, all from the University of Tokyo, Tokyo, Japan. He was a Research Fellow of the Japan Society for the Promotion of Science, from 1999 to 2000, and a Visiting Researcher of LAAS-CNRS, Toulouse, France, in 2007. He is now a Member of the Intelligent Systems Research Institute, National Institute of Advanced Industrial Science and Technology Tsukuba, Japan. His research interests include humanoid robots and developmental software systems.



Kanako Miura received the BE, ME and PhD degrees, in 1999, 2001 and 2004, respectively, all from Tohoku University, Sendai, Japan. She also received the double-degree of the PhD, in 2004, from Louis Pasteur University, Strasbourg, France. She was a Postdoctoral Researcher of Tohoku University, from 2004 to 2005, and a Researcher of NTT DOCOMO, Japan's mobile communications company, from 2005 to 2007. She is now a Member of the Intelligent Systems Research Institute, National Institute of Advanced Industrial Science and Technology Tsukuba, Japan. Her research interests include humanoid robots and motion

generation.



Eiichi Yoshida received the ME and PhD degrees on Precision Machinery Engineering from the Graduate School of Engineering, University of Tokyo, in 1993 and 1996, respectively. In 1996, he joined the former Mechanical Engineering Laboratory, Tsukuba, Japan. He is currently Senior Research Scientist in the Intelligent Systems Research Institute, National Institute of Advanced Industrial Science and Technology (AIST), Tsukuba, Japan. From 1990 to 1991, he was a Visiting Research Associate at the Swiss Federal Institute of Technology at Lausanne. He served as Co-Director of the AIST/IS-CNRS/ST2I Joint French–Japanese Robotics Laboratory (JRL) at LAAS-CNRS, Toulouse, France, from 2004 to 2008. He is currently Co-Director of CNRS-AIST JRL, UMI3218/CRT, AIST, Japan, since 2009. His research interests include robot task and motion planning, modular robotic systems, and humanoid robots.