Doctoral Thesis

Study on Biped Walking Robot with Environmental Force Interaction

(環境との相互作用を考慮した２足歩行ロボットの研究)

Yasutaka Fujimoto

Division of Electrical and Computer Engineering
Graduate School of Engineering
Yokohama National University

Supervisor Professor Atsuo Kawamura

March 1998

Copyright © 1998 by Yasutaka Fujimoto

All rights reserved. Published 1998.
No part of this thesis may be reproduced in any form,
nor may it be stored in a retrieval system or transmitted
in any form, without written permission from the author.

Printed in Japan

ACKNOWLEDGMENTS

This research was accomplished under the orientation of Professor Atsuo KAWAMURA, towards who I am particularly grateful. He gave me an opportunity for absorption in this work.

During more than 6 years, Professor Kawamura gave me advice, valuable instructions, technical support and encouragement to realize this job. His contribution for my research, and also his cordial help for all the problems I encountered during my course, deserve my deepest thanks.

All the members, staffs, students, and researchers of Kawamura Laboratory, are also gratefully acknowledged. They constantly helped me in my work.

Finally, I would like to note that part of this research is carried out with a subsidy of the Scientific Research Fund of the Ministry of Education.

Yasutaka Fujimoto

論文要旨

歩行ロボットはその非常に優れた機動性から、さまざまな特殊用途への応用が期待できる。具体的には、不整地での旅客貨物運搬、災害救助、人間の立ち入ることのできない危険区域での作業などが挙げられる。特に 2 足で歩行可能な人間型ロボットは、これまで人間しか出来なかったような複雑な作業を行なう、汎用性のある自動化機器となる可能性を秘めている。最も利用価値の高い応用として、高齢化社会における老人介護や、一般家庭での家事サービスなどが考えられる。2 足歩行ロボットの研究は、そのような人間型ロボットの実現に必要な基礎研究の一つである。

本研究では、脚と床との接触力を直接制御することで単脚支持状態を安定化する歩行システムを提案した。本手法では、直接、接触の安定条件を考慮することができるだけでなく、ロバストな力制御系により床の動特性の不確定さの影響を受け難くすることができる。具体的には、

1. 環境との相互作用力を考慮した階層制御システム
   (a) ロボットの姿勢による慣性変動に不感な関節ロバストサーボ制御システム
   (b) ロバストな足底床反力・反トルク制御システム
   (c) 足底床反力・反トルクの物理的制約を考慮した姿勢制御システム
   (d) 離散時間倒立振子モデルを用いた脚交換による実時間重心追従制御システム
2. 環境との接触力を厳密に考慮した 3 次元運動シミュレーション手法

を提案した。本手法では環境との相互作用を直接制御するため、外乱に強いロバストな 2 足歩行が実現できる。提案する手法を 20 軸人間型ロボットに適用し、速度 0.25 m/sec、周期 0.5 sec/step の歩行を実現した。

14 軸 2 足ロボットの実験により、簡略化した姿勢制御の有効性を示した。重心位置の制御誤差が ± 0.03[m]、上体姿勢の制御誤差が ± 0.04 [rad] $(= \pm 2.3$ [deg]) 以内に安定に制御できた。

ABSTRACT

This paper describes an autonomous biped walking control system based on the reactive force interaction at the foothold using the precise 3D (three dimensional) dynamic simulation. The contents of this research are as follows.

1. A hierarchical control system considering environmental force interaction
   (a) An inertia fluctuation insensitive robust servo control system in joint space
   (b) A robust force/torque control system in Cartesian space
   (c) A posture controller considering the physical constraints of the reactive force/torque on the foot by quadratic programming
   (d) A real-time COM (center of mass) tracking controller by the foot placement with a discrete inverted pendulum model
2. The 3D dynamic simulation scheme with precise contact with the environment.

The proposed approach realizes the robust biped locomotion because the environmental interaction is directly controlled. The proposed autonomous control system is applied to the 20 axes simulation model, and the stable biped locomotion with a velocity 0.25 m/sec and a stepping time 0.5 sec/step is realized.

In the experiments of the 14 axes biped robot, the stable attitude control of the body of the biped robot is realized. The position of COM and the attitude of the body is well controlled within ± 0.03 [m] and ± 0.04 [rad] ($= \pm$ 2.3 [deg]) errors.

Contents

ACKNOWLEDGMENTS iii

論文要旨 v

ABSTRACT vii

1 Introduction 1

2 Modeling of Biped Robot 5
2.1 Dynamic Equation 5

3 Hierarchical Control 7
3.1 Robust Servo Control in Joint Space 8
3.1.1 Plant Model 8
3.1.2 Robust Control Considering Inertia Fluctuation 9
3.1.3 Calculation of Inertia Matrix 13
3.1.4 Unified Controller in Joint Space 14
3.2 Inverse Kinematics 14
3.3 Robust Force Control in Environment 16
3.4 Posture Control with Optimal Force Distribution 17
3.5 Yaw Moment Compensation by Arm Motion 21
3.6 Globally Stable Walking Pattern Generation 22

4 Dynamic Simulator of Legged Robot 25
4.1 Forward Dynamic Simulator 25
4.1.1 Numerical Integration 25
4.1.2 Exact Model of Legged Robot 26
4.1.3 Exact Model of Environmental Force Interaction 30
4.1.4 Procedure of Forward Dynamic Simulation 33
4.2 Inverse Dynamic Simulator 34
4.2.1 Interaction Force Calculation 34
4.2.2 Procedure of Inverse Dynamic Simulation 35

5 Simulation Results 37
5.1 21-link Biped Walking 37
5.2 Walking on Unknown Slope 42

5.3 Walking on Low μ Terrain 45

6 Experiments **49**
6.1 Hardware Design 49
6.1.1 Specifications 49
6.1.2 Actuators 53
6.2 Implementation Aspects 54
6.2.1 Robust Servo Control in Joint Space 54
6.2.2 Force Control in Environment 57
6.2.3 Posture Control with Optimal Force Distribution 59

7 Discussion **67**
7.1 Feedback or Feedforward? 67
7.2 On the Honda's Humanoid Robot 67

8 Conclusion **69**

BIBLIOGRAPHY **70**

LIST OF FIGURES **75**

LIST OF TABLES **77**

APPENDIX **78**

A Related Publications **81**

Chapter 1

Introduction

Since a legged robot has a very efficient mobility, it is expected to apply in various industrial uses, for example, transportation of passengers and cargos on a rough terrain, disaster relief, operation in a danger zone, etc. Especially, a humanoid type robot which can walk by its own two legs has potential for a substitute for a part of human work such as household matters or aid of elderly people because the biped form is available in any environment where the human can go. The biped form is the best for the human living environment. The biped machine locomotion is one of fundamental technology to realize the humanoid. But technique of the biped robot is still under development. The reasons are because the biped system is

- unstable,
- non-linear,
- and subject to restrictions caused by the environment.

Concerning the third point of these characteristics, it is difficult to stabilize the biped robot when dynamics of the environment is unknown.

In detail, Fig. 1.1 illustrates a biped robot whose foot interacts to the ground and whose body receives the reactive force. The movement of the robot depends on the reactive force. This system is expressed as the block diagram shown in Fig. 1.2. The legged robot can be separated to two parts, the body and the legs. The controller calculates the actuator torque from joint angle and another sensor information, then the legs are moved. The legs interact to the environment, which generates the reactive force. Then the body is driven by the reactive force.

In this system, dynamics of the legged robot has restrictions because it does not have fixed point on the ground and there are limitations of the force and the torque on the foothold. This point is different from a case of ordinary robot manipulators. Despite the information of the reactive force on the foot is very important, it is hardly used as a main index in the feedback control loop in the previous works. It is mostly used as a sub-index in the control loop[5] or a main index in the reference generation[41]. The conventional control systems calculate trajectories of joint angle or joint torque so as to approximately satisfy the stable contact condition. The whole dynamic equation of the robot and the contact condition is considered to generate joints references in [41], but it is off-line type planning due to the complexity of dynamics of the biped robot. Kajita et'al have proposed an linear inverted pendulum model[13]. The direct control of zero moment point is proposed in [36].

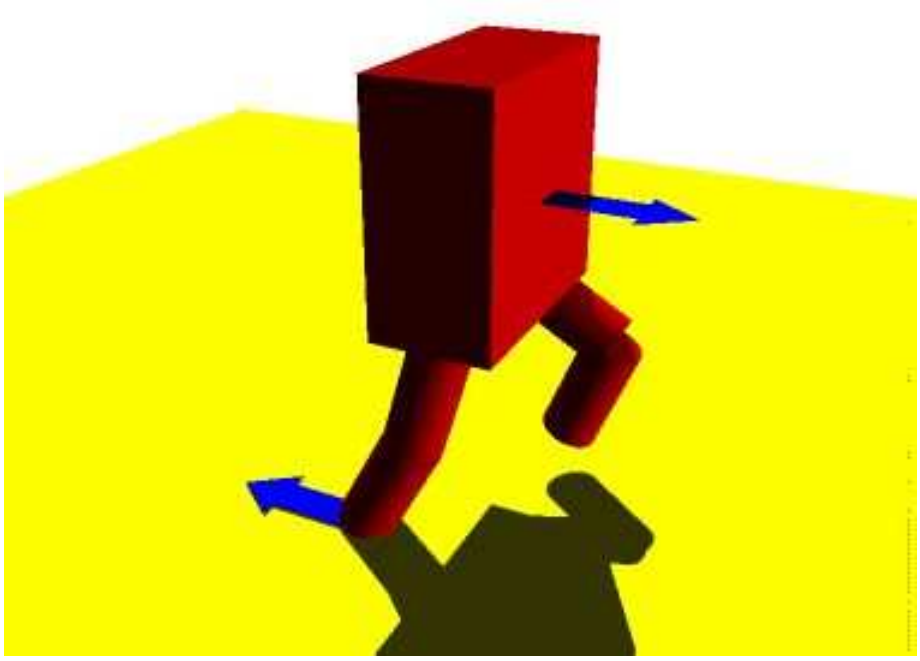


Figure 1.1: An illustration of a legged system.

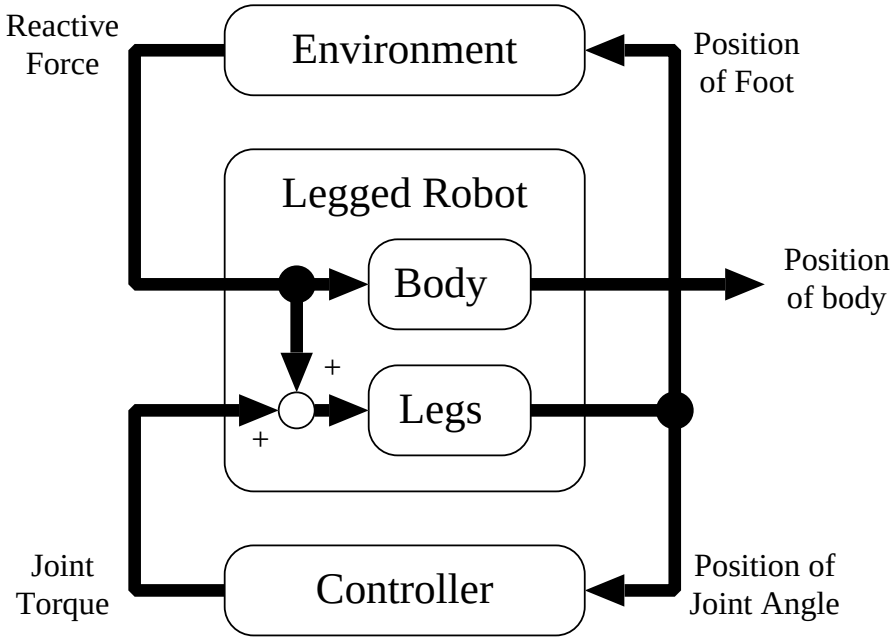


Figure 1.2: A block diagram of the legged system.

These approaches can be said to be model based control. Thus there are the modeling errors in the control systems, which yields lack of walking robustness. One of the solutions is in use of an adaptive method [18][42][14][31][4].

The other solution is a robust method. In this paper, a new hierarchical control system based on the reactive force control on the foothold and the force distribution system is proposed in order to improve the walking robustness, in which the physical constraints of the contact force on the foothold are precisely considered. The robust force controller of the support foot can locally suppresses unknown disturbances on the terrain. With a new algorithm of the foot placement, an on-line controlled autonomous biped locomotion is realized. Also, the precise 3D dynamic simulator with the environmental interaction is proposed to investigate the control scheme.

Chapter 2

Modeling of Biped Robot

2.1 Dynamic Equation

A legged robot is modeled as a free-fall manipulator which has no fixed-point but has interaction to the ground. Dynamics of a free-falling manipulator is formulated by introducing the variables representing position and attitude of a *base-link*. Let generalized coordinates $\boldsymbol{x}$, generalized velocities $\boldsymbol{v}$, and generalized forces $\boldsymbol{u}$ be

$$\boldsymbol{x}^T = \left[\boldsymbol{p}_B^T, \boldsymbol{A}_B^T, \boldsymbol{\theta}^T\right] \in R^3 \times SO(3) \times R^N \tag{2.1}$$

$$\boldsymbol{v}^T = \left[\boldsymbol{v}_B^T, \boldsymbol{\omega}_B^T, \boldsymbol{\omega}^T\right] \in R^3 \times R^3 \times R^N \tag{2.2}$$

$$\boldsymbol{u}^T = \left[\boldsymbol{f}_B^T, \boldsymbol{n}_B^T, \boldsymbol{\tau}^T\right] \in R^3 \times R^3 \times R^N \tag{2.3}$$

where

$\boldsymbol{p}_B$: 3×1 vector specifying base-link position
$\boldsymbol{A}_B$: 3×3 matrix specifying base-link attitude
$\boldsymbol{\theta}$: $N \times 1$ vector specifying joint angle
$\boldsymbol{v}_B$: 3×1 vector specifying base-link velocity
$\boldsymbol{\omega}_B$: 3×1 vector specifying angular velocity of base-link
$\boldsymbol{\omega}$: $N \times 1$ vector specifying joint angular velocity
$\boldsymbol{f}_B$: 3×1 force vector generated in base-link
$\boldsymbol{n}_B$: 3×1 torque vector generated in base-link
$\boldsymbol{\tau}$: $N \times 1$ torque vector generated by actuator
N : number of joints of robot

$\boldsymbol{A}_B$ is the direction matrix of the inertial base-link-fixed axes relative to the ground-fixed axes. (See Fig. 2.1, $\boldsymbol{A}_B = [\boldsymbol{x}_B, \boldsymbol{y}_B, \boldsymbol{z}_B]$.) The attitude $\boldsymbol{A}_B$ moves in the Lie group $SO(3)$.

The equations of motion of the robot become:

$$\dot{\boldsymbol{p}}_B = \boldsymbol{v}_B \tag{2.4}$$

$$\dot{\boldsymbol{A}}_B = \boldsymbol{\omega}_B \times \boldsymbol{A}_B \tag{2.5}$$

$$\dot{\boldsymbol{\theta}} = \boldsymbol{\omega} \tag{2.6}$$

and

$$\boldsymbol{H}(\boldsymbol{x})\dot{\boldsymbol{v}} + \boldsymbol{C}(\boldsymbol{x},\boldsymbol{v})\boldsymbol{v} + \boldsymbol{g}(\boldsymbol{x}) = \boldsymbol{u} + \boldsymbol{u}_E \tag{2.7}$$

where

$\boldsymbol{H}(\boldsymbol{x})$: $(N+6)\times(N+6)$ inertia matrix
$\boldsymbol{C}(\boldsymbol{x},\boldsymbol{v})$: $(N+6)\times(N+6)$ matrix specifying centrifugal and Corioli's effects
$\boldsymbol{g}(\boldsymbol{x})$: $(N+6)\times 1$ vector specifying gravity effect
$\boldsymbol{u}_E$: $(N+6)\times 1$ vector specifying generalized forces generated by external forces

The detailed generalized external forces $\boldsymbol{u}_E$ is given in the following sections, on which there are two type of formulations. One is for the control and the other is for the simulation.

Equation (2.7) represents a general form of the dynamic equation.

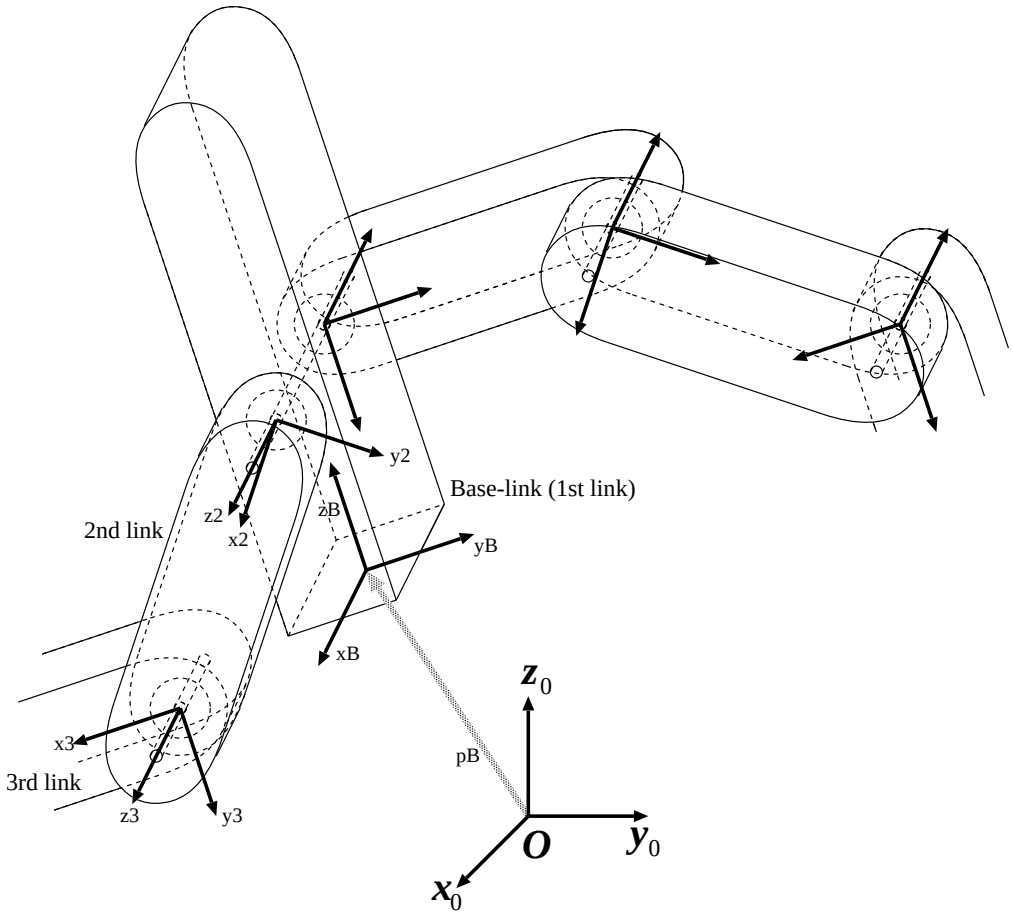


Figure 2.1: Representation of link-fixed coordinates.

Chapter 3

Hierarchical Control

To control a legged robot (2.4)–(2.7), we must consider physical constraints on the foothold directly or indirectly. In this paper, a hierarchical control system with a direct and real-time method is proposed, which realizes the robust contact of the foothold and the stable biped locomotion. The overview of the proposed hierarchical control system is shown in Fig. 3.1.

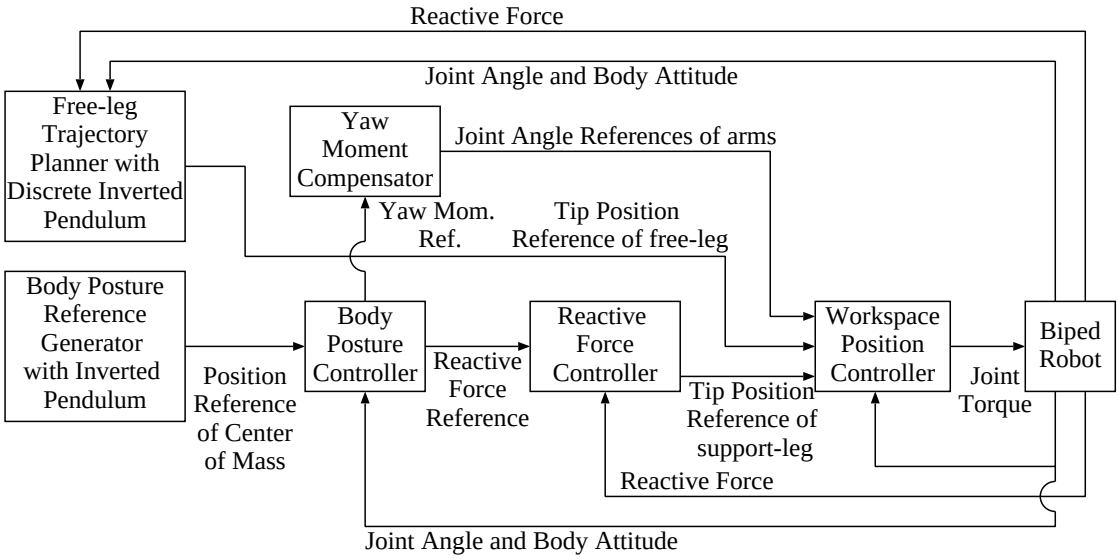


Figure 3.1: Biped walking control system.

The block of "Workspace Position Controller" represents the 6-degree-of-freedom tip position controller of each foot with respect to the body link, which consists of a robust joint control with inertia fluctuation insensitivity in the section 3.1 and an inverse kinematics by Newton-Raphson method in the section 3.2.

The block of "Reactive Force Controller" represents the 6-degree-of-freedom reactive force controller of each foot, which is obtained by robust servo design technique in the section 3.3.

The block of "Body Posture Controller" represents the 6-degree-of-freedom position and attitude controller of the body considering the physical condition (zero moment point condition,

etc.) on the reactive force, which is shown in the section 3.4.

The block of "Yaw Moment Compensator" shows the controller of yaw axis rotation of the body attitude using arm swing motion, which is introduced in the section 3.5.

The block of "Body Posture Reference Generator" represents the posture reference generator of the body based on the inverted pendulum model, which is shown in the section 3.6.

Also the block of "Free-leg Trajectory Planner" represents the trajectory generator of free-leg based on the discrete inverted pendulum model, which stabilize the global biped locomotion (shown in 3.6.)

3.1 Robust Servo Control in Joint Space

The values of the inertia of the robot vary in real-time depending on the robotic posture. Moreover, the fluctuation is widely depending on the environmental constraints. For example, the inertia of the ankle joint of the legged robot is small in case that the leg is in non-support phase, which becomes big in case of support phase. The inertia fluctuation causes unstability of the joint servo system.

In this section a new robust control of robotic system is proposed, in which the robust stability against fluctuation of inertia matrix depending on the posture is guaranteed.

3.1.1 Plant Model

According to the generalized coordinates, the biped robot (2.7) can be decomposed into three parts as follows:

$$\begin{bmatrix} \boldsymbol{H}_{11} & \boldsymbol{H}_{12} & \boldsymbol{H}_{13} \\ \boldsymbol{H}_{21} & \boldsymbol{H}_{22} & \boldsymbol{H}_{23} \\ \boldsymbol{H}_{31} & \boldsymbol{H}_{32} & \boldsymbol{H}_{33} \end{bmatrix} \begin{bmatrix} \dot{\boldsymbol{v}}_B \\ \dot{\boldsymbol{\omega}}_B \\ \ddot{\boldsymbol{\theta}} \end{bmatrix} + \begin{bmatrix} \boldsymbol{b}_1 \\ \boldsymbol{b}_2 \\ \boldsymbol{b}_3 \end{bmatrix} = \begin{bmatrix} \mathbf{o} \\ \mathbf{o} \\ \boldsymbol{\tau} \end{bmatrix} + \begin{bmatrix} \boldsymbol{u}_{E1} \\ \boldsymbol{u}_{E2} \\ \boldsymbol{u}_{E3} \end{bmatrix} \tag{3.1}$$

where $[\boldsymbol{b}_1^T \ \boldsymbol{b}_2^T \ \boldsymbol{b}_3^T]$ represents biasing vector

$$\begin{bmatrix} \boldsymbol{b}_1^T & \boldsymbol{b}_2^T & \boldsymbol{b}_3^T \end{bmatrix}^T = \boldsymbol{b} = \boldsymbol{C}(\boldsymbol{x}, \boldsymbol{v})\boldsymbol{v} + \boldsymbol{g}(\boldsymbol{x}). \tag{3.2}$$

To control joint angle, we pay attention to the 3rd row of the biped robot (3.1) which corresponds to plant dynamics in joint space:

$$\boldsymbol{H}_{31}\dot{\boldsymbol{v}}_B + \boldsymbol{H}_{32}\dot{\boldsymbol{\omega}}_B + \boldsymbol{H}_{33}\ddot{\boldsymbol{\theta}} + \boldsymbol{b}_3 = \boldsymbol{\tau} + \boldsymbol{u}_{E3}. \tag{3.3}$$

(3.3) can be transformed as

$$\ddot{\boldsymbol{\theta}} = \boldsymbol{H}_{33}^{-1}(\boldsymbol{\tau} - \boldsymbol{d}) \tag{3.4}$$

where

$$\boldsymbol{d} = \boldsymbol{H}_{31}\dot{\boldsymbol{v}}_B + \boldsymbol{H}_{32}\dot{\boldsymbol{\omega}}_B + \boldsymbol{b}_3 - \boldsymbol{u}_{E3}. \tag{3.5}$$

Fig. 3.2 shows the system (3.4) where $\boldsymbol{H} = \boldsymbol{H}_{33}$ in this section.

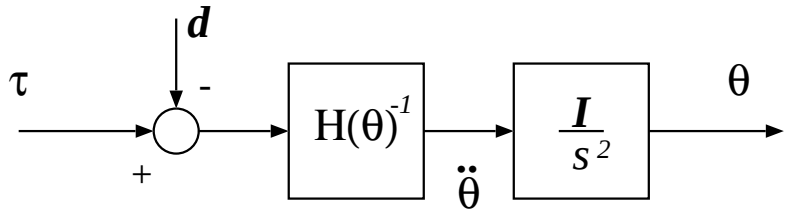


Figure 3.2: Plant system in joint space.

3.1.2 Robust Control Considering Inertia Fluctuation

In order to realize a decoupled system, we introduce a virtual input $\boldsymbol{\tau}'$ as

$$\boldsymbol{\tau} = \boldsymbol{H}_n(\boldsymbol{\theta})\boldsymbol{\tau}' \tag{3.6}$$

where $\boldsymbol{H}_n(\boldsymbol{\theta})$ denotes a nominal inertia matrix which includes the fluctuation depending on the posture and the inertia products of the non-diagonal components. It is shown in Fig. 3.3. In this case (3.4) is equivalent to

$$\begin{aligned} \ddot{\boldsymbol{\theta}} &= \boldsymbol{H}(\boldsymbol{\theta})^{-1}\boldsymbol{H}_n(\boldsymbol{\theta})[\boldsymbol{\tau}' - \boldsymbol{d}'(\boldsymbol{\theta}, \dot{\boldsymbol{\theta}})] \\ &= (\boldsymbol{I} + \boldsymbol{\Delta})[\boldsymbol{\tau}' - \boldsymbol{d}'(\boldsymbol{\theta}, \dot{\boldsymbol{\theta}})] \end{aligned} \tag{3.7}$$

where

$$\boldsymbol{d}'(\boldsymbol{\theta}, \dot{\boldsymbol{\theta}}) = \boldsymbol{H}_n(\boldsymbol{\theta})^{-1}\boldsymbol{b}(\boldsymbol{\theta}, \dot{\boldsymbol{\theta}}) \tag{3.8}$$

$$\boldsymbol{\Delta} = \boldsymbol{H}(\boldsymbol{\theta})^{-1}\boldsymbol{H}_n(\boldsymbol{\theta}) - \boldsymbol{I}. \tag{3.9}$$

Here $\boldsymbol{\Delta}$ means multiplicative perturbations of the virtual plant.

Considering the virtual plant system in Fig. 3.3, equivalently we can obtain a decoupled system of double integrator as shown in Fig. 3.4.

Design of H_∞ Robust Control System

In this section, a design method of H_∞ robust control system for the virtual plant (3.7) is shown. There are various design methods of linear robust control systems for robot manipulators such as *Disturbance Observer*[25][17][24], *Two-Degree-of-Freedom Control*[37][33], and H_∞ *Control*[27]. The H_∞ Control is most suitable for a consideration of robust stability. The system remain stable even if there are plant perturbations.

Consider designing a linear controller $\boldsymbol{C}(s)$ of the virtual plant (3.7), which makes the system insensitive to $\boldsymbol{d}'$ and stable for the parameter fluctuation $\boldsymbol{\Delta}$, as shown in Fig. 3.5. In this method, the non-linear terms $\boldsymbol{d}'$ including friction, centrifugal, and Corioli's force, which are difficult to model accurately, are suppressed by the linear controller $\boldsymbol{C}(s)$. This approach is different from the non-linear compensation called the *Computed Torque Method*.

Suppose the maximum singular value of the multiplicative plant perturbations $\boldsymbol{\Delta}$ is smaller than some known value λ;

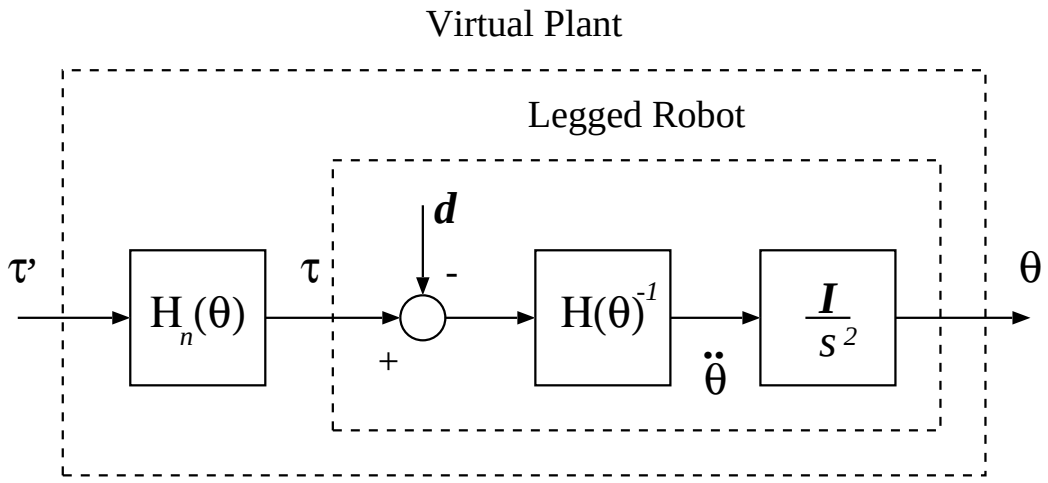


Figure 3.3: Virtual plant.

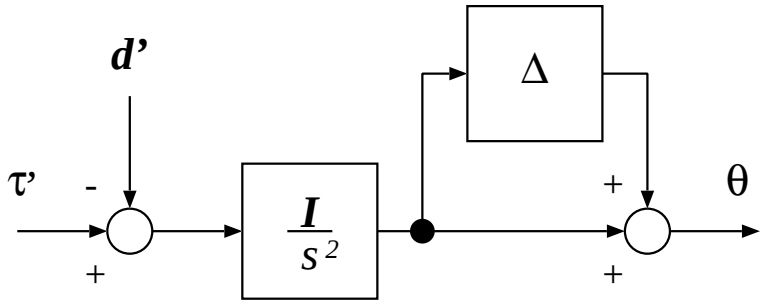


Figure 3.4: Equivalent system of virtual plant.

$$\bar{\sigma}(\boldsymbol{\Delta}) \leq \lambda. \tag{3.10}$$

Against the perturbations (3.10), the system becomes stable with a controller $\boldsymbol{C}(s)$ which satisfies

$$\bar{\sigma}(\boldsymbol{T}(s)) \leq |W_T(s)^{-1}| \tag{3.11}$$

where $W_T(s)$ denotes a scalar transfer function such as

$$\lambda \leq |W_T(j\omega)| \tag{3.12}$$

and $\boldsymbol{T}$ denotes the complementary sensitivity function defined as

$$\boldsymbol{T} = \boldsymbol{PC}(\boldsymbol{I} + \boldsymbol{PC})^{-1}. \tag{3.13}$$

$\boldsymbol{P}$ also denotes the nominal virtual plant

$$\boldsymbol{P} = \frac{\boldsymbol{I}}{s^2}. \tag{3.14}$$

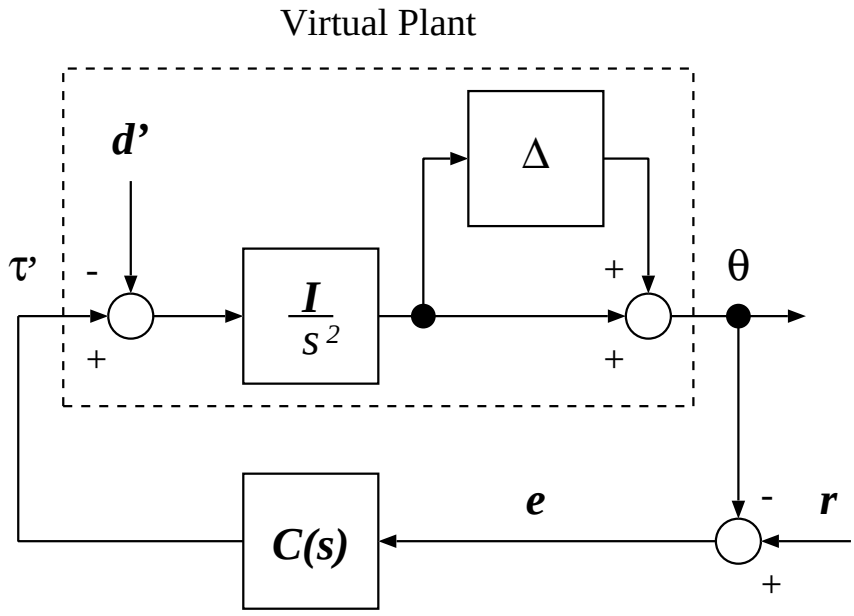


Figure 3.5: Control system.

In the design of the controller, the disturbance attenuation must be also considered. Since the closed-loop transfer function from the disturbance $\boldsymbol{d}'$ to the output $\boldsymbol{\theta}$ is $(\boldsymbol{I}+\boldsymbol{PC})^{-1}\boldsymbol{P}$, the sensitivity function $\boldsymbol{S}$ defined as

$$\boldsymbol{S} = (\boldsymbol{I}+\boldsymbol{PC})^{-1} \tag{3.15}$$

determine the disturbance attenuation. Thus we introduce a scalar transfer function of a disturbance attenuation factor with a specification such as

$$\bar{\sigma}(\boldsymbol{S}(j\omega)) \leq \gamma |W_S(j\omega)^{-1}|. \tag{3.16}$$

Then the problem is to find the controller which satisfies (3.11) and (3.16) with $\gamma(>0)$ as small as possible. This is known as the *Mixed-Sensitivity Approach.*

Since sensitivity function $\boldsymbol{S}(s)$ corresponds to the transfer function from the reference $\boldsymbol{r}$ to the error $\boldsymbol{e}$, the problem is equivalent to finding the controller such as

$$||\boldsymbol{G}_{zr}||_\infty \leq 1 \tag{3.17}$$

where $\boldsymbol{G}_{zr}$ is the transfer function from $\boldsymbol{r}$ to $\boldsymbol{z} = [\ \boldsymbol{z}_1 \quad \boldsymbol{z}_2\ ]^T$ on the following augmented plant of the augmented system shown in Fig. 3.6.

$$\left[\begin{array}{c} \boldsymbol{z}_1 \\ \boldsymbol{z}_2 \\ \hline \boldsymbol{e} \end{array}\right] = \left[\begin{array}{c|c} \frac{1}{\gamma}W_S\boldsymbol{I} & -\frac{1}{\gamma}W_S\boldsymbol{I}/s^2 \\ \mathbf{o} & W_T\boldsymbol{I}/s^2 \\ \hline \boldsymbol{I} & -\boldsymbol{I}/s^2 \end{array}\right] \left[\begin{array}{c} \boldsymbol{r} \\ \hline \boldsymbol{\tau}' \end{array}\right] \tag{3.18}$$

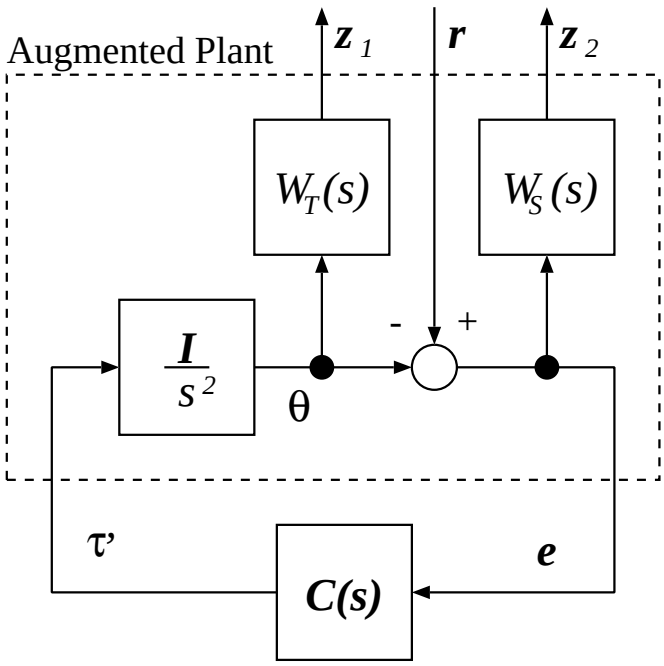


Figure 3.6: Augmented system.

Here, we can treat the *MIMO* (Multi Input Multi Output) system design problem as the *SISO* (Single Input Single Output) system design problem, since the augmented plant Fig. 3.6 is a decoupled system. Thus we need to find the controller $C(s)$ of the SISO subsystem

$$\begin{aligned} \left[\begin{array}{c} z_{1i} \\ \hline z_{2i} \\ \hline e_i \end{array}\right] &= \left[\begin{array}{c|c} \frac{1}{\gamma}W_S & -\frac{1}{\gamma}W_S/s^2 \\ \hline 0 & W_T/s^2 \\ \hline 1 & -1/s^2 \end{array}\right] \left[\begin{array}{c} r_i \\ \hline u_i' \end{array}\right] \\ i &= 1, \ldots, N \end{aligned} \tag{3.19}$$

subject to the constraints

$$||\boldsymbol{G}_{zri}||_\infty = |||[z_{1i} z_{2i}]^T / r_i||_\infty \leq 1. \tag{3.20}$$

Then the H_∞ optimal controller of the MIMO system $\boldsymbol{C}(s)$ is obtained by

$$\boldsymbol{C}(s) = \mathrm{diag}\{C(s), C(s), \ldots, C(s)\}. \tag{3.21}$$

The following is the design procedure of the H_∞ controller of the subsystem (3.19).

When the plant has the pole on the imaginary axis, it is not a standard H_∞ control problem. The simple loop transformation[23] can deal with the problem as follows. Consider the factorization of the $j\omega$ pole of the plant in to a scalar transfer function $a(s)$ such as

$$P(s) = \widehat{P}(s) a(s). \tag{3.22}$$

Then the H_∞ controller of the new plant $\widehat{P}(s)$ which does not have $j\omega$ pole is the modified controller $\widehat{C}(s)$, in which $a(s)$ is inserted into the original controller. The original controller is given by

$$C(s) = a(s)^{-1}\widehat{C}(s). \tag{3.23}$$

Now the plant is the double integrator then let $a(s) = (s+1)^2/s^2$. The cost functions are selected as

$$W_S{}^{-1} = \frac{s^3}{\omega_c^3} \tag{3.24}$$

$$W_T{}^{-1} = \frac{\omega_c^2}{\lambda(s+\omega_c)^2} \tag{3.25}$$

where W_S is chosen to have triple origin poles against the double origin poles $1/s^2$ of the plant and steady disturbance $1/s$. ω_c denotes the cut-off frequency.

Fig. 3.7 shows the frequency characteristics of the cost functions, the sensitivity function, and the complementary sensitivity function in a case of $\omega_c = 1100[rad/s]$ and $\lambda = 0.9$. The design was done with a CAD tools $MATLAB^{TM}$[1]. Finally, 430 [rad/s] cut-off frequency of the sensitivity function was obtained through the γ-iteration.

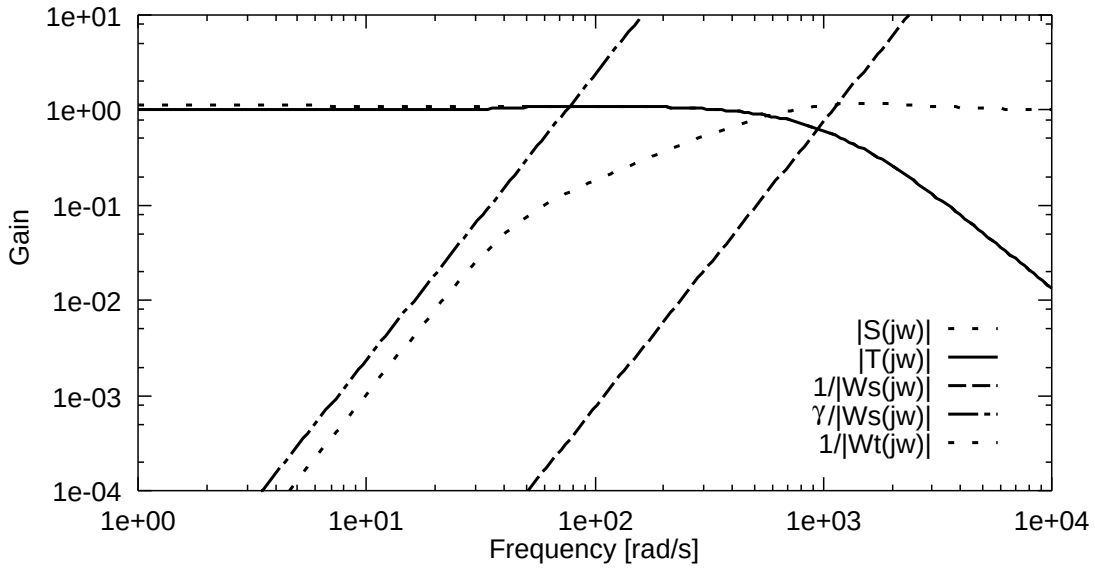


Figure 3.7: Frequency characteristics of cost functions, sensitivity function, and complementary sensitivity function

3.1.3 Calculation of Inertia Matrix

According to the relation of (3.6), the control input of H_∞ controller for the virtual plant $\boldsymbol{\tau}' = [\ \tau_1' \ \ \tau_2' \ \ \cdots \ \ \tau_N' \ ]^T$ must be transformed into the actual control input $\boldsymbol{\tau}$. In fact, the calculation of the matrix $\boldsymbol{H}_n(\boldsymbol{q})$ does not need directory. The transformation is equivalent to the calculation of the joint torque in a case that the all joints are moving with the acceleration $\boldsymbol{\tau}'$. This calculation is done by inverse dynamics shown in (4.8)–(4.26) of the next chapter with setting $\dot{\boldsymbol{v}}_B = 0$, $\dot{\boldsymbol{\omega}}_B = 0$, $\boldsymbol{\omega}_B = 0$, $\ddot{q}_i = \tau_i'$, $\dot{q}_i = 0$ and q_i to current state for $1 \le i \le N$.

In this paper the inertia calculation is called *Inertia Torque Computation Filter*

3.1.4 Unified Controller in Joint Space

The configuration of the proposed joint control system is shown in Fig. 3.8.

$$\boldsymbol{\tau} = \boldsymbol{H}_n(\boldsymbol{\theta})\boldsymbol{C}(s)(\boldsymbol{r} - \boldsymbol{\theta}). \tag{3.26}$$

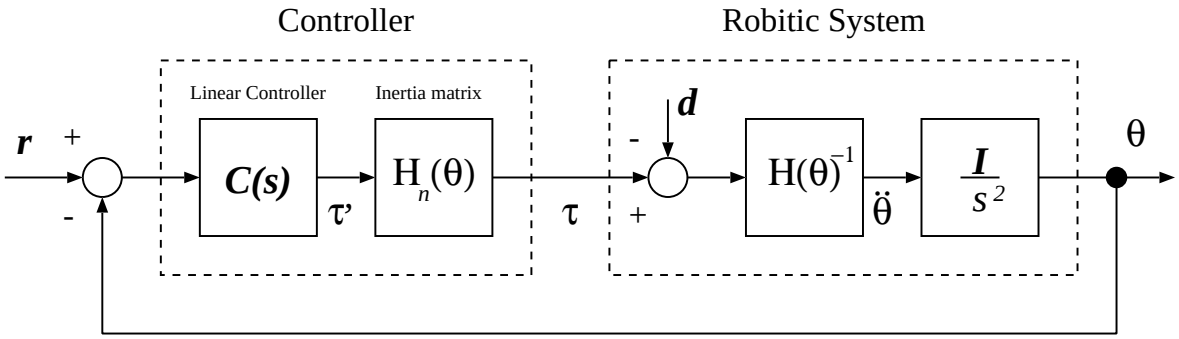


Figure 3.8: Global control system.

3.2 Inverse Kinematics

To control the tip positions and directions of both feet, those references in the workspace are transformed into joint angle references by the inverse kinematics calculation as follows.

Fig. 3.9 shows kinematic variables, where $\boldsymbol{p}_B$ denotes the position of the body with respect to the origin of the world coordinates system and $\boldsymbol{x}_R$ denotes the position of the right foot with respect to the origin of the body-fixed coordinates system; $\boldsymbol{n}_R$, $\boldsymbol{s}_R$, and $\boldsymbol{a}_R$ are the unit normal, unit slide, and unit approach vectors of the right foot, respectively. Fig. 3.9 shows only a right foot case. The left foot has also same kinematic properties with the variables replacing the subscript "R" with "L".

Let $\boldsymbol{A}_R = [\boldsymbol{n}_R\ \boldsymbol{s}_R\ \boldsymbol{a}_R]$. Suppose the references $\boldsymbol{x}_{ref} = (\boldsymbol{x}_{Rref},\ \boldsymbol{A}_{Rref})$ is given by the upper layer of the controller (shown in the next section). Then the position and orientation error vectors become

$$\boldsymbol{e}_{Rp} = \boldsymbol{x}_{Rref} - \boldsymbol{x}_R \tag{3.27}$$

$$\boldsymbol{e}_{Ro} = \frac{1}{2}(\boldsymbol{n}_R \times \boldsymbol{n}_{Rref} + \boldsymbol{s}_R \times \boldsymbol{s}_{Rref} + \boldsymbol{a}_R \times \boldsymbol{a}_{Rref}). \tag{3.28}$$

The joint angle reference $\boldsymbol{r}$ can be obtained by Newton-Raphson method with update law

$$\boldsymbol{r} \leftarrow \boldsymbol{r} + \boldsymbol{J}_R(\boldsymbol{r})^{-1}\boldsymbol{e}_R(\boldsymbol{r}) \tag{3.29}$$

where

$$\boldsymbol{e}_R = \begin{bmatrix} \boldsymbol{e}_{Rp} \\ \boldsymbol{e}_{Ro} \end{bmatrix} \tag{3.30}$$

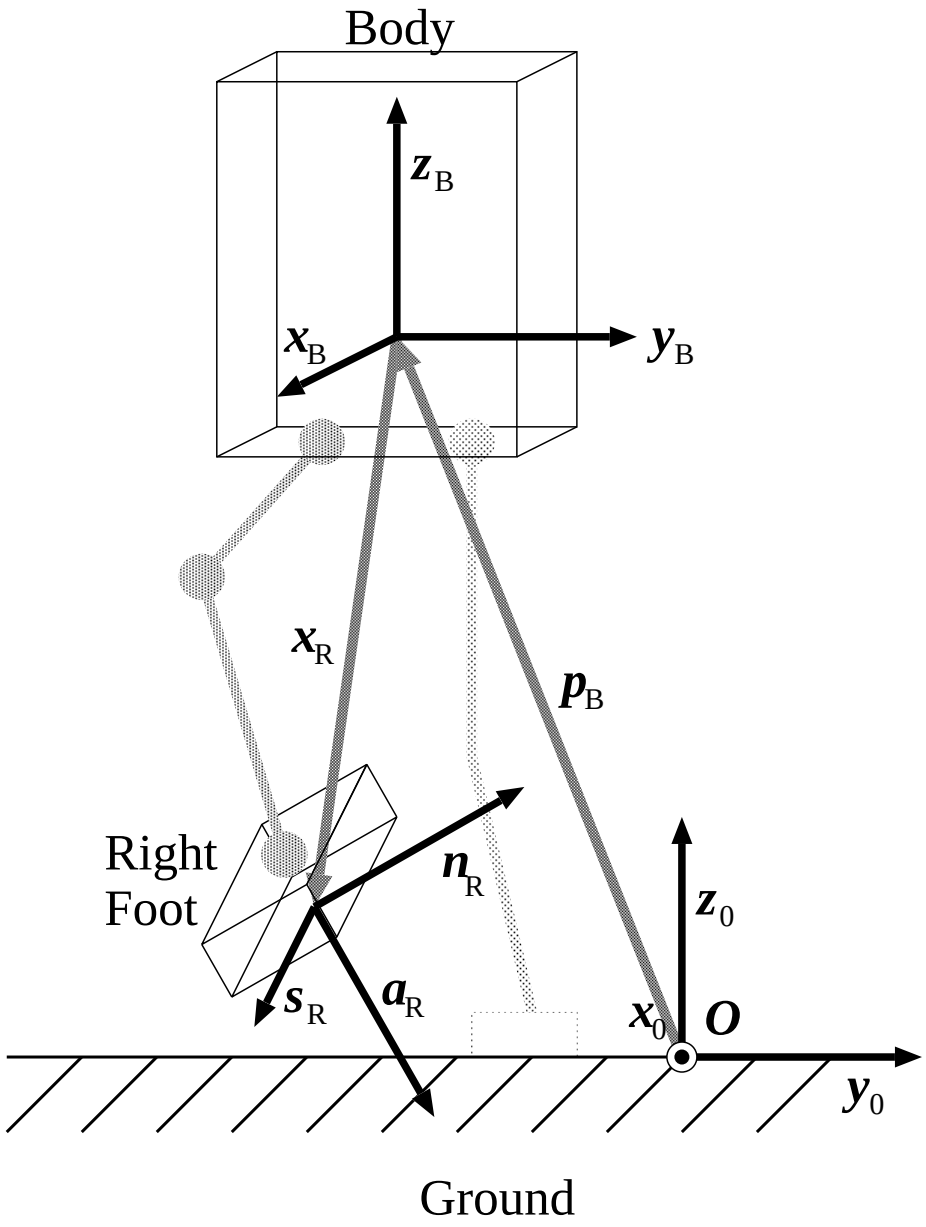


Figure 3.9: Kinematics of legged robot.

and $\boldsymbol{J}_R(\boldsymbol{r})$ denotes the Jacobian matrix

$$\boldsymbol{J}_R(\boldsymbol{r}) = -\frac{\partial \boldsymbol{e}_R}{\partial \boldsymbol{r}^T}. \tag{3.31}$$

In fact, the Jacobian matrix represents the relation between the joint angular velocity and the velocity and angular velocity of the foot such as

$$\begin{bmatrix} \boldsymbol{v}_R \\ \boldsymbol{\omega}_R \end{bmatrix} = \boldsymbol{J}_R \boldsymbol{\omega} \tag{3.32}$$

where $\boldsymbol{v}_R$ and $\boldsymbol{\omega}_R$ denote the velocity and angular velocity of the right foot; $\boldsymbol{\omega}$ is the joint angular velocity.

3.3 Robust Force Control in Environment

The hybrid position/force control is applied to each leg. It becomes the force control mode when the leg is in the support phase, otherwise the position control mode is activated. The workspace position control system consists of the H_∞ robust servo control and the inverse kinematics by the Newton-Raphson method. Assuming the transfer characteristics of the Cartesian position control system is almost unity by the robust controller, the hybrid position/force controller is easily applied to the upper layer of the system. The configuration of the force control system is shown in Fig. 3.10, where $\boldsymbol{f}$ and $\boldsymbol{f}_{ref}$ denote the 6×1 force/torque vector and its reference on the support foot, respectively. The plant system $\boldsymbol{P}_f(s)$ includes dynamics of the environment and the Cartesian position control system, whose control input is $\boldsymbol{x}_{ref}$, the Cartesian position and orientation reference of the support foot.

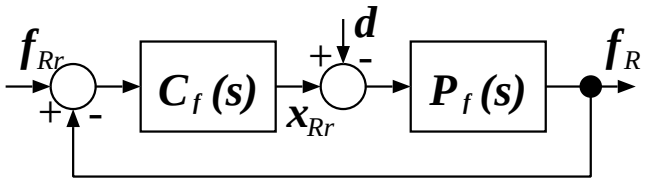


Figure 3.10: Force control system.

When the nominal plant model $\boldsymbol{P}_{fn}(s)$ is given, a very simple parameterization of the robust servo controller can be obtained as follows

$$\boldsymbol{C}_f(s) = \boldsymbol{P}_{fn}(s)^{-1}(\boldsymbol{I} - \boldsymbol{Q}_f(s))^{-1}\boldsymbol{Q}_f(s) \tag{3.33}$$

Here, $\boldsymbol{Q}_f(s)$ is the free parameter representing a complementary sensitivity function and is subject to

$$\boldsymbol{Q}_f(s),\ \boldsymbol{P}_{fn}(s)^{-1}\boldsymbol{Q}_f(s),\ \boldsymbol{P}_{fn}(s)[\boldsymbol{I} - \boldsymbol{Q}_f(s)]\boldsymbol{d}(s),\ [\boldsymbol{I} - \boldsymbol{Q}_f(s)]\boldsymbol{r}(s)\ \ \in \boldsymbol{RH}_\infty \tag{3.34}$$

Here, $\boldsymbol{RH}_\infty$ expresses a set of proper and stable transfer function matrices. These conditions are obtained from the internal stability and the output regulation.

In a case of the force control, the free parameter $\boldsymbol{Q}_f(s)$ and the nominal plant model $\boldsymbol{P}_{fn}(s)$ can be set as

$$\boldsymbol{Q}_f(s) = \mathrm{diag}\{Q_{f1}(s), \ldots, Q_{f6}(s)\} \tag{3.35}$$

$$\boldsymbol{P}_{fn}(s) = \mathrm{diag}\{P_{fn1}(s), \ldots, P_{fn6}(s)\} \tag{3.36}$$

$$\tag{3.37}$$

where

$$Q_{fi}(s) = \frac{3\tau_i^2 s^2 + 3\tau_i s + 1}{(\tau_i s + 1)^3} \tag{3.38}$$

$$P_{fni}(s) = m_i s^2 + b_i s + k_i \tag{3.39}$$

$i = 1, 2, \ldots, 6$. Thus, the robust force controller is obtained from (3.33) as follows.

$$\boldsymbol{C}_f(s) = \mathrm{diag}\{C_{f1}(s), \ldots, C_{f6}(s)\} \tag{3.40}$$

$$C_{fi}(s) = \frac{3\tau_i^2 s^2 + 3\tau_i s + 1}{\tau_i^3 s^3 (m_i s^2 + b_i s + k_i)} \tag{3.41}$$

3.4 Posture Control with Optimal Force Distribution

In this section, the method to control the parallel and rotational motion of the body is presented. In a case of a biped robot, $\boldsymbol{u}_E$ in (2.7), the external force on the generalized coordinates, is expressed as follows.

$$\boldsymbol{u}_E = \begin{bmatrix} \boldsymbol{I}_3 & \mathbf{o} \\ [\boldsymbol{x}_R\times] & \boldsymbol{I}_3 \\ \boldsymbol{J}_{R1}^T & \boldsymbol{J}_{R2}^T \end{bmatrix} \begin{bmatrix} \boldsymbol{f}_R \\ \boldsymbol{n}_R \end{bmatrix} + \begin{bmatrix} \boldsymbol{I}_3 & \mathbf{o} \\ [\boldsymbol{x}_L\times] & \boldsymbol{I}_3 \\ \boldsymbol{J}_{L1}^T & \boldsymbol{J}_{L2}^T \end{bmatrix} \begin{bmatrix} \boldsymbol{f}_L \\ \boldsymbol{n}_L \end{bmatrix} \tag{3.42}$$

where

$\boldsymbol{f}_R$, $\boldsymbol{f}_L$: 3×1 vector of reactive force at the center of right or left foot
$\boldsymbol{n}_R$, $\boldsymbol{n}_L$: 3×1 vector of reactive torque at the center of right or left foot
$\boldsymbol{J}_{Ri}$, $\boldsymbol{J}_{Li}$: $3 \times N$ Jacobian matrix of right or left foot
$\boldsymbol{x}_R$, $\boldsymbol{x}_L$: 3×1 position vector of the right or left foot with respect to the origin of $\boldsymbol{p}_B$

Here, $[\boldsymbol{a}\times]$ denotes a matrix representing a cross product, and $\boldsymbol{I}_n$ denotes an $n \times n$ identity matrix. Fig. 3.11 illustrates the kinematic property.

While there is no actuated control input for the position and the attitude of the body, i. e., $\boldsymbol{f}_B = \boldsymbol{n}_B = 0$, those of the body are still controllable by using the reactive force and torque ($\boldsymbol{f}_R$, $\boldsymbol{f}_L$, $\boldsymbol{n}_R$, $\boldsymbol{n}_L$) as indirect control inputs. To use the reactive force and torque as indirect control inputs, the hybrid position/force control is applied to each leg. If the leg is in the support phase, the force control is activated. Otherwise the position control becomes active. The workspace position control system consists of the inertia fluctuation insensitive servo controland the inverse kinematics by the Newton method. The force controller is applied as the upper layer of the position controller(See Fig. 3.1)

Then the posture control system is applied as a supervisory control to the reactive force controller. The objective is to make the *center of mass* (COM) of the robot and the attitude of the body converge to its given reference trajectories. The parallel motion of the COM of the robot and the rotational motion of the body can be modeled from the first 6 rows of (2.7) and (3.42) as follows.

$$\tilde{\boldsymbol{H}}\dot{\tilde{\boldsymbol{v}}} + \tilde{\boldsymbol{b}} = \tilde{\boldsymbol{u}}_E \tag{3.43}$$

$$\tilde{\boldsymbol{u}}_E = \tilde{\boldsymbol{K}}\boldsymbol{f}_A \tag{3.44}$$

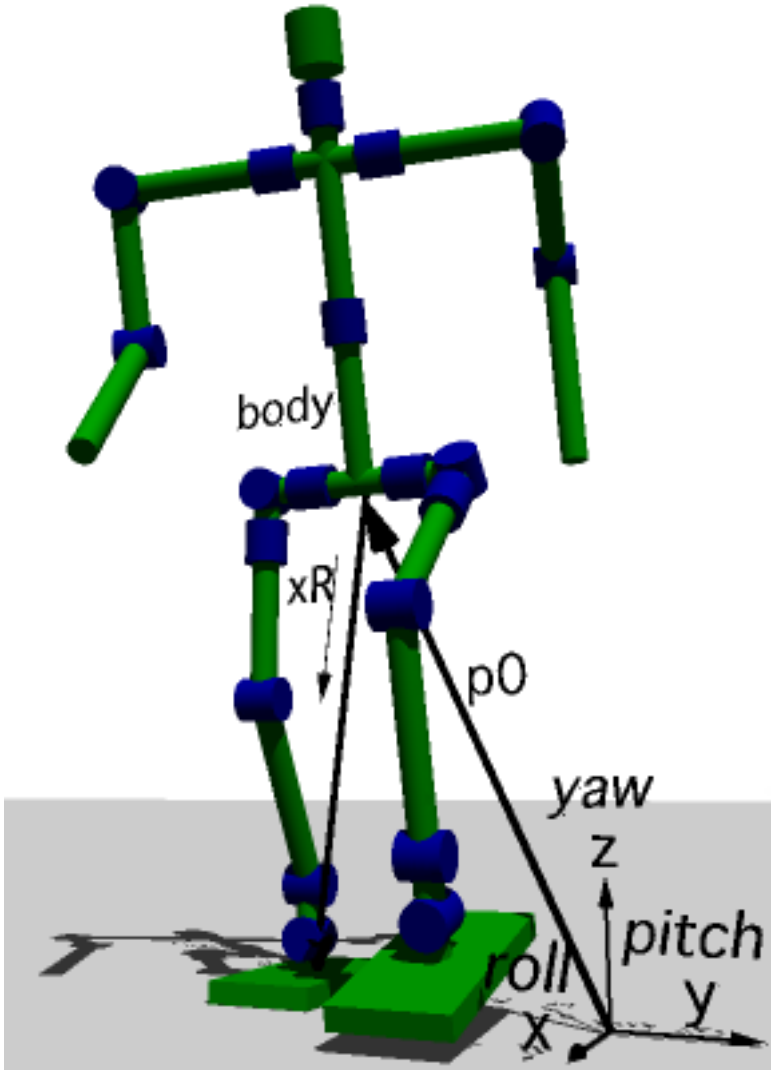


Figure 3.11: An illustration of the kinematic property.

and

$$\tilde{\boldsymbol{x}}^T = [\ \boldsymbol{p}_C^T \quad \boldsymbol{A}_B^T\ ] \tag{3.45}$$

$$\tilde{\boldsymbol{v}}^T = [\ \boldsymbol{v}_C^T \quad \boldsymbol{\omega}_B^T\ ] \tag{3.46}$$

$$\boldsymbol{f}_A^T = [\ \boldsymbol{f}_R^T \quad \boldsymbol{n}_R^T \quad \boldsymbol{f}_L^T \quad \boldsymbol{n}_L^T\ ] \tag{3.47}$$

$$\tilde{\boldsymbol{K}} = \begin{bmatrix} \boldsymbol{I}_3 & \mathbf{o} & \boldsymbol{I}_3 & \mathbf{o} \\ [\boldsymbol{x}_R\times] & \boldsymbol{I}_3 & [\boldsymbol{x}_L\times] & \boldsymbol{I}_3 \end{bmatrix} \tag{3.48}$$

$$\dot{\boldsymbol{p}}_C = \boldsymbol{v}_C \tag{3.49}$$

where

$\tilde{\boldsymbol{H}}$: 6×6 inertia matrix
$\tilde{\boldsymbol{b}}$: 6×1 vector specifying gravity effect and non-linear terms
$\tilde{\boldsymbol{u}}_E$: 6×1 vector specifying external force at COM and external torque around body
$\boldsymbol{p}_C$: 3×1 vector specifying COM of the robot

The ideal external force input $\tilde{\boldsymbol{u}}_E^*$ at the COM of the robot and around the body is determined by the state feedback.

$$\tilde{\boldsymbol{u}}_E^* = \tilde{\boldsymbol{H}}_n[\boldsymbol{K}_p(\tilde{\boldsymbol{x}}_{ref} - \tilde{\boldsymbol{x}}) + \boldsymbol{K}_d(\dot{\tilde{\boldsymbol{x}}}_{ref} - \dot{\tilde{\boldsymbol{x}}}) + \ddot{\tilde{\boldsymbol{x}}}_{ref}] + \tilde{\boldsymbol{b}}_n \tag{3.50}$$

where

$\tilde{\boldsymbol{H}}_n$: 6×6 diagonal matrix specifying nominal inertia
$\tilde{\boldsymbol{b}}_n$: 6×1 constant vector specifying non-linear terms, gravity effect, and non-diagonal inertial force

and $\boldsymbol{K}_p$, $\boldsymbol{K}_d$ represent gain matrices. The posture reference $\tilde{\boldsymbol{x}}_{ref}$ is generated by linearized inverted pendulum models

The contact force and torque are, however, physically limited to the repulsive condition, the friction condition, and *zero moment point* (ZMP) condition. Thus the ideal force $\tilde{\boldsymbol{u}}_E^*$ is not always realized by the reactive force and torque $\boldsymbol{f}_A$. We need to consider the following physical conditions.

Let the components of $\boldsymbol{f}_R$, $\boldsymbol{n}_R$, $\boldsymbol{f}_L$, and $\boldsymbol{n}_L$ be

$$\boldsymbol{f}_R^T = [f_{Rx}\ f_{Ry}\ f_{Rz}] \tag{3.51}$$
$$\boldsymbol{n}_R^T = [n_{Rx}\ n_{Ry}\ n_{Rz}] \tag{3.52}$$
$$\boldsymbol{f}_L^T = [f_{Lx}\ f_{Ly}\ f_{Lz}] \tag{3.53}$$
$$\boldsymbol{n}_L^T = [n_{Lx}\ n_{Ly}\ n_{Lz}]. \tag{3.54}$$

The normal component of the reactive force on the ground plane is not attractive but repulsive, which yields the following non-negative conditions.

$$f_{Rz} \geq 0 \tag{3.55}$$
$$f_{Lz} \geq 0 \tag{3.56}$$

The friction force, i. e., the tangent component of the reactive force on the ground plane always exists within the friction cone.

$$\sqrt{f_{Rx}^2 + f_{Ry}^2} \leq \mu f_{Rz} \tag{3.57}$$
$$\sqrt{f_{Lx}^2 + f_{Ly}^2} \leq \mu f_{Lz} \tag{3.58}$$
$$|n_{Rz}| \leq \mu' f_{Rz} \tag{3.59}$$
$$|n_{Lz}| \leq \mu' f_{Lz} \tag{3.60}$$

where μ and μ' denote friction coefficients. It is possible to break out slips at the contact points when the equality in the (3.58)–(3.60) is realized.

The tangent component of the reactive torque at the center of the foot on the ground plane is also limited due to finiteness of the contact area.

$$|n_{Rx}| \leq d_y f_{Rz} \tag{3.61}$$
$$|n_{Lx}| \leq d_y f_{Lz} \tag{3.62}$$
$$|n_{Ry}| \leq d_x f_{Rz} \tag{3.63}$$
$$|n_{Ly}| \leq d_x f_{Lz} \tag{3.64}$$

where d_x and d_y denote a half of the length and the width of the foot, respectively. (3.62)–(3.64) are equivalent to the *zero moment point* conditions.

Due to the physical limitations (3.55)–(3.64), the ideal force input $\tilde{\boldsymbol{u}}_E^*$ by (3.50) cannot be always realized by the reactive force and torque $\boldsymbol{f}_A$. Thus the following performance indices are introduced, which should be minimized under the limitations.

$$J_{main} = \frac{1}{2}(\boldsymbol{u}_E - \boldsymbol{u}_E^*)^T \boldsymbol{C}_1 (\boldsymbol{u}_E - \boldsymbol{u}_E^*) \tag{3.65}$$
$$J_{sub} = \frac{1}{2}\boldsymbol{u}_E^T \boldsymbol{C}_2 \boldsymbol{u}_E \tag{3.66}$$

The index J_{main} corresponds to the square error between the ideal force and torque and the realizable ones. The index J_{sub} corresponds to the square error between the force and torque of the left foot and those of the right one. The reactive force and torque input $\boldsymbol{f}_A$ is determined by the quadratic programming, which minimizes the performance index under the linearized constraints of (3.55)–(3.64).

$$\min_{f_A} \quad J_{main} + \epsilon J_{sub} \tag{3.67}$$
$$\text{subject to} \quad \boldsymbol{A}\boldsymbol{f}_A \leq \boldsymbol{b} \tag{3.68}$$

where

$$\boldsymbol{C}_1 = \mathrm{diag}\{w_1, w_2, w_3, w_4, w_5, w_6\} \tag{3.69}$$
$$\boldsymbol{C}_2 = \begin{bmatrix} \boldsymbol{I}_6 & -\boldsymbol{I}_6 \\ -\boldsymbol{I}_6 & \boldsymbol{I}_6 \end{bmatrix} \tag{3.70}$$
$$\boldsymbol{A} = \begin{bmatrix} \boldsymbol{A}' & \mathbf{o} \\ \mathbf{o} & \boldsymbol{A}' \end{bmatrix} \tag{3.71}$$
$$\boldsymbol{A}' = \begin{bmatrix} 1 & 0 & -\mu & 0 & 0 & 0 \\ -1 & 0 & -\mu & 0 & 0 & 0 \\ 0 & 1 & -\mu & 0 & 0 & 0 \\ 0 & -1 & -\mu & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & -1 & 0 & 0 & 0 \\ 0 & 0 & -d_y & 1 & 0 & 0 \\ 0 & 0 & -d_y & -1 & 0 & 0 \\ 0 & 0 & -d_x & 0 & 1 & 0 \\ 0 & 0 & -d_x & 0 & -1 & 0 \\ 0 & 0 & -\mu' & 0 & 0 & 1 \\ 0 & 0 & -\mu' & 0 & 0 & -1 \end{bmatrix} \tag{3.72}$$

$$\boldsymbol{b}^T = [\ \boldsymbol{b}_R^T \quad \boldsymbol{b}_L^T\ ] \tag{3.73}$$

$$\boldsymbol{b}_R^T = [\,0\ 0\ 0\ 0\ z_R\ 0\ 0\ 0\ 0\ 0\ 0\ 0\,] \tag{3.74}$$

$$\boldsymbol{b}_L^T = [\,0\ 0\ 0\ 0\ z_L\ 0\ 0\ 0\ 0\ 0\ 0\ 0\,] \tag{3.75}$$

$$z_R = \begin{cases} any & \text{if right leg is in support phase} \\ 0 & \text{otherwise} \end{cases} \tag{3.76}$$

$$z_L = \begin{cases} any & \text{if left leg is in support phase} \\ 0 & \text{otherwise} \end{cases} \tag{3.77}$$

and ϵ is a small positive real number.

The main performance index J_{main} approaches the solution to the ideal force and torque $\boldsymbol{u}_c^*$ given by the state feedback. The sub performance index J_{sub} distributes the inner force and torque to the both foot in balance. Because ϵ is very small, the sub performance index has almost no influence on the main performance index. The constrained (3.68) acts as a kind of a limiter in multi-input systems.

The optimization problem (3.67) and (3.68) are equivalent to the following quadratic programming problem.

$$\min_{f_A} \quad \frac{1}{2}\boldsymbol{f}_A^T\boldsymbol{Q}\boldsymbol{f}_A - \boldsymbol{c}_0^T\boldsymbol{f}_A \tag{3.78}$$

$$\text{subject to} \quad \boldsymbol{A}\boldsymbol{f}_A \leq \boldsymbol{b} \tag{3.79}$$

where $\boldsymbol{Q} = \tilde{\boldsymbol{K}}^T\boldsymbol{C}_1\tilde{\boldsymbol{K}} + \epsilon\boldsymbol{C}_2$ and $\boldsymbol{c}_0 = \tilde{\boldsymbol{K}}^T\boldsymbol{C}_1\boldsymbol{u}_c^*$. The reactive force and torque reference can be obtained by solving it for each sampling period.

The controller does not need the exact information about the upper bound of the friction coefficient of the ground but the smaller value than it. Thus it is sufficient for stability if the value of the friction coefficient in the controller is set smaller than the actual friction coefficient.

3.5 Yaw Moment Compensation by Arm Motion

Since the proposed system has 6 degree-of-freedom force controller for each leg, it can also control yaw rotation. However, the yaw moment gets large when the walking speed gets fast. It becomes beyond the moment which the foot can generate and the walking gets unstable. The same situation will occur in a case of a walking on low μ terrain. The off-line type calculation method is proposed in order to compensate the moment[41].

In the discussion of the previous section, the equality $\boldsymbol{u}_c = \boldsymbol{u}_c^*$ is realized only if the solution without constraint exists within the convex polytope and it is not activated. In the case of a human walk, the yaw moment generated on the foot plane is too small to compensate the disturbance moment by free-leg motion, etc., so the moment of arm motions is used to do it.

The model of the arms is shown in Fig. 3.12. The synchronized motion of the right and left arms generates only yaw moment:

$$u_c^{yaw} = 2abm(\ddot{q}_a\cos\theta_a - \dot{\theta}_a\sin\theta_a) \tag{3.80}$$

where θ_a denotes angular reference of the shoulder. a and b express the properties of the position of the COM of the arm (See Fig. 3.12). m denotes the mass of the arm. Linearizing the right

side of the equation around $\theta_a = 0$ and adding high-pass filter to remove dc offset, we have angular reference of the shoulder such as:

$$\theta_a^{ref} = \frac{s^3}{s^3 + 2\omega_c s^2 + 2\omega_c^2 s + \omega_c^3} \cdot \frac{K_a}{s^2}(u_c^{*yaw} - u_c^{yaw}) \tag{3.81}$$

where $K_a = 1/(2abm)$, and ω_c represents cut-off frequency of the high-pass filter.

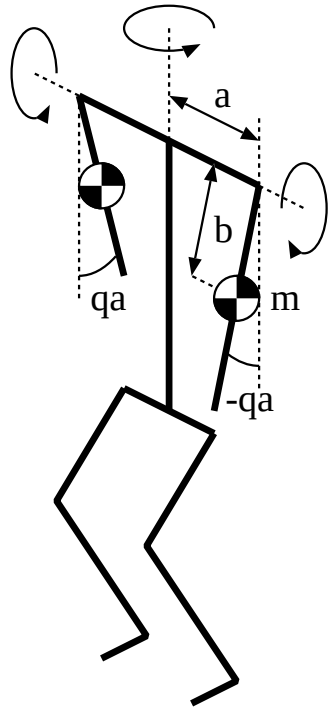


Figure 3.12: Model of arms.

3.6 Globally Stable Walking Pattern Generation

Owing to the posture controller with the environmental interaction mentioned above, the on-line type autonomous walking control system can be realized as follows.

The posture controller mentioned above stabilizes the robot in the single and double support phases but does not realize the global locomotion. In order to achieve a stable walking, the foot must be periodically landed at adequate point. In this section, the tracking control of the COM to an arbitral trajectory by planning the land point of the foot is described.

Dynamics of the COM of the robot behaves like an inverted pendulum by the posture controller, i. e., $\ddot{p}_{cx} = \omega^2(p_{cx} - p_{gx})$ in sagittal plane and $\ddot{p}_{cy} = \omega^2(p_{cy} - p_{gy})$ in lateral plane, where p_{cx} an p_{cy} denote the COM position in sagittal plane and in lateral plane, respectively. p_{gx} and p_{gy} denote the foot position of the support leg with respect to the origin of the world fixed coordinates in sagittal plane and in lateral plane, respectively. Digitalizing the inverted

pendulum with one step period T, we have the discrete-time inverted pendulum (3.82). (Here, only the equations in the sagittal plane are described. The forms of the equations in the lateral plane are completely the same.)

$$\begin{bmatrix} p_{cx}(t+T) \\ \dot{p}_{cx}(t+T) \end{bmatrix} = \begin{bmatrix} c & \frac{s}{\omega} \\ \omega s & c \end{bmatrix} \begin{bmatrix} p_{cx}(t) \\ \dot{p}_{cx}(t) \end{bmatrix} + \begin{bmatrix} 1-c \\ -\omega s \end{bmatrix} p_{gx}(t) \tag{3.82}$$

where $c = \cosh \omega T$ and $s = \sinh \omega T$.

The movement of the free-leg must just begin at one step period in advance. Thus the augmented system including one step period delay is considered

$$\begin{bmatrix} p_{cx}(t+T) \\ \dot{p}_{cx}(t+T) \\ p_{gx}(t+T) \end{bmatrix} = \begin{bmatrix} c & s/\omega & 1-c \\ \omega s & c & -\omega s \\ 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} p_{cx}(t) \\ \dot{p}_{cx}(t) \\ p_{gx}(t) \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} p_{gx}^{ref}(t) \tag{3.83}$$

The reference of the landing point $p_{gx}^{ref}(t)$ which sets the system characteristic polynomial $\phi(z)$ to $\phi(z) = z^3 + \alpha_2 z^2 + \alpha_1 z + \alpha_0$ is obtained by the state feedback as follows.

$$\begin{aligned} &p_{gx}^{ref}(t+T) = \\ &\quad f_0[p_{cx}(t) - p_{cx}^{cmd}(t)] + f_1[\dot{p}_{cx}(t) - \dot{p}_{cx}^{cmd}(t)] + f_2[p_{gx}(t) - p_{gx}^{cmd}(t)] + p_{gx}^{cmd}(t+T) \end{aligned} \tag{3.84}$$

where

$$f_0 = \frac{1 - \alpha_0 - \alpha_1 + \alpha_2 + 2(1-\alpha_2)c - 4c^2}{2(c-1)} \tag{3.85}$$

$$f_1 = \frac{1 + \alpha_0 - \alpha_1 - \alpha_2 - 2(1+\alpha_2)c - 4c^2}{2\omega s} \tag{3.86}$$

$$f_2 = \alpha_2 + 2c \tag{3.87}$$

p_{cx}^{cmd} denotes the reference trajectory of COM in sagittal plane and p_{gx}^{cmd} denotes the offset of the landing points synchronizing trajectory of COM in sagittal plane.

The trajectory of the free-leg is given by connecting the next and previous landing point with a smooth function as follows.

$$p_{gx}^{ref}(t+m) = p_{gx}^{ref}(t-T) + \frac{1-\cos \pi m}{2}[p_{gx}^{ref}(t+T) - p_{gx}^{ref}(t-T)] \tag{3.88}$$

and the height of free-leg is also given by

$$p_{gz}^{ref}(t+m) = h^{ref} \frac{1-\cos 2\pi m}{2} \tag{3.89}$$

where $0 \leq m \leq T$.

The global system configuration is shown in Fig. 3.1.

Chapter 4

Dynamic Simulator of Legged Robot

This chapter describes new two type of simulation methods of legged robots. One is a forward dynamic simulator which numerically calculates trajectories of the joint angles, the body position, and the body attitude of the legged robot given the actuator input torques. The other is an inverse dynamic simulator which numerically calculates trajectories of the joint torques and the interaction forces from the ground given the position, the velocity, the acceleration of the joint angles and those of the body of the robot. The forward dynamic simulator is available for investigation of various control algorithms. On the other hand, the inverse dynamic simulator is useful in analyzing the joint torque and the reactive forces from the ground when the time series data of the joint angles are given, which might be measured from human walking or generated by various trajectory generators.

4.1 Forward Dynamic Simulator

This section describes a new 3-dimensional forward dynamic simulation method of multi-degree-of-freedom mechanical systems with the time-dependent contact and the Coulomb friction, such as a legged robot interacting with a ground, a satellite-mounted manipulator catching an object, etc. The proposed method is the extension of the open link manipulator simulation[38] and the contact simulation of rigid body mechanics[19]. The proposed simulation model is mathematically exact, thus this simulator enables essential investigation of various control algorithms of mechanical systems.

4.1.1 Numerical Integration

It is very complicated to obtain an algebraic equation of a legged robot. But it is easy to simulate a dynamic motion of a legged robot (2.4)–(2.7) numerically by integrating $\dot{\boldsymbol{v}}$ and $\boldsymbol{v}$ on each time step after solving (2.7) for the acceleration $\dot{\boldsymbol{v}}$ given $\boldsymbol{x}$, $\boldsymbol{v}$, $\boldsymbol{u}$, and $\boldsymbol{u}_E$. In a case of the Euler integration, we have

$$\boldsymbol{p}_B(t+h) = \boldsymbol{p}_B(t) + h\boldsymbol{v}_B(t) \tag{4.1}$$

$$\boldsymbol{A}_B(t+h) = \boldsymbol{T}(h\boldsymbol{\omega}_B(t))\boldsymbol{A}_B(t) \tag{4.2}$$

$$\boldsymbol{\theta}(t+h) = \boldsymbol{\theta}(t) + h\boldsymbol{\omega}(t) \tag{4.3}$$

$$\boldsymbol{v}(t+h) = \boldsymbol{v}(t) + h\dot{\boldsymbol{v}}(t) \tag{4.4}$$

$$\dot{\boldsymbol{v}}(t) \;=\; \boldsymbol{H}(\boldsymbol{x}(t))^{-1}[\boldsymbol{u}(t) + \boldsymbol{u}_E(\boldsymbol{x}(t), \boldsymbol{v}(t)) - \boldsymbol{b}(\boldsymbol{x}(t), \boldsymbol{v}(t))] \tag{4.5}$$

where $\boldsymbol{b}(\boldsymbol{x}(t), \boldsymbol{v}(t))$ represents biasing vector

$$\boldsymbol{b}(\boldsymbol{x}(t), \boldsymbol{v}(t)) = \boldsymbol{C}(\boldsymbol{x}(t), \boldsymbol{v}(t))\boldsymbol{v}(t) + \boldsymbol{g}(\boldsymbol{x}(t)) \tag{4.6}$$

and h is the time-step. $\boldsymbol{T}(h\boldsymbol{\omega}_B)$ acts as a rotational transformer around $\boldsymbol{\omega}_B$ axis with angle $h|\boldsymbol{\omega}_B|$. (See Fig. 4.1 and [26].) $\boldsymbol{T}(h\boldsymbol{\omega}_B)$ is obtained by

$$\boldsymbol{T}(h\boldsymbol{\omega}_B) = \left[(\cos\psi)\boldsymbol{I}_3 + (1-\cos\psi)\boldsymbol{r}\boldsymbol{r}^T + (\sin\psi)[\boldsymbol{r}\times]\right] \tag{4.7}$$

where $\psi = h|\boldsymbol{\omega}_B|$, $\boldsymbol{r} = \boldsymbol{\omega}_B/|\boldsymbol{\omega}_B|$.

$\boldsymbol{H}(\boldsymbol{x})$ and $\boldsymbol{b}(\boldsymbol{x}, \boldsymbol{v})$ can be obtained by inverse dynamics calculation using the Newton-Euler formulation, i. e., solution for $\boldsymbol{u}$ given $\boldsymbol{x}$, $\boldsymbol{v}$, and $\dot{\boldsymbol{v}}$. In fact, $\boldsymbol{H}(\boldsymbol{x})$ can be calculated by solving inverse dynamics with setting $\boldsymbol{x}$ to the current state and $\dot{\boldsymbol{v}} = \boldsymbol{e}_j$, and ignoring centrifugal forces, Corioli's forces, gravity effects, and external forces[38]. Here, $\boldsymbol{e}_j$ means a unit vector with its jth element equal to 1 and others are 0. The solution about $\boldsymbol{u}$ corresponds to the jth column of $\boldsymbol{H}$. (See (2.7).) The biasing vector $\boldsymbol{b}(\boldsymbol{x}, \boldsymbol{v})$ can be also computed by setting $(\boldsymbol{x}, \boldsymbol{v})$ to the current state and $\dot{\boldsymbol{v}} = 0$, and ignoring external forces.

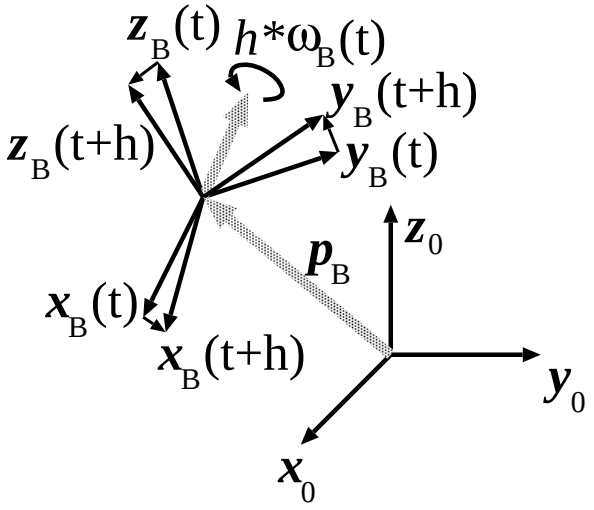


Figure 4.1: Rotation of coordinates.

4.1.2 Exact Model of Legged Robot

To obtain the motion of the legged robot, we need to calculate inverse dynamics which can be calculated by following recursive equations. The formulation is based on the algorithm in [20], but several points are different. The advantages of the new method proposed in this section are: 1) there is no fixed point in the robotic system and the base-link is movable, and 2) the expression of the link-fixed coordinates is modified to deal with branching links.

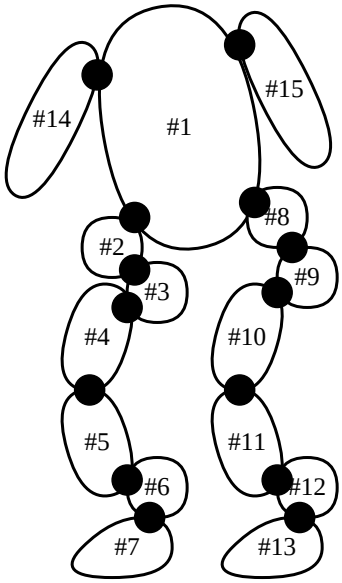


Figure 4.2: Indices of links.

Table 4.1: Examples of a inner link and outer links in Fig. 4.2 case.

link i	1	2	7	8	14
inner link $\underline{i}$	–	1	6	1	1
outer links O_i	2, 8, 14, 15	3	$\emptyset$	9	$\emptyset$

The index numbers of links are introduced as shown in Fig. 4.2 in a 14-axis biped case. Here, the link $\underline{i}$ is defined as the *inner link* of ith link, whose index is the smallest in all links connected to ith link. Also the set of links O_i is defined as the *outer links* of ith link, whose indices are larger than i in all links connected to ith link. The examples are shown in Table 4.1. To compute inverse dynamics efficiently, the link-fixed coordinates shown in Fig. 4.3 are introduced. The ith coordinates are defined as

$$
\begin{aligned}
{}^0\boldsymbol{A}_i &= {}^0\boldsymbol{A}_1 {}^1\boldsymbol{A}_2 \ldots {}^{\underline{i}}_{i}\boldsymbol{A}_i && (4.8)\\
{}^0\boldsymbol{A}_1 &= \boldsymbol{A}_B && (4.9)
\end{aligned}
$$

where

$$
{}^{\underline{i}}_{i}\boldsymbol{A}_i = [{}^{\underline{i}}\boldsymbol{x}_i, {}^{\underline{i}}\boldsymbol{y}_i, {}^{\underline{i}}\boldsymbol{z}_i]. \qquad (4.10)
$$

${}^{\underline{i}}_{i}\boldsymbol{A}_i$ correspond to ith coordinates referenced to $\underline{i}$th coordinates system which transforms any vector with reference to ith coordinates system to $\underline{i}$th coordinates system. (Note: ${}^{\underline{i}}_{i}\boldsymbol{A}_i^{-1} = {}^{\underline{i}}_{i}\boldsymbol{A}_i^T = {}^{i}\boldsymbol{A}_{\underline{i}}$.) In a case of the parameterization of Fig. 4.3,

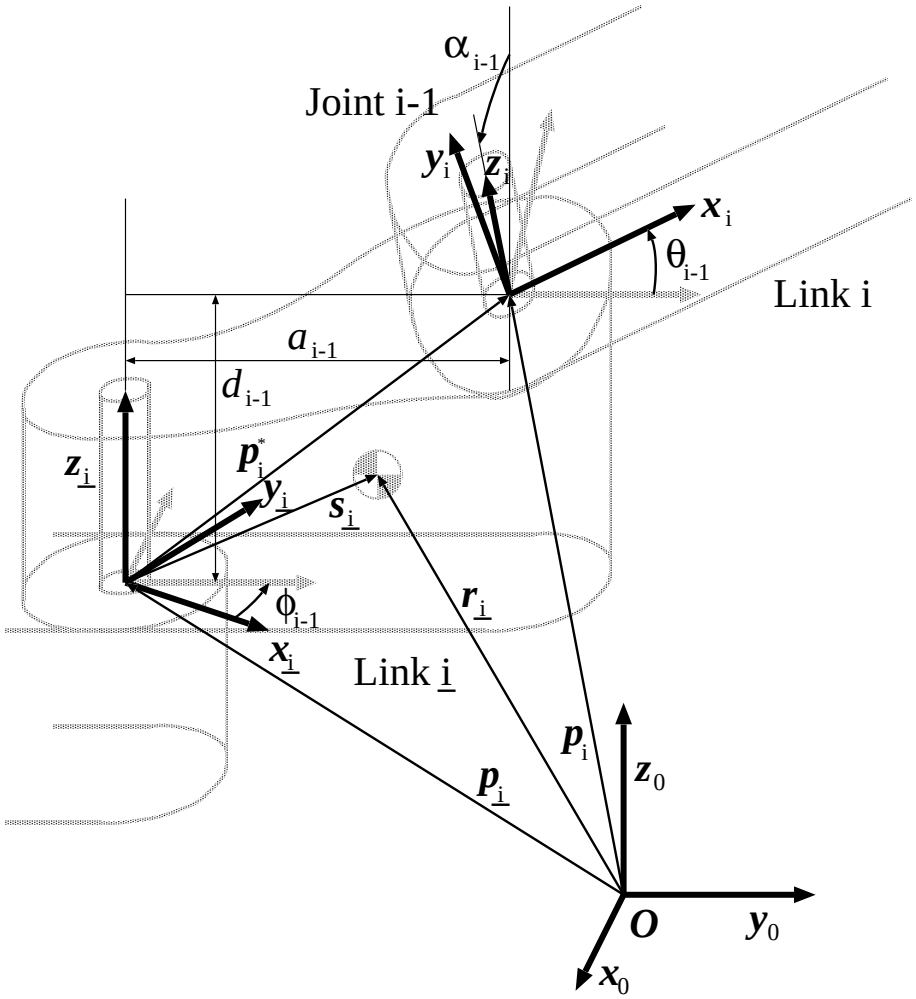


Figure 4.3: Coordinates and parameters of links.

$$ {}^{i}\boldsymbol{A}_i = \begin{bmatrix} c_\phi c_\theta - s_\phi c_\alpha s_\theta & -c_\phi s_\theta - s_\phi c_\alpha c_\theta & s_\phi s_\alpha \\ s_\phi c_\theta + c_\phi c_\alpha s_\theta & -s_\phi s_\theta + c_\phi c_\alpha c_\theta & -c_\phi s_\alpha \\ s_\alpha s_\theta & s_\alpha c_\theta & c_\alpha \end{bmatrix} \tag{4.11} $$

where $s_\phi = \sin\phi_{i-1}$, $c_\phi = \cos\phi_{i-1}$, $s_\alpha = \sin\alpha_{i-1}$, $c_\alpha = \cos\alpha_{i-1}$, $s_\theta = \sin\theta_{i-1}$, and $c_\theta = \cos\theta_{i-1}$.

When $\dot{\boldsymbol{v}}_B$, $\dot{\boldsymbol{\omega}}_B$, $\boldsymbol{\omega}_B$, $\ddot{q}_i$, $\dot{q}_i$ and q_i, $1 \le i \le N$, are given, the angular velocity ${}^{i}\boldsymbol{\omega}_i$, the angular acceleration ${}^{i}\dot{\boldsymbol{\omega}}_i$, and the acceleration of the origin ${}^{i}\ddot{\boldsymbol{p}}_i$ of ith link referenced to its own link coordinates can be recurrently obtained as follows for $i = 1$

$$ {}^{1}\ddot{\boldsymbol{p}}_1 = {}^{1}\boldsymbol{A}_0(\dot{\boldsymbol{v}}_B + \boldsymbol{g}) \tag{4.12} $$

$$ {}^{1}\boldsymbol{\omega}_1 = {}^{1}\boldsymbol{A}_0\boldsymbol{\omega}_B \tag{4.13} $$

$$ {}^{1}\dot{\boldsymbol{\omega}}_1 = {}^{1}\boldsymbol{A}_0\dot{\boldsymbol{\omega}}_B, \tag{4.14} $$

and for $2 \le i \le N$

$$ {}^{i}\boldsymbol{\omega}_i = {}^{i}\boldsymbol{A}_{\underline{i}}\,{}^{\underline{i}}\boldsymbol{\omega}_{\underline{i}} + \boldsymbol{z}_0\dot{q}_{i-1} \tag{4.15} $$

$$
\begin{aligned}
{}^i\dot{\boldsymbol{\omega}}_i &= {}^i\boldsymbol{A}_{\underline{i}}\,{}^{\underline{i}}\dot{\boldsymbol{\omega}}_{\underline{i}} + \boldsymbol{z}_0\ddot{q}_{i-1} + ({}^i\boldsymbol{A}_{\underline{i}}\boldsymbol{\omega}_{\underline{i}}) \times \boldsymbol{z}_0\dot{q}_{i-1} && (4.16)\\
{}^i\ddot{\boldsymbol{p}}_i &= {}^i\boldsymbol{A}_{\underline{i}}\left[{}^{\underline{i}}\dot{\boldsymbol{\omega}}_{\underline{i}} \times {}^{\underline{i}}\boldsymbol{p}_i^* + {}^{\underline{i}}\boldsymbol{\omega}_{\underline{i}} \times ({}^{\underline{i}}\boldsymbol{\omega}_{\underline{i}} \times {}^{\underline{i}}\boldsymbol{p}_i^*) + {}^{\underline{i}}\ddot{\boldsymbol{p}}_{\underline{i}}\right]. && (4.17)
\end{aligned}
$$

Thus the acceleration of the center of mass $\ddot{\boldsymbol{r}}_i$, the total force $\boldsymbol{F}_i$, and the total moment $\boldsymbol{N}_i$ of the ith link can be calculated as follows for $1 \le i \le N$,

$$
\begin{aligned}
{}^i\ddot{\boldsymbol{r}}_i &= {}^i\dot{\boldsymbol{\omega}}_i \times {}^i\boldsymbol{s}_i + {}^i\boldsymbol{\omega}_i \times ({}^i\boldsymbol{\omega}_i \times {}^i\boldsymbol{s}_i) + {}^i\ddot{\boldsymbol{p}}_i && (4.18)\\
{}^i\boldsymbol{F}_i &= {}^i m_i\,{}^i\ddot{\boldsymbol{r}}_i && (4.19)\\
{}^i\boldsymbol{N}_i &= {}^i\boldsymbol{J}_i\,{}^i\dot{\boldsymbol{\omega}}_i + {}^i\boldsymbol{\omega}_i \times ({}^i\boldsymbol{J}_i\,{}^i\boldsymbol{\omega}_i). && (4.20)
\end{aligned}
$$

Here, $\boldsymbol{p}_i^*$ denotes $\boldsymbol{p}_i - \boldsymbol{p}_{\underline{i}}$ which is calculated as

$$
{}^{\underline{i}}\boldsymbol{p}_i^* = [a_{i-1}\cos\phi_{i-1}, a_{i-1}\sin\phi_{i-1}, d_{i-1}]^T. \qquad (4.21)
$$

$\boldsymbol{z}_i$ denotes the direction of the joint $i-1$, and $\boldsymbol{s}_i$ denotes the center of mass with respect to the origin of link i coordinates. (See Fig. 4.3.) The gravity effect can be considered by adding a gravity acceleration on the one of the base link $\dot{\boldsymbol{v}}_B$.

Then, $\boldsymbol{f}_i$ and $\boldsymbol{n}_i$, the force and moment exerted on link i by inner link $\underline{i}$ can be calculated as follows for $1 \le i \le N$

$$
\begin{aligned}
{}^i\boldsymbol{f}_i &= {}^i\boldsymbol{F}_i + \sum_{j\in O_i} {}^{\underline{j}}\boldsymbol{A}_j\,{}^j\boldsymbol{f}_j + {}^i\boldsymbol{A}_0 \sum_{j\in M_i} {}^0\boldsymbol{f}_{Ej} && (4.22)\\
{}^i\boldsymbol{n}_i &= {}^i\boldsymbol{N}_i + \sum_{j\in O_i}\left[{}^{\underline{j}}\boldsymbol{A}_j\,{}^j\boldsymbol{n}_j + {}^{\underline{j}}\boldsymbol{p}_j^* \times ({}^{\underline{j}}\boldsymbol{A}_j\,{}^j\boldsymbol{f}_j)\right] + {}^i\boldsymbol{s}_i \times {}^i\boldsymbol{F}_i - \sum_{j\in M_i}\left[{}^j\boldsymbol{c}_j \times ({}^i\boldsymbol{A}_0\,{}^0\boldsymbol{f}_{Ej})\right] && (4.23)
\end{aligned}
$$

where

$\boldsymbol{f}_{Ej}$	:	3×1 vector specifying jth external force
M_i	:	a set of index numbers of external forces which are imposed on link i
$\boldsymbol{c}_j$	:	3×1 position vector of jth contact point with respect to the origin of its own link-fixed coordinates as shown in Fig. 4.4

As a result, the solution of inverse dynamics $\boldsymbol{u}^T$ $(= [\boldsymbol{f}_B^T, \boldsymbol{n}_B^T, \boldsymbol{\tau}^T])$ given $\boldsymbol{x}$, $\boldsymbol{v}$, and $\dot{\boldsymbol{v}}$ is obtained as follows for $1 \le i \le N-1$

$$
\tau_i = {}^{i+1}\boldsymbol{n}_{i+1}^T\boldsymbol{z}_0 - D_i\dot{q}_i, \qquad (4.24)
$$

and for $i = 1$

$$
\begin{aligned}
\boldsymbol{f}_B &= {}^0\boldsymbol{A}_1\,{}^1\boldsymbol{f}_1 && (4.25)\\
\boldsymbol{n}_B &= {}^0\boldsymbol{A}_1\,{}^1\boldsymbol{n}_1 && (4.26)
\end{aligned}
$$

where $\boldsymbol{f}_B$ and $\boldsymbol{n}_B$ are the force and the moment exerted on the origin of the base link, respectively. τ_i denotes the torque on ith joint. D_i denotes the viscous damping coefficient of joint i.

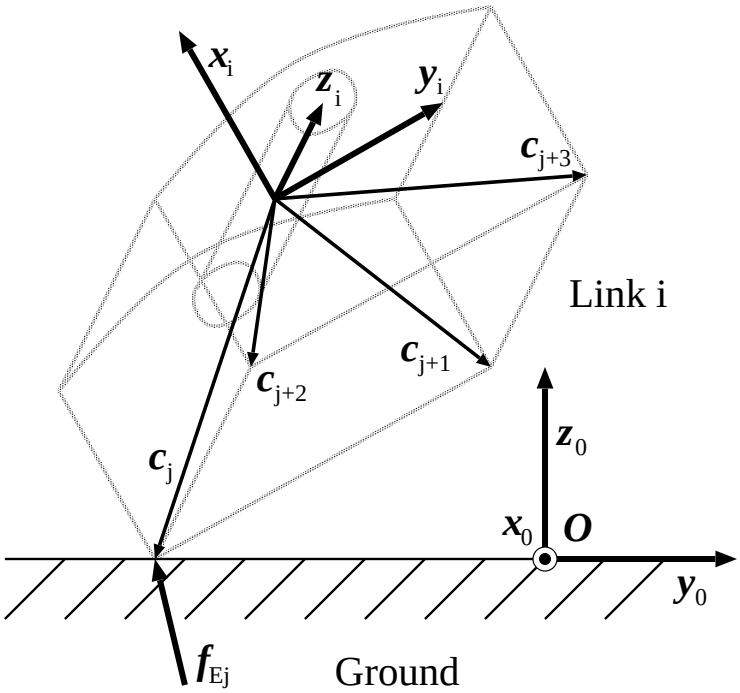


Figure 4.4: Contact points.

4.1.3 Exact Model of Environmental Force Interaction

In order to support the legged robot on the ground, the appropriate reactive forces from the ground should be determined at each time step in the simulation. In general, springs and dampers model at the contact points are introduced as a collision interaction. That simulation model, however, requires more shorter time step than the mechanical system itself does. Therefore we introduce a collision model based on a mathematical programming, which is the extension of the plastic collision model[19] and can deal with the three dimensional contact and arbitrary repulsion between plasticity and elasticity. In addition, a motion with the continuous collision behaves as an exact ideal constrained motion.

Since the generalized external force $\boldsymbol{u}_E$ is linear in terms of the jth external force $\boldsymbol{f}_{Ej}$, we can write

$$\boldsymbol{u}_E = \sum_{j \in M_A} \boldsymbol{K}_j \boldsymbol{f}_{Ej} = \boldsymbol{K} \boldsymbol{f}_E \tag{4.27}$$

where

$$M_A = \bigcup_{i=1}^{N} M_i \tag{4.28}$$

$\boldsymbol{K}_{Ej}$: $(N+6)\times 3$ matrix specifying transformer from jth external force to generalized forces
M_A : a set of index numbers of all active contact point
$\boldsymbol{f}_E$: $(3M)\times 1$ vector which consists of all components of all active contact forces
$\boldsymbol{K}$: $N\times(3M)$ matrix specifying transformer from $\boldsymbol{f}_E$ to generalized forces
M : number of active contact points which is time-variant

In convenience, we introduce three subsets of the external force $(\boldsymbol{f}_{Ex}, \boldsymbol{f}_{Ey}, \boldsymbol{f}_{Ez})$ which are decomposed in the Cartesian space.

$$\boldsymbol{f}_E^T = \left[\begin{array}{ccc} \boldsymbol{f}_{Ex}^T & \boldsymbol{f}_{Ey}^T & \boldsymbol{f}_{Ez}^T \end{array}\right] \tag{4.29}$$

$$\boldsymbol{K} = \left[\begin{array}{ccc} \boldsymbol{K}_x & \boldsymbol{K}_y & \boldsymbol{K}_z \end{array}\right] \tag{4.30}$$

where

$\boldsymbol{f}_{Ex}$, $\boldsymbol{f}_{Ey}$: $M\times 1$ vectors which consist of tangent components of all active external forces; $\boldsymbol{f}_{Ex}$ and $\boldsymbol{f}_{Ey}$ are orthogonal each other in the world coordinates.
$\boldsymbol{f}_{Ez}$: $M\times 1$ vector which consists of normal components of all active contact forces
$\boldsymbol{K}_x$, $\boldsymbol{K}_y$, $\boldsymbol{K}_z$: $N\times M$ matrices specifying transformers from $\boldsymbol{f}_{Ex}$, $\boldsymbol{f}_{Ey}$, and $\boldsymbol{f}_{Ez}$ to generalized forces, respectively

The dimensions of $\boldsymbol{K}$ and $\boldsymbol{f}_E$ are depend on the number of the contact points which is time-variant. In advance, $\boldsymbol{K}$ can be obtained by solving inverse dynamics mentioned above with setting $\boldsymbol{x}$ to the current state, $\boldsymbol{f}_E = \boldsymbol{e}_j$, $\dot{\boldsymbol{v}} = 0$, and ignoring gravity, centrifugal, and Corioli's effects.

When the external impulsive force

$$\Delta\boldsymbol{f} = h\boldsymbol{f}_E \tag{4.31}$$

are imposed in the system (2.7), the following equation is realized.

$$\boldsymbol{H}(\boldsymbol{v}_+ - \boldsymbol{v}_-) = \Delta\boldsymbol{f}_b + \boldsymbol{K}\Delta\boldsymbol{f} \tag{4.32}$$

where $\boldsymbol{v}_+$ and $\boldsymbol{v}_-$ denote the velocity after the collision and before the collision, respectively. $\Delta\boldsymbol{f}_b$ denotes the uncontrollable impulse force, i. e., $\Delta\boldsymbol{f}_b \approx h[\boldsymbol{u} - \boldsymbol{b}(\boldsymbol{x}, \boldsymbol{v}_-)]$.

Thus, the kinetic energy after the collision can be calculated as follows,

$$\begin{aligned} \frac{1}{2}\boldsymbol{v}_+^T\boldsymbol{H}\boldsymbol{v}_+ &= \frac{1}{2}\boldsymbol{v}_-^T\boldsymbol{H}\boldsymbol{v}_- + \frac{1}{2}\Delta\boldsymbol{f}^T\boldsymbol{K}^T\boldsymbol{H}^{-1}\boldsymbol{K}\Delta\boldsymbol{f} \\ &\quad +(\boldsymbol{v}_- + \boldsymbol{H}^{-1}\Delta\boldsymbol{f}_b)^T\boldsymbol{K}\Delta\boldsymbol{f} + \frac{1}{2}\Delta\boldsymbol{f}_b^T\boldsymbol{H}^{-1}\Delta\boldsymbol{f}_b \end{aligned} \tag{4.33}$$

The plastic collision is defined as the energy minimization with given conditions which yield a quadratic programming (QP) problem[19]. In this paper, the model is expanded to the following

problem to deal with three dimensional contact and arbitrary repulsion between plasticity and elasticity.

$$\underset{\Delta \boldsymbol{f}}{\text{minimize}} \quad \frac{1}{2}\Delta \boldsymbol{f}^T \boldsymbol{Q}_1 \Delta \boldsymbol{f} + \boldsymbol{q}_1^T \Delta \boldsymbol{f} \tag{4.34}$$

$$\text{subject to} \quad \begin{cases} \Delta f_{zi} & \geq & 0 \\ \sqrt{\Delta f_{xi}^2 + \Delta f_{yi}^2} & \leq & \mu \Delta f_{zi} \end{cases} \tag{4.35}$$

where

$$\boldsymbol{Q}_1 = \boldsymbol{K}^T \boldsymbol{H}^{-1} \boldsymbol{K} \tag{4.36}$$

$$\boldsymbol{q}_1 = \boldsymbol{K}^T (\boldsymbol{v}_- + \boldsymbol{H}^{-1} \Delta \boldsymbol{f}_b) + \begin{bmatrix} \boldsymbol{o} \\ \boldsymbol{o} \\ \lambda \boldsymbol{K}_z^T \end{bmatrix} \boldsymbol{v}_- \tag{4.37}$$

And λ represents a repulsion coefficient which is arbitrary set within $0 \leq \lambda \leq 1$.

The problem is equivalent to

$$\underset{\Delta \boldsymbol{f}_\theta}{\text{minimize}}\, F(\Delta \boldsymbol{f}_\theta) \tag{4.38}$$

where

$$F(\Delta \boldsymbol{f}_\theta) = \min_{\Delta \boldsymbol{f}_r, \Delta \boldsymbol{f}_z} \left(\frac{1}{2} \begin{bmatrix} \Delta \boldsymbol{f}_r \\ \Delta \boldsymbol{f}_z \end{bmatrix}^T \boldsymbol{Q}_2 \begin{bmatrix} \Delta \boldsymbol{f}_r \\ \Delta \boldsymbol{f}_z \end{bmatrix} + \boldsymbol{q}_2^T \begin{bmatrix} \Delta \boldsymbol{f}_r \\ \Delta \boldsymbol{f}_z \end{bmatrix} \right) \tag{4.39}$$

$$\text{subject to} \begin{cases} \Delta f_{zi} & \geq & 0 \\ |\Delta f_{ri}| & \leq & \mu \Delta f_{zi} \end{cases} \tag{4.40}$$

where

$$\boldsymbol{Q}_2 = \begin{bmatrix} \boldsymbol{K}_r^T \\ \boldsymbol{K}_z^T \end{bmatrix} \boldsymbol{H}^{-1} \begin{bmatrix} \boldsymbol{K}_r & \boldsymbol{K}_z \end{bmatrix} \tag{4.41}$$

$$\boldsymbol{q}_2 = \begin{bmatrix} \boldsymbol{K}_r^T \\ \boldsymbol{K}_z^T \end{bmatrix} (\boldsymbol{v}_- + \boldsymbol{H}^{-1} \Delta \boldsymbol{f}_b) + \begin{bmatrix} \boldsymbol{o} \\ \lambda \boldsymbol{K}_z^T \end{bmatrix} \boldsymbol{v}_- \tag{4.42}$$

$$\Delta f_{ri} = \sqrt{\Delta f_{xi}^2 + \Delta f_{yi}^2} \tag{4.43}$$

$$\Delta f_{\theta i} = \arctan \Delta f_{yi} / \Delta f_{xi} \tag{4.44}$$

$$\boldsymbol{K}_r = \boldsymbol{K}_x \cos \Delta \boldsymbol{f}_\theta + \boldsymbol{K}_y \sin \Delta \boldsymbol{f}_\theta \tag{4.45}$$

The constraints of the sub-problem become linear. Then the problem can be solved by non-constraint optimization algorithm (for example quasi Newton method) with the sub-problem of the quadratic programming. In the optimization (4.38), the initial search point of $\Delta \boldsymbol{f}_\theta$ is set to the opposite directions to the velocities of the contact points projected onto the ground, which is an approximate solution of $\Delta \boldsymbol{f}_\theta$.

Characteristics of Solution

The solution of the problem satisfy the Kuhn-Tucker's necessary and sufficient conditions as follows:

$$\mu\ \Delta f^*_{ri} \geq |\Delta f^*_{zi}| \tag{4.46}$$

$$\Delta f^*_{ri}\ \boldsymbol{k}^T_{ri}\ \boldsymbol{v}_+ \leq 0 \tag{4.47}$$

$$(\mu\ \Delta f^*_{zi} - |\Delta f^*_{ri}|)\ \boldsymbol{k}^T_{ri}\ \boldsymbol{v}_+ = 0 \tag{4.48}$$

$$\Delta f^*_{zi} \geq 0 \tag{4.49}$$

$$\boldsymbol{k}^T_{zi}\ \boldsymbol{v}_+ \geq -\lambda \boldsymbol{k}^T_{zi}\ \boldsymbol{v}_- \tag{4.50}$$

$$\begin{cases} \Delta f^*_{zi}\ (\boldsymbol{k}^T_{zi}\ \boldsymbol{v}_+ + \lambda \boldsymbol{k}^T_{zi}\ \boldsymbol{v}_-) = 0 & \text{if } |\Delta f^*_{ri}| < \mu\ \Delta f^*_{zi} \\ \boldsymbol{k}^T_{zi}\ \boldsymbol{v}_+ + \lambda \boldsymbol{k}^T_{zi}\ \boldsymbol{v}_- \geq \mu\ |\boldsymbol{k}^T_{ri}\ \boldsymbol{v}_+| & \text{if } |\Delta f^*_{ri}| = \mu\ \Delta f^*_{zi} \end{cases} \tag{4.51}$$

where $\boldsymbol{k}_{ri}$ and $\boldsymbol{k}_{zi}$ denote ith row of the matrices $\boldsymbol{K}_r$ and $\boldsymbol{K}_z$, respectively.

The conditions (4.46) to (4.48) are related to the tangent motion of the contact points. Note that $\boldsymbol{k}^T_{ri}\boldsymbol{v}_+$ expresses the tangent velocity of the contact point after the collision due to the principle of the virtual work. Especially, (4.48) means that the friction condition is activated ($|\Delta f^*_{ri}| = \mu\ \Delta f^*_{zi}$) if the contact point slips. If the friction condition is not activated ($|\Delta f^*_{ri}| < \mu\ \Delta f^*_{zi}$), the tangent velocity at the contact point becomes zero.

The conditions (4.49) to (4.51) are related to the normal motion of the contact points. The condition (4.50) means that in the proposed method the solution of the contact force realizes arbitrary repulsion between plasticity and elasticity, since $\boldsymbol{k}^T_{zi}\ \boldsymbol{v}_+$ represents the normal velocity of the contact point after the collision and $\boldsymbol{k}^T_{zi}\ \boldsymbol{v}_-$ corresponds to that before the collision, respectively. The motion with the proposed collision behaves an exact ideal constrained motion.

4.1.4 Procedure of Forward Dynamic Simulation

To summarize the forward dynamic simulation method in this section, a list of the procedure is shown as follows.

1. set the initial state as the current state.
2. calculate the inertia matrix $\boldsymbol{H}(\boldsymbol{x})$ numerically by solving inverse dynamics (4.8)–(4.26) with setting $\boldsymbol{x}$ to the current state and $\dot{\boldsymbol{v}} = \boldsymbol{e}_j$ for $1 \leq j \leq N+6$, and ignoring centrifugal forces, Corioli's forces, gravity effects, and external forces.
3. calculate the biasing vector $\boldsymbol{b}(\boldsymbol{x}, \boldsymbol{v})$ numerically by solving inverse dynamics (4.8)–(4.26) with setting $(\boldsymbol{x}, \boldsymbol{v})$ to the current state and $\dot{\boldsymbol{v}} = 0$, and ignoring external forces.
4. calculate the transformer matrix $\boldsymbol{K}$ numerically by solving inverse dynamics (4.8)–(4.26) with setting $\boldsymbol{x}$ to the current state, $\boldsymbol{f}_E = \boldsymbol{e}_j$ for $1 \leq j \leq M$, and $\dot{\boldsymbol{v}} = 0$, and ignoring gravity, centrifugal, and Corioli's effects.
5. calculate the external force $\boldsymbol{f}_E$ by solving mathematical programming problem (4.38)–(4.45) and eq. (4.31)
6. calculate the generalized acceleration $\dot{\boldsymbol{v}}$ by (4.27) and (4.5)
7. update the generalized states $(\boldsymbol{x}, \boldsymbol{v})$ by (4.1)–(4.4) and (4.7)
8. return to 2.

Using this procedure, we can completely simulate a 3-dimensional dynamic motion of a legged robot. This method is mathematically exact, thus this simulator will useful for essential investigation of various control algorithms of legged systems.

4.2 Inverse Dynamic Simulator

This section shows a method to analyze joint torques of a legged robot and interaction forces from the ground when the trajectories of the position, the velocity, and the acceleration of the joint angles, those of the body, and those of the body attitude are given.

4.2.1 Interaction Force Calculation

In detail, the problem is solution for the joint torque $\boldsymbol{\tau}$ and the external force $\boldsymbol{u}_E$ given the generalized coordinates $\boldsymbol{x}$, the generalized velocity $\boldsymbol{v}$, and the generalized acceleration $\dot{\boldsymbol{v}}$ in (2.1)–(2.7). When $\boldsymbol{x}$, $\boldsymbol{v}$, and $\dot{\boldsymbol{v}}$ are given, we can completely calculate the left side of the eq. (2.7) using the Newton-Euler method mentioned in the section 4.1.2 by setting external forces $\boldsymbol{f}_E = 0$. Let the results of the calculation of the left side be $\boldsymbol{u}_a$ as follows.

$$\boldsymbol{u}_a(\boldsymbol{x}, \boldsymbol{v}, \dot{\boldsymbol{v}}) = \boldsymbol{H}(\boldsymbol{x})\dot{\boldsymbol{v}} + \boldsymbol{C}(\boldsymbol{x}, \boldsymbol{v})\boldsymbol{v} + \boldsymbol{g}(\boldsymbol{x}) \tag{4.52}$$

which corresponds to the generalized forces generated by inertial forces, centrifugal forces, Coriolis's forces, and gravity effects. Then the following equation exists from (2.7) and (4.27).

$$\boldsymbol{u}_a = \boldsymbol{u} + \boldsymbol{K}\boldsymbol{f}_E \tag{4.53}$$

Here, $\boldsymbol{K}$ can be calculated in the same way mentioned in the section 4.1.3 because the active contact points are determined from the state of the robot $\boldsymbol{x}$ and the kinematic property of the robot.

This equation is equivalent to

$$\begin{bmatrix} \boldsymbol{f}_a \\ \boldsymbol{n}_a \\ \boldsymbol{\tau}_a \end{bmatrix} = \begin{bmatrix} \boldsymbol{f}_B \\ \boldsymbol{n}_B \\ \boldsymbol{\tau} \end{bmatrix} + \begin{bmatrix} \boldsymbol{K}_f \\ \boldsymbol{K}_n \\ \boldsymbol{K}_\tau \end{bmatrix} \boldsymbol{f}_E \tag{4.54}$$

where

$$\boldsymbol{u}_a^T = \begin{bmatrix} \boldsymbol{f}_a^T & \boldsymbol{n}_a^T & \boldsymbol{\tau}_a^T \end{bmatrix} \tag{4.55}$$

$$\boldsymbol{K}^T = \begin{bmatrix} \boldsymbol{K}_f^T & \boldsymbol{K}_n^T & \boldsymbol{K}_\tau^T \end{bmatrix}. \tag{4.56}$$

Since there is no actuated force $\boldsymbol{f}_B$ and torque $\boldsymbol{n}_B$ on the body of the robot, those must be zero:

$$\boldsymbol{f}_B = \boldsymbol{n}_B = 0. \tag{4.57}$$

Thus the eq. (4.54) becomes

$$\begin{bmatrix} \boldsymbol{f}_a \\ \boldsymbol{n}_a \end{bmatrix} = \begin{bmatrix} \boldsymbol{K}_f \\ \boldsymbol{K}_n \end{bmatrix} \boldsymbol{f}_E \tag{4.58}$$

$$\boldsymbol{\tau}_a = \boldsymbol{\tau} + \boldsymbol{K}_\tau \boldsymbol{f}_E. \tag{4.59}$$

In a case that the matrix

$$\boldsymbol{K}_{fn} = \begin{bmatrix} \boldsymbol{K}_f \\ \boldsymbol{K}_n \end{bmatrix} \tag{4.60}$$

has full row rank, the solution for the external force $\boldsymbol{f}_E$ exists in (4.58). The minimal norm solution of $\boldsymbol{f}_E$ is obtained by

$$\boldsymbol{f}_E = \boldsymbol{K}_{fn}^T(\boldsymbol{K}_{fn}\boldsymbol{K}_{fn}^T)^{-1}\begin{bmatrix}\boldsymbol{f}_a\\ \boldsymbol{n}_a\end{bmatrix}. \tag{4.61}$$

When the matrix $\boldsymbol{K}_{fn}$ has full column rank and does not have full row rank, there might be no solution in (4.58). In this case, the approximate solution which has minimal norm error is given by

$$\boldsymbol{f}_E = (\boldsymbol{K}_{fn}^T\boldsymbol{K}_{fn})^{-1}\boldsymbol{K}_{fn}^T\begin{bmatrix}\boldsymbol{f}_a\\ \boldsymbol{n}_a\end{bmatrix}. \tag{4.62}$$

Then the joint torque generated by the actuator can be obtained as follows from (4.59).

$$\boldsymbol{\tau} = \boldsymbol{\tau}_a - \boldsymbol{K}_\tau \boldsymbol{f}_E \tag{4.63}$$

4.2.2 Procedure of Inverse Dynamic Simulation

To summarize the inverse dynamic simulation method in this section, a list of the procedure is shown as follows.

1. calculate the generalized force $\boldsymbol{u}_a$ numerically by solving inverse dynamics (4.52) and (4.8)–(4.26) with setting $(\boldsymbol{x}, \boldsymbol{v}, \dot{\boldsymbol{v}})$ to current state and ignoring external forces
2. calculate the transformer matrix $\boldsymbol{K}$ numerically by solving inverse dynamics (4.8)–(4.26) with setting $\boldsymbol{x}$ to the current state, $\boldsymbol{f}_E = \boldsymbol{e}_j$ for $1 \leq j \leq M$, and $\dot{\boldsymbol{v}} = 0$, and ignoring gravity, centrifugal, and Corioli's effects.
3. calculate the external force $\boldsymbol{f}_E$ by solving (4.61) or (4.62) depending on the rank of $\boldsymbol{K}_{fn}$
4. calculate the joint torque $\boldsymbol{\tau}$ by solving (4.63)
5. return to 1.

Using this procedure, we can completely analyze joint torques of a legged robot and interaction forces from the ground when the trajectories of the position, the velocity, and the acceleration of the joint angles, those of the body, and those of the body attitude are given. This inverse dynamic simulation method is useful in investigating legged systems when time series data of joint angles are given, which might be measured from human walking or generated by various trajectory generators.

Chapter 5

Simulation Results

5.1 21-link Biped Walking

The proposed control is applied to 20 axes human-type biped robot and is investigated by the proposed precise 3D dynamic simulator. The parameters of the robot is shown in Table 5.1. The friction coefficient μ and restitution coefficient λ on the environment is set to $\mu = 1.0$ and $\lambda = 0.0$, respectively. The control parameter is set as follows. The walking velocity: 0.25 [m/s], the walking period: 0.5 [sec/step], the double support period: 0.05 [sec], and the characteristic polynomial: $(z - 0.3)^3$. The QP is solved by the algorithm in [2]. In the force distribution controller, the upper bound of the friction coefficient is set to 0.5. The control period is 1 [ms].

The snapshots of the simulation is shown in Fig. 5.1. The initial movement of COM is finished between [0, 1]. After that, the walking motion starts. Fig. 5.2 shows the trajectory of the zero moment point (ZMP). Fig. 5.3 shows the response of COM with the reference trajectory, in which the tracking error converges to zero. From the magnified figures (c) and (d), the COM follows the inverted pendulum model even among the stepping period.

Table 5.1: Parameters of biped robot.

parts	size [m]	weight [kg]
all	0.99	28.744
head	$0.14 \times 0.14 \times 0.14$ ($d \times w \times h$)	2.744
arm	0.3	3.5
body	0.4	8
thigh	0.2	2
shin	0.2	2
foot	0.2×0.1 ($d \times w$)	1.5

Figure 5.1: Snapshots of biped walking simulation. (More details are found in the web, http://www.kawalab.dnj.ynu.ac.jp/)

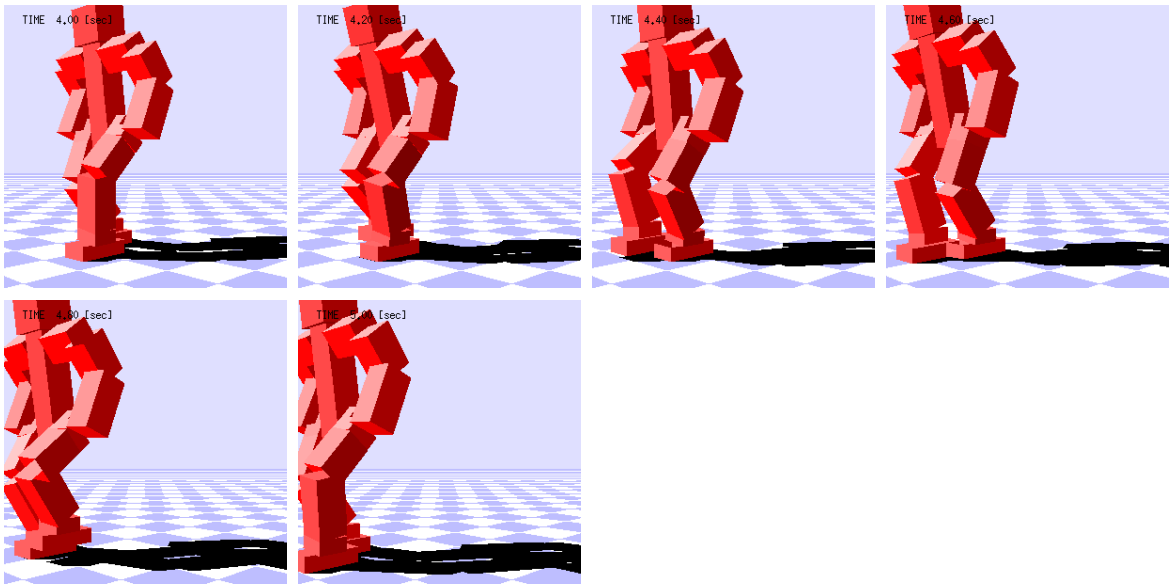
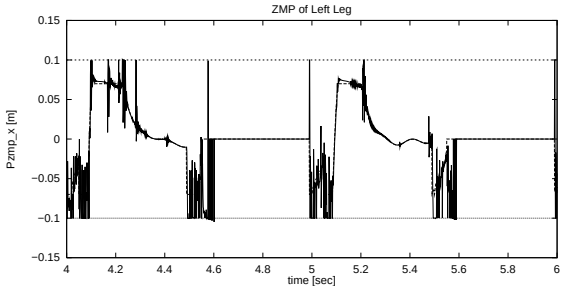
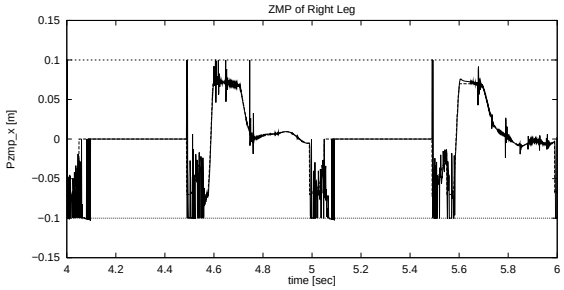


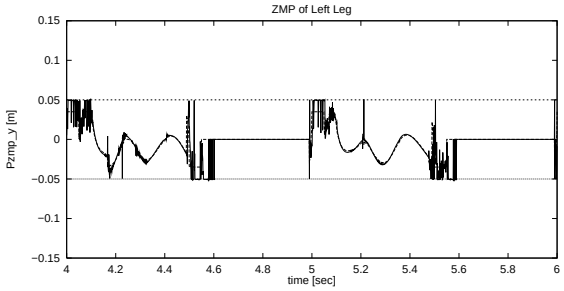
Figure 5.1: Snapshots of biped walking simulation. (continued.)



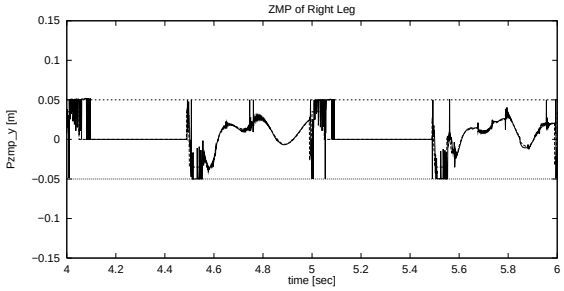
(a) ZMP of left foot in sagittal plane.



(b) ZMP of right foot in sagittal plane.

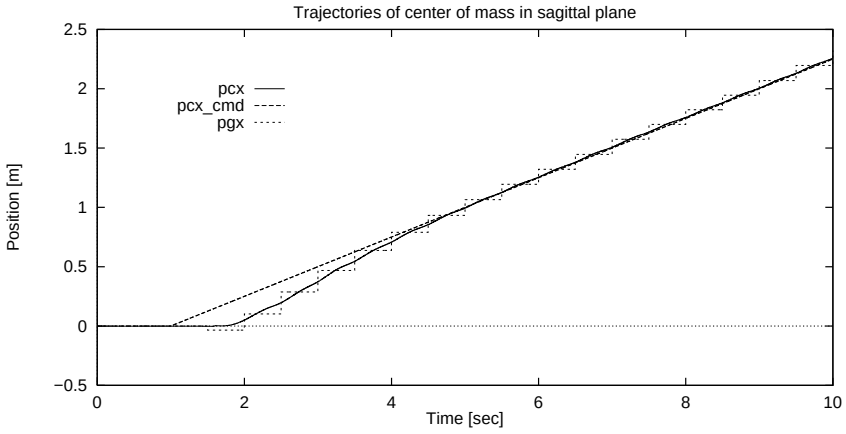


(c) ZMP of left foot in lateral plane.

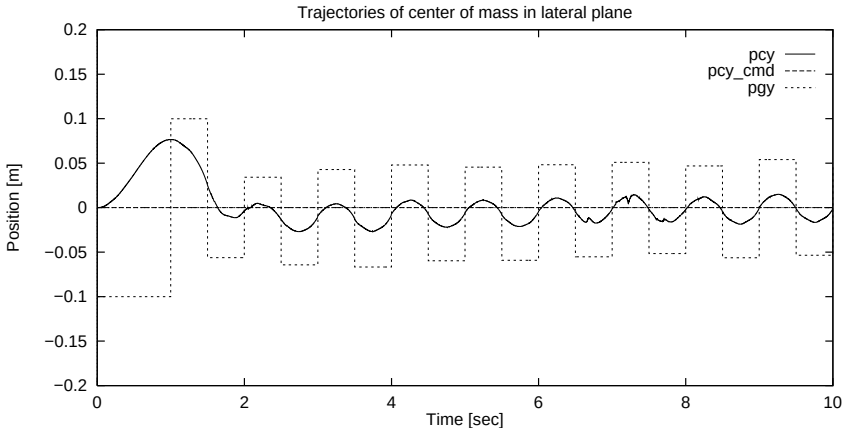


(d) ZMP of right foot in lateral plane.

Figure 5.2: Trajectory of zero moment point.

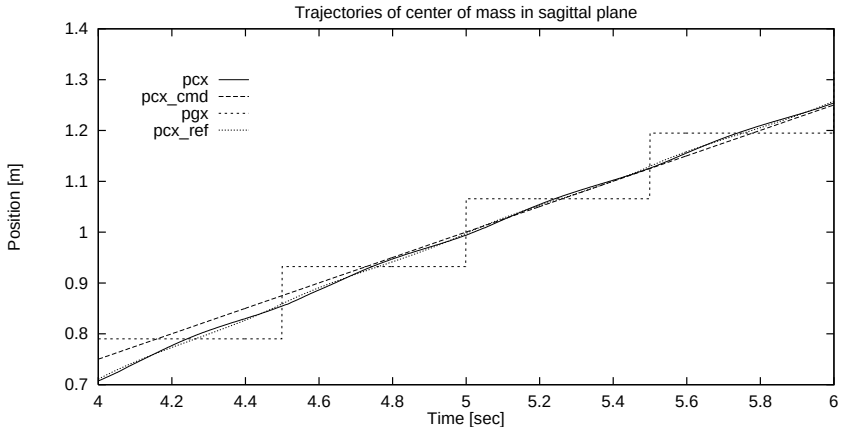


(a) In sagittal plane.

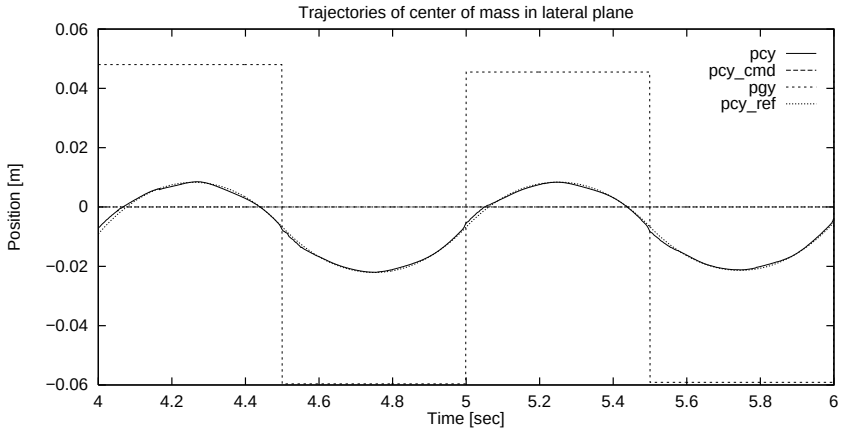


(b) In lateral plane.

Figure 5.3: Trajectory of center of mass of the robot.



(c) In sagittal plane (magnified.)



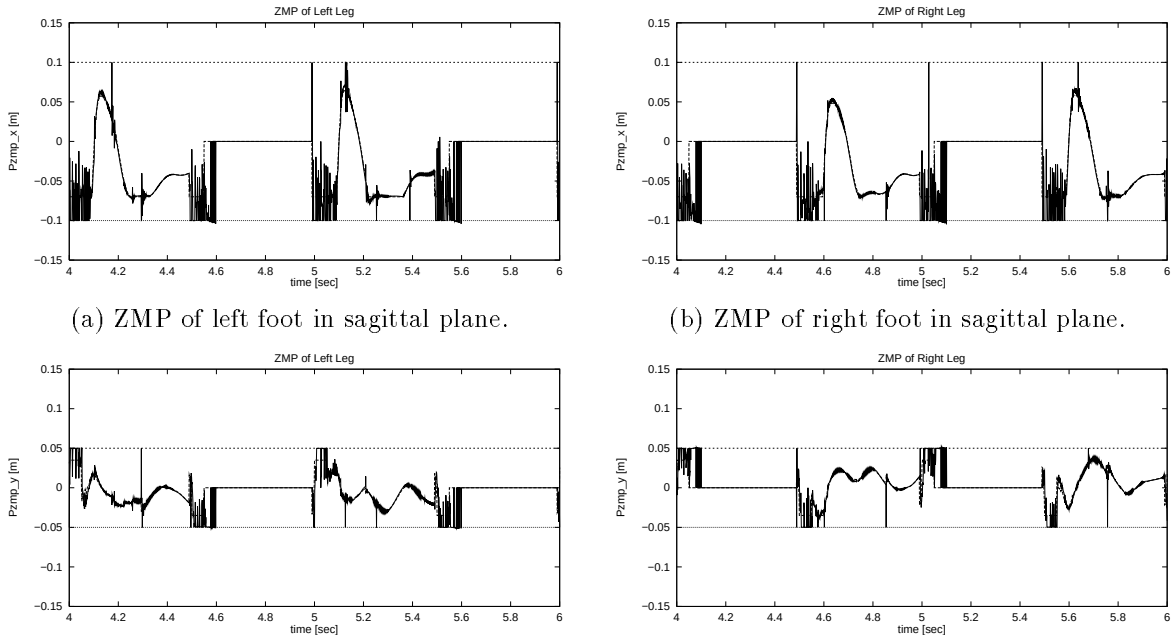
(d) In lateral plane (magnified.)

Figure 5.3: Trajectory of center of mass of the robot (magnified.)

5.2 Walking on Unknown Slope

This section shows the biped walking simulation with slope environment whose information is not used in the controller. The controller used in the simulation is exactly as same as that of in the section 5.1. Thus the proposed control algorithm is robust against the environmental uncertainty.

The slope is set to 5 [deg] up, which is not used in the controller. Fig. 5.4 shows the trajectory of the ZMP. Due to the slope, the trajectory of the ZMP shifts to the heel in Fig. 5.4 (a) and (b), compared with the flat terrain case in Fig. 5.2 (a) and (b). Fig. 5.5 shows the response of COM with the reference trajectory.



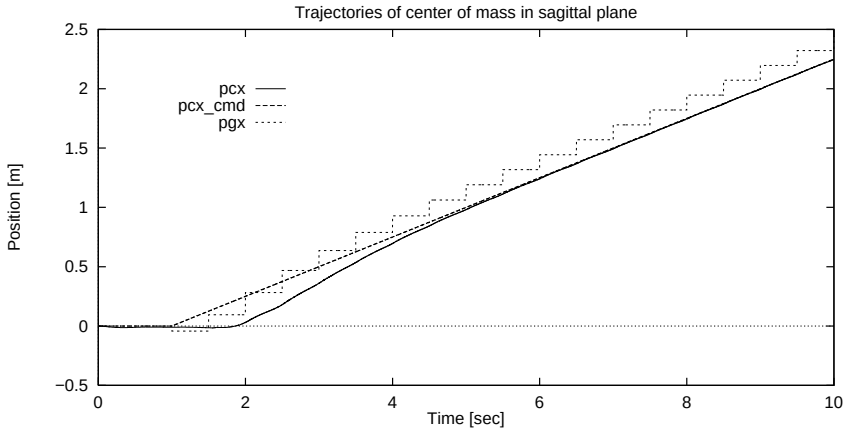
(a) ZMP of left foot in sagittal plane.

(b) ZMP of right foot in sagittal plane.

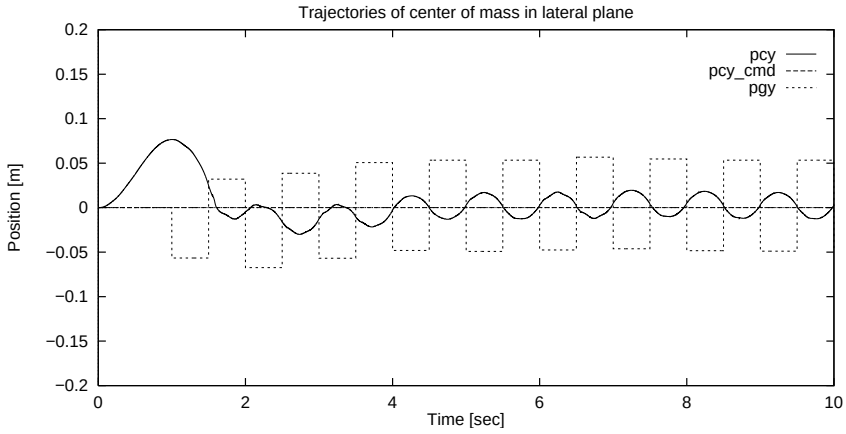
(c) ZMP of left foot in lateral plane.

(d) ZMP of right foot in lateral plane.

Figure 5.4: Trajectory of zero moment point.

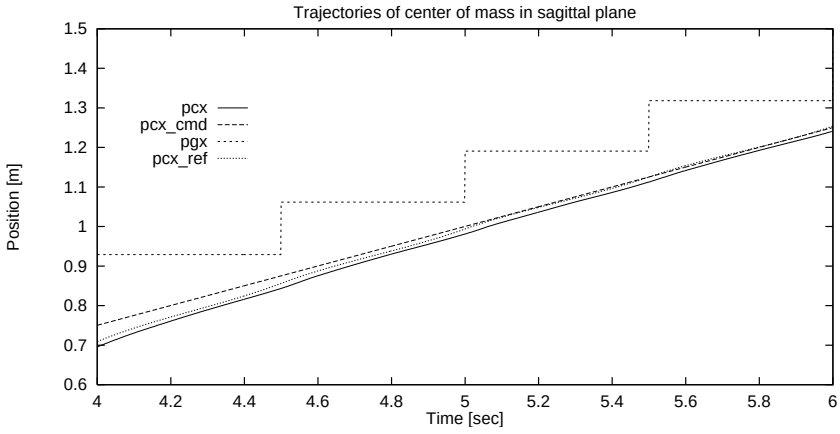


(a) In sagittal plane.

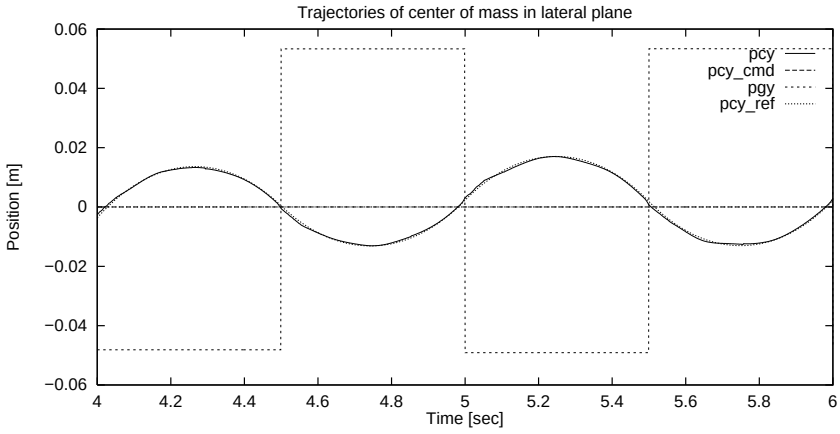


(b) In lateral plane.

Figure 5.5: Trajectory of center of mass of the robot.



(c) In sagittal plane (magnified.)



(d) In lateral plane (magnified.)

Figure 5.5: Trajectory of center of mass of the robot (magnified.)

5.3 Walking on Low μ Terrain

In this section, the effectiveness of proposed compensation of yaw axis moment by arm swing motion is shown. The yaw moment gets large when the walking speed gets fast. It becomes beyond the moment which the foot can generate and the walking gets unstable. The same situation will occur in a case of a walking on low μ terrain.

The friction coefficient μ and restitution coefficient λ on the environment is set to $\mu = 0.2$ and $\lambda = 0.0$, respectively. In the force distribution controller, the upper bound of the friction coefficient is set to 0.15.

Fig. 5.6 shows the required yaw moment to maintain the posture by (3.50) and the constrained yaw moment by (3.68). The stabilization of the contact yields a large error between them. Fig. 5.7(a) shows that the yaw rotation gets unstable during walk when there is no compensation of yaw moment. It is stabilized by arm motion shown in Fig. 5.7(b). Fig. 5.8 shows the snapshots of the simulation. The stable walking with a velocity 0.25 m/sec and a stepping time 0.5 sec/step is realized.

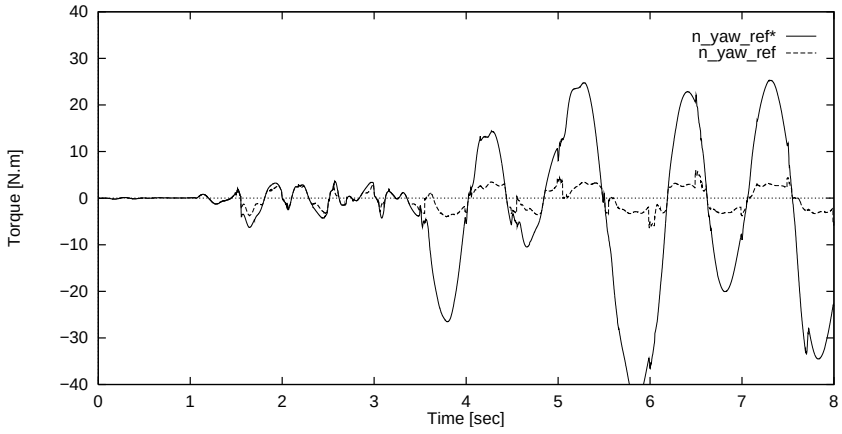
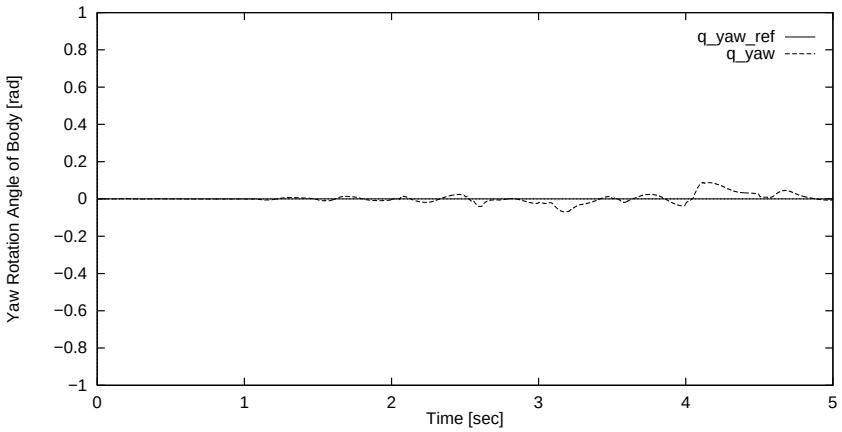
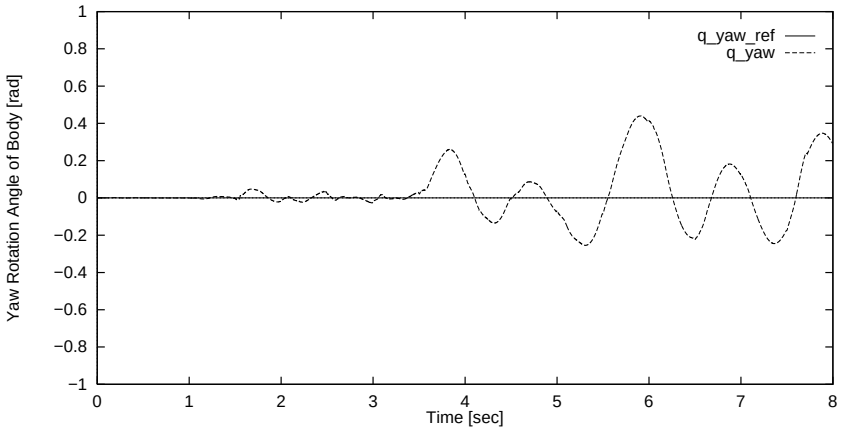


Figure 5.6: Required yaw moment and realized one. (in a case of no compensation)



(a) with compensation by arms.



(b) without compensation.

Figure 5.7: Rotation around yaw axis.

Figure 5.8: Snapshots of biped walking simulation with arm swing motion. (More details are found in the web, http://www.kawalab.dnj.ynu.ac.jp/)

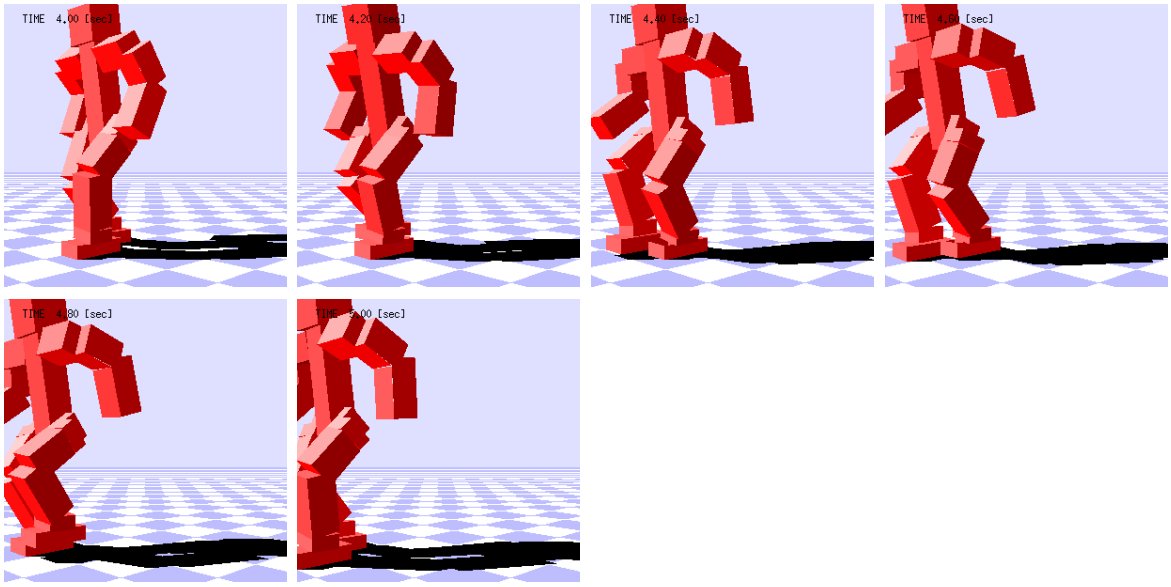


Figure 5.8: Snapshots of biped walking simulation with arm swing motion. (continued.)

Chapter 6

Experiments

6.1 Hardware Design

6.1.1 Specifications

The 14-axis biped robot has been developed as shown in Fig. 6.1. The number of axes is reduced from 20 to 14 due to the weight capacity of the robot.

The specification of the robot is as follows.

- 6-degree-of-freedom for each leg.
- 1-degree-of-freedom for each arm.
- Dc servo motor with 50:1 harmonic gear.
- Rotary encoder sensor for each joints.
- 6DOF force/torque sensor on each ankle.
- 3DOF gyro scope.
- Controller: DSP (TMS320C32-50MHz) $\times$ 2

The robot has 6 joints for each leg so that the position and orientation of the foot can be chosen any posture in the 3-dimensional space. The freedom of arms is 1 for each arm. Dc servo motors with 50:1 harmonic gear is used as the actuators, which have rotary encoders of resolution 1000 [pulse/rev] on the motor shafts. The encoder counter enables the resolution four times as precise as original one. The 6DOF force/torque sensors are attached to both ankles. Also 3DOF gyro scope is attached to the body which senses the attitude of the body. All calculation of the control is done by DSP board with TMS320C32-50MHz. The programs can be written in the C language.

The dimension of the robot is shown in Fig. 6.2. The robot is about 1.2 [m] height and 20 [kg] weight.

Fig. 6.3 shows the global system configuration. The host computer is used in cross compiling of DSP programs.

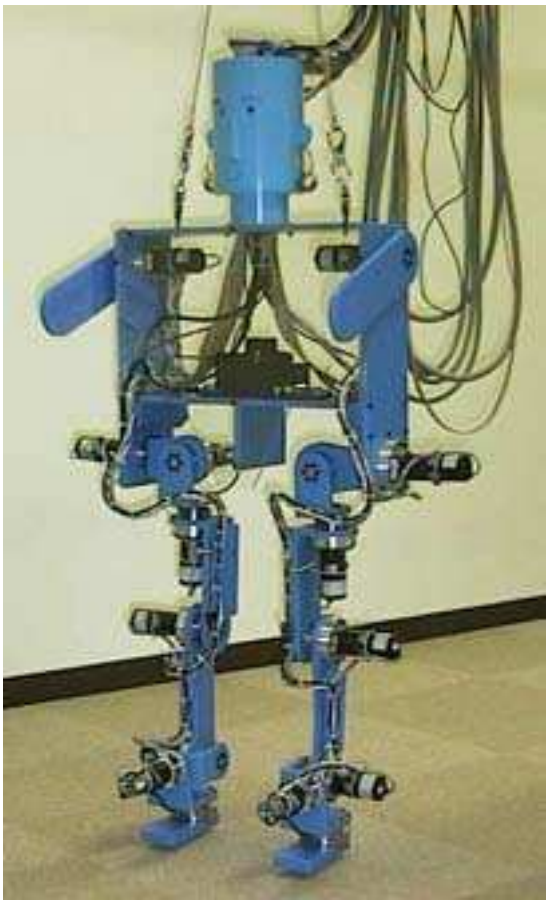


Figure 6.1: A photo of 14-axis biped robot.

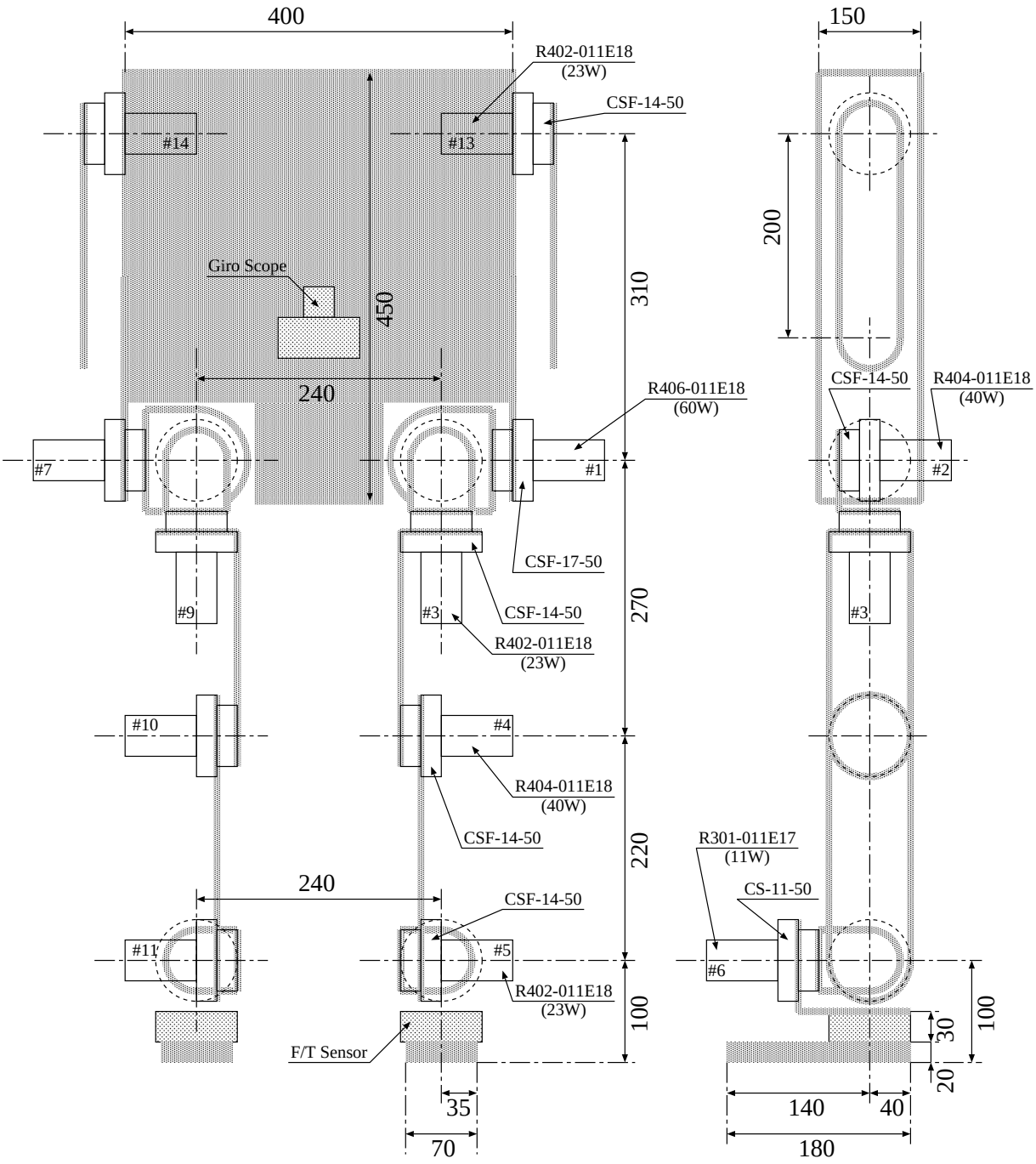


Figure 6.2: Dimension of 14-axis biped robot.

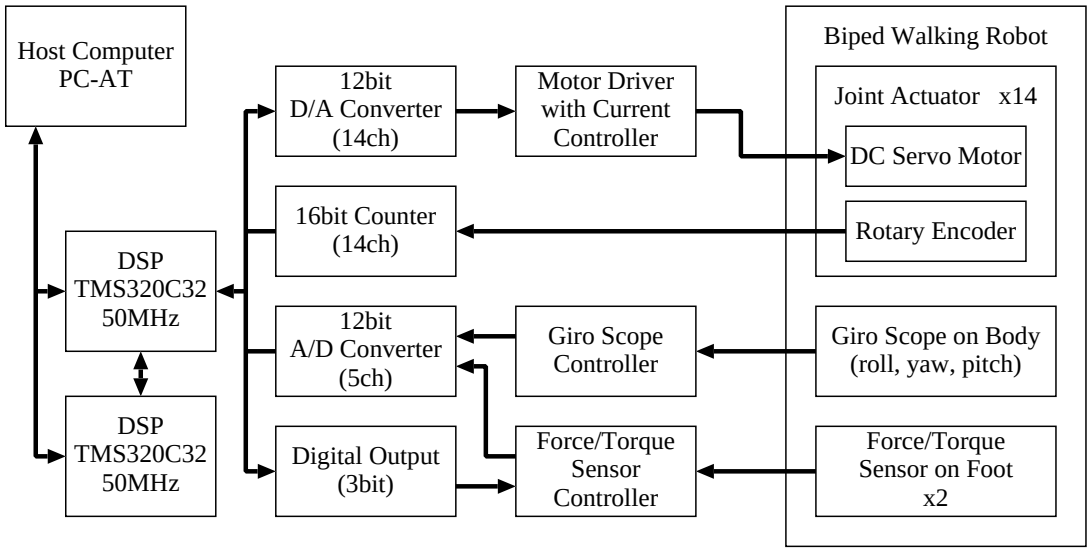


Figure 6.3: System configuration.

6.1.2 Actuators

The rated power and torque were estimated from simulations as shown in Table 6.1. Here the rated value means the average value. Then dc motors with harmonic gears (50:1) were selected as shown in Table 6.2.

These actuators are little rack of the power and torque. But this is alternative selection since the high power actuators become heavy.

Table 6.1: Estimated rated power and torque of joints from simulations.

Joint No.	Rated Power [W]	Rated Torque [N·m]
# 1, # 7	80	42
# 2, # 8	17	41
# 3, # 9	12	24
# 4, #10	37	45
# 5, #11	28	22
# 6, #12	4	11
#13, #14	1	7

Table 6.2: Selected motors and gears.

Joint No.	Power [W]	Rated Torque [N·m]	Peak Torque [N·m]
#1, #7	60	9.3	62.2
#2, #4, #8, #10	40	6.6	39.2
#3, #5, #13, #9, #11, #14	23	3.7	21.1
#6, #12	11	1.76	7.4

6.2 Implementation Aspects

Although computing ability of recent micro processor progresses rapidly, it is not enough to implement all of the proposed control algorithm in real-time. Thus some simplified algorithms are shown in this section. As the result, the control period becomes 2 [ms].

6.2.1 Robust Servo Control in Joint Space

Since the calculation of the proposed joint control shown in 3.1 costs too much for 14-axis robot, the nominal inertia matrix is fixed to constant.

The parameters of H_∞ control were chosen as $\omega_c = 500[rad/s]$ and $\lambda = 0.9$. Fig. 6.4 shows the bode diagram of the cost functions, the sensitivity functions, and the complementary sensitivity functions in this case. Finally the cutoff frequency of the sensitivity function becomes 197 [rad/s].

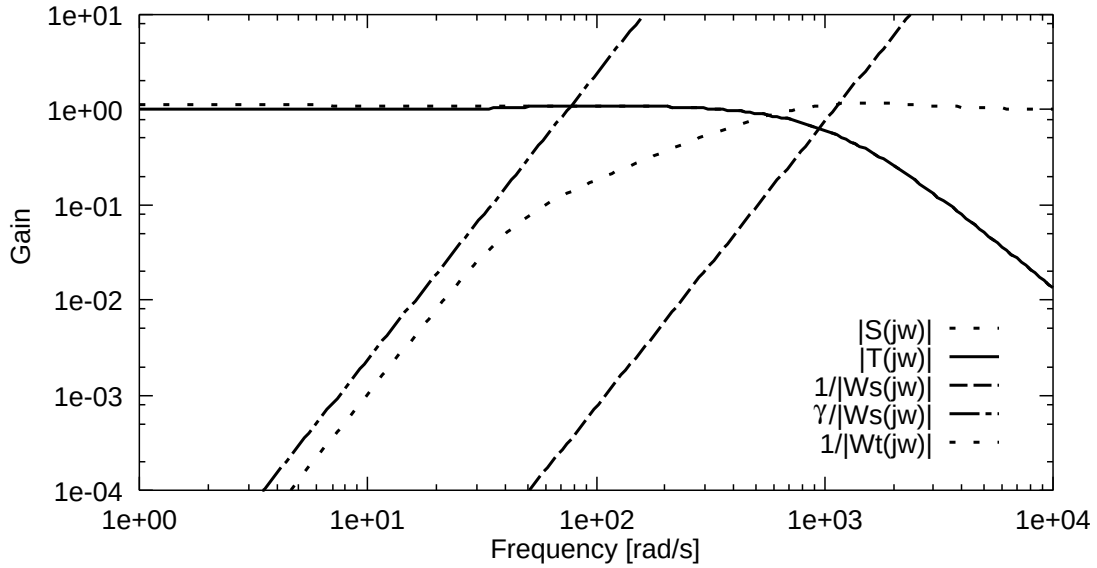


Figure 6.4: Frequency characteristics of cost functions, sensitivity function, and complementary sensitivity function in experiments

The controller becomes as

$$C(s) = \frac{3.1494 \times 10^{10} s^2 + 3.9385 \times 10^{12} s + 7.8906 \times 10^{13}}{s^3 + 1.1441 \times 10^5 s^2 + 1.1679 \times 10^8 s - 6.3471 \times 10^{-6}}. \tag{6.1}$$

Then, after digitalizing the controller by the bilinear transformation, we have a canon form as

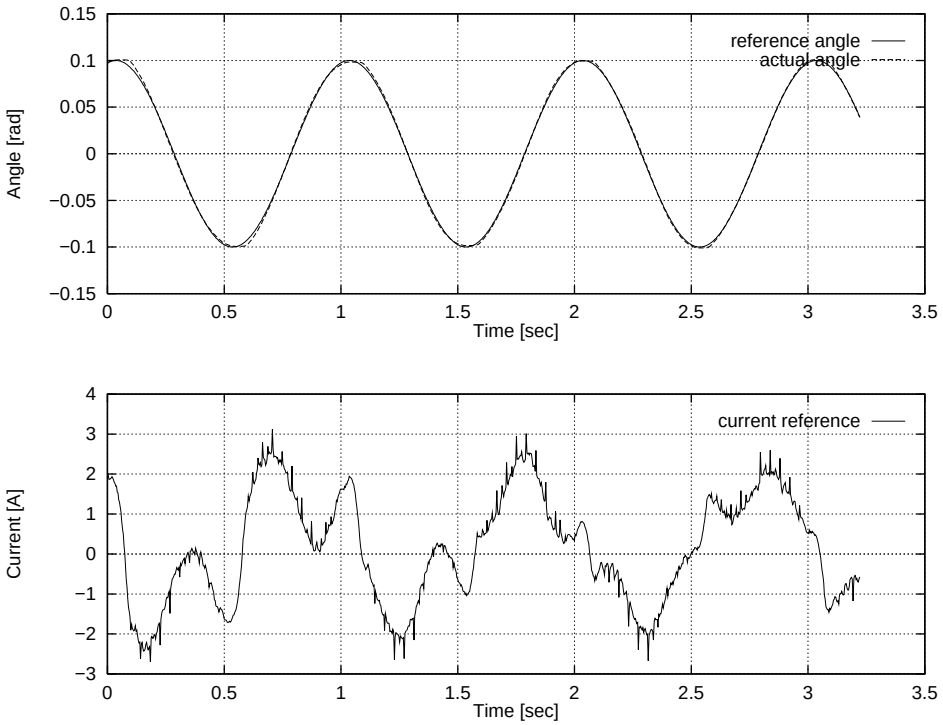
$$A_c = \begin{bmatrix} 0.3548 & 0.9542 & -0.3090 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix} \tag{6.2}$$

$$b_c^T = [1\ 0\ 0] \tag{6.3}$$

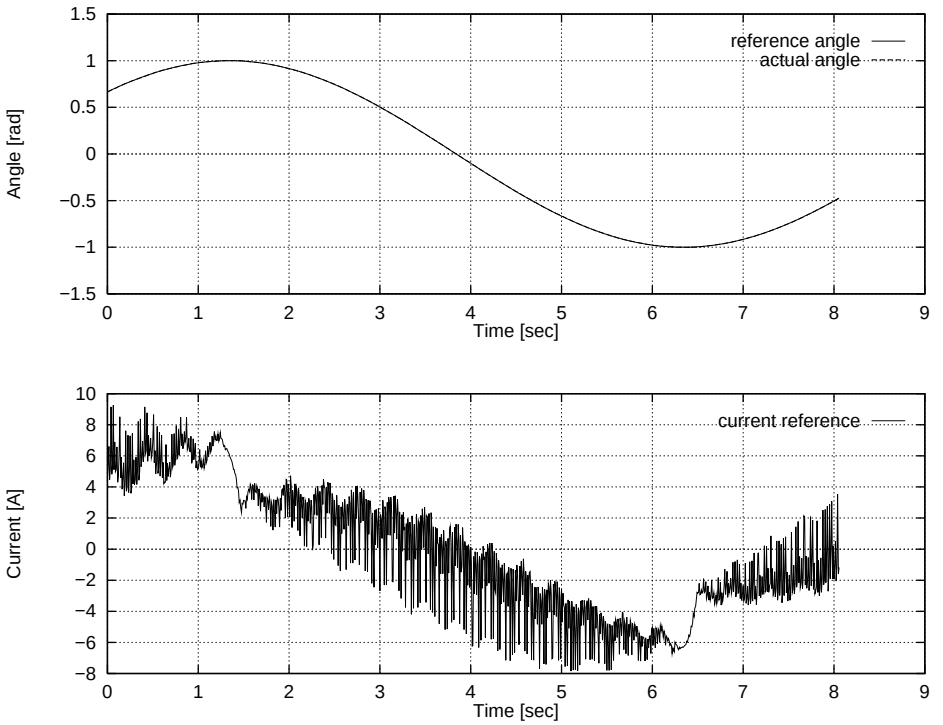
$$c_c^T = [-1.006 \times 10^5 \ -8.315 \times 10^3 \ 1.098 \times 10^5] \quad (6.4)$$

$$d_c = 1.915 \times 10^5 \quad (6.5)$$

The frequency responses of the servo system have been measured as shown in Fig. 6.5. The tracking performance is very accurate.



(a) Amplitude 1 [rad], Frequency 0.1 [Hz]



(b) Amplitude 0.1 [rad], Frequency 1 [Hz]

Figure 6.5: H_∞ joint servo control.

6.2.2 Force Control in Environment

The simple force control in Cartesian Space is implemented in each foot. In the experiments, the free parameter and the nominal plant model in (3.35) and (3.36) were set as

$$Q_{fi}(s) = \frac{1}{\tau_i s + 1} \tag{6.6}$$

$$P_{fni}(s) = b_i s + k_i \tag{6.7}$$

$i = 1, 2, \ldots, 6$. Then the robust force controller is obtained from (3.33) as follows.

$$\boldsymbol{C}_f(s) = \mathrm{diag}\{C_{f1}(s), \ldots, C_{f6}(s)\} \tag{6.8}$$

$$C_{fi}(s) = \frac{1}{\tau_i s(b_i s + k_i)} \tag{6.9}$$

Table 6.3 shows the values of the parameters.

Table 6.3: Parameters of force controller.

index	k	b	τ
1	10000	200	0.0133
2	10000	200	0.0133
3	10000	200	0.025
4	16	0.32	0.313
5	100	2	0.0667
6	100	2	0.02

Fig. 6.6 shows the reactive force response between foot and ground. The reference values are set to 0 [N] in x and y axes and -50 [N] in z axis. The tracking errors rapidly converged on zero.

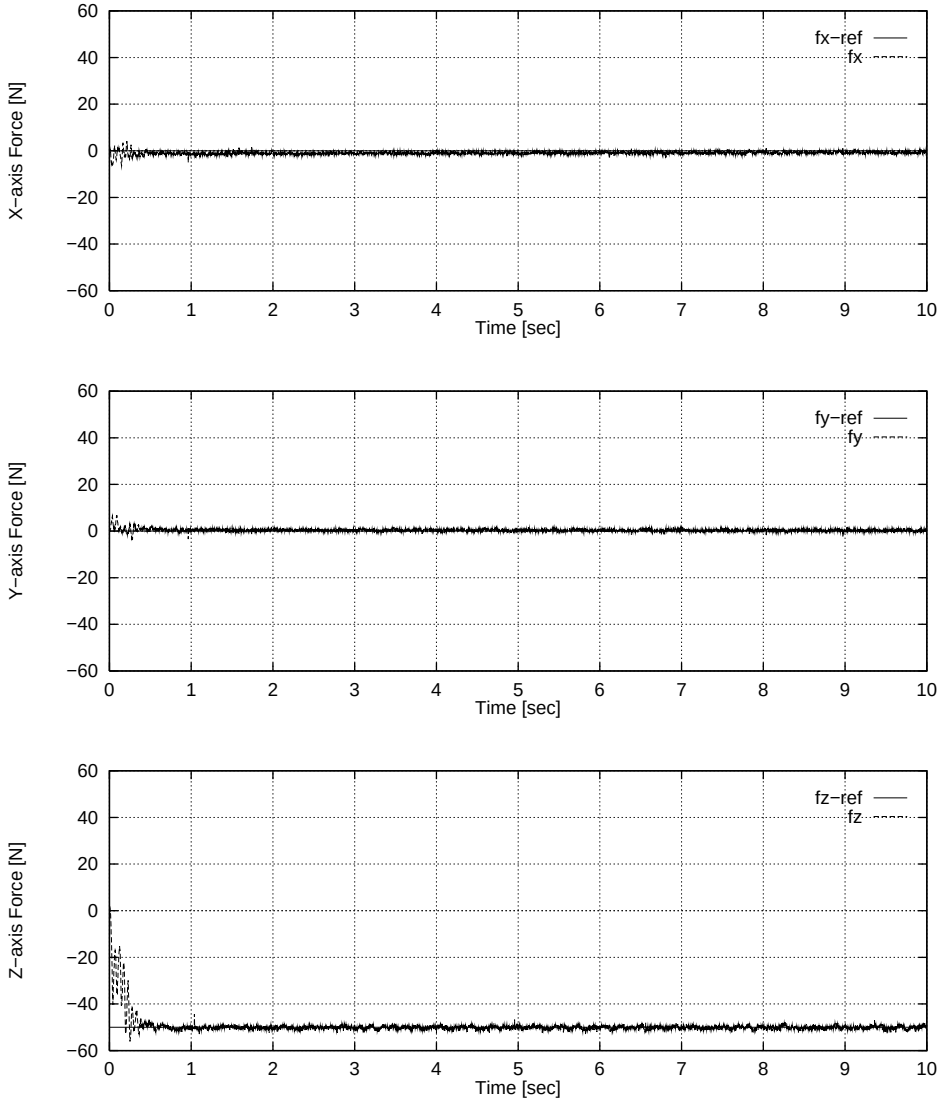


Figure 6.6: Force control on foothold.

6.2.3 Posture Control with Optimal Force Distribution

The calculation of quadratic programming costs very much. Then it is not suitable for realtime control for the present.

The simplified method is introduced as follows. Consider the simplified model of the motion of the COM (center of mass) of the robot and the rotational motion of the body (3.43) and (3.44) as

$$
\begin{aligned}
m\ddot{p}_{Cx} &= f_{Cx} && (6.10)\\
m\ddot{p}_{Cy} &= f_{Cy} && (6.11)\\
m\ddot{p}_{Cz} + mg &= f_{Cz} && (6.12)\\
J_x\dot{\omega}_{Bx} &= n_{Bx} && (6.13)\\
J_y\dot{\omega}_{By} &= n_{By} && (6.14)\\
J_z\dot{\omega}_{Bz} &= n_{Bz} && (6.15)
\end{aligned}
$$

where

$$
\begin{aligned}
f_{Cx} &= f_{Rx} + f_{Lx} && (6.16)\\
f_{Cy} &= f_{Ry} + f_{Ly} && (6.17)\\
f_{Cz} &= f_{Rz} + f_{Lz} && (6.18)\\
n_{Bx} &= -x_{Rz}f_{Ry} + x_{Ry}f_{Rz} + n_{Rx} - x_{Lz}f_{Ly} + x_{Ly}f_{Lz} + n_{Lx} && (6.19)\\
n_{By} &= x_{Rz}f_{Rx} - x_{Rx}f_{Rz} + n_{Ry} + x_{Lz}f_{Lx} - x_{Lx}f_{Lz} + n_{Ly} && (6.20)\\
n_{Bz} &= -x_{Ry}f_{Rx} + x_{Rx}f_{Ry} + n_{Rz} - x_{Ly}f_{Lx} + x_{Lx}f_{Ly} + n_{Lz}. && (6.21)
\end{aligned}
$$

$\boldsymbol{f}_C = [f_{Cx}, f_{Cy}, f_{Cz}]^T$ is the force at the COM of the robot and $\boldsymbol{n}_B = [n_{Bx}, n_{By}, n_{Bz}]^T$ is the torque around the body.

On the other hand we obtain the force and torque at the origin as

$$
\begin{aligned}
f_{0x} &= f_{Cx} && (6.22)\\
f_{0y} &= f_{Cy} && (6.23)\\
f_{0z} &= f_{Cz} && (6.24)\\
n_{0x} &= -p_{Bz}f_{Cy} + p_{By}f_{Cz} + n_{Bx} && (6.25)\\
n_{0y} &= p_{Bz}f_{Cx} - p_{Bx}f_{Cz} + n_{By} && (6.26)\\
n_{0z} &= -p_{By}f_{Cx} + p_{Bx}f_{Cy} + n_{Bz}. && (6.27)
\end{aligned}
$$

Then the *Zero Moment Point* (ZMP) with respect to the origin are defined by

$$
\begin{aligned}
ZMP_x &= -n_{0y}/f_{0z} && (6.28)\\
ZMP_y &= n_{0x}/f_{0z} && (6.29)
\end{aligned}
$$

where ZMP_x and ZMP_y are in sagittal and frontal planes, respectively. From the equations (6.22)–(6.29) we have the equivalent system to (6.10)–(6.15) as follows.

$$
m\ddot{p}_{Cx} = \frac{f_{Cz}}{p_{Cz}}(p_{Cx} - ZMP_x) + \Delta f_{Cx} \qquad (6.30)
$$

$$m\ddot{p}_{Cy} = \frac{f_{Cz}}{p_{Cz}}(p_{Cy} - ZMP_y) + \Delta f_{Cy} \tag{6.31}$$

$$m\ddot{p}_{Cz} + mg = f_{Cz} \tag{6.32}$$

$$J_x\dot{\omega}_{Bx} = p_{Cz}\Delta f_{Cy} + n_{dx} \tag{6.33}$$

$$J_y\dot{\omega}_{By} = -p_{Cz}\Delta f_{Cx} + n_{dy} \tag{6.34}$$

$$J_z\dot{\omega}_{Bz} = n_{Bz} \tag{6.35}$$

where Δf_{Cx} and Δf_{Cy} are new variables defined by

$$\Delta f_{Cx} = -\frac{n_{By}}{p_{Cz}} + \frac{(p_{Cz} - p_{Bz})f_{Cx} - (p_{Cx} - p_{Bx})f_{Cz}}{p_{Cz}} \tag{6.36}$$

$$\Delta f_{Cy} = \frac{n_{Bx}}{p_{Cz}} + \frac{(p_{Cz} - p_{Bz})f_{Cy} - (p_{Cy} - p_{By})f_{Cz}}{p_{Cz}} \tag{6.37}$$

and n_{dx}, n_{dy} are the disturbances defined by

$$n_{dx} = -(p_{Cz} - p_{Bz})f_{Cy} + (p_{Cy} - p_{By})f_{Cz} \tag{6.38}$$

$$n_{dy} = (p_{Cz} - p_{Bz})f_{Cx} - (p_{Cx} - p_{Bx})f_{Cz}. \tag{6.39}$$

Then we can regard ZMP_x, ZMP_x, f_{Cz}, Δf_{Cx}, Δf_{Cy}, and n_{Bz} as the control inputs of the system (6.30)–(6.35). Thus the feedback control law is designed as follows.

$$ZMP_x = \frac{p_{Cz}m}{f_{Cz}}[K_{11}(p_{Cx} - p_{Cx}^{ref}) + K_{12}\dot{p}_{Cx}] - p_{Cx} \tag{6.40}$$

$$ZMP_y = \frac{p_{Cz}m}{f_{Cz}}[K_{21}(p_{Cy} - p_{Cy}^{ref}) + K_{22}\dot{p}_{Cy}] - p_{Cy} \tag{6.41}$$

$$f_{Cz} = -m[K_{31}(p_{Cz} - p_{Cz}^{ref}) + K_{32}\dot{p}_{Cz}] + mg \tag{6.42}$$

$$\Delta f_{Cy} = -\frac{J_x}{p_{Cz}}[K_{41}(\theta_{Bx} - \theta_{Bx}^{ref}) + K_{42}\omega_{Bx}] \tag{6.43}$$

$$\Delta f_{Cx} = \frac{J_y}{p_{Cz}}[K_{51}(\theta_{By} - \theta_{By}^{ref}) + K_{52}\omega_{By}] \tag{6.44}$$

$$n_{Bz} = -J_z[K_{61}(\theta_{Bz} - \theta_{Bz}^{ref}) + K_{62}\omega_{Bz}] \tag{6.45}$$

subject to the limitations

$$ZMP_x^l \leq ZMP_x \leq ZMP_x^u \tag{6.46}$$

$$ZMP_y^l \leq ZMP_y \leq ZMP_y^u. \tag{6.47}$$

Then the equivalent control input on the system (6.10)–(6.15), that is the force input $\tilde{\boldsymbol{u}}_E^*$ at the COM of the robot and around the body instead of (3.50), can be obtained by

$$\tilde{\boldsymbol{u}}_E^* = [f_{Cx}, f_{Cy}, f_{Cz}, n_{Bx}, n_{By}, n_{Bz}]^T \tag{6.48}$$

where

$$f_{Cx} = \frac{f_{Cz}}{p_{Cz}}(p_{Cx} - ZMP_x) \tag{6.49}$$

$$f_{Cy} = \frac{f_{Cz}}{p_{Cz}}(p_{Cy} - ZMP_y) \tag{6.50}$$

$$n_{Bx} = p_{Cz}\Delta f_{Cy} \tag{6.51}$$

$$n_{By} = -p_{Cz}\Delta f_{Cx} \tag{6.52}$$

$$\tag{6.53}$$

The force reference of both feet $\boldsymbol{f}_A$ is generated by the proposed simplified posture controller as follows.

$$\boldsymbol{f}_A = \boldsymbol{W}^2 \tilde{\boldsymbol{K}}^T (\tilde{\boldsymbol{K}} \boldsymbol{W}^2 \tilde{\boldsymbol{K}}^T)^{-1} \tilde{\boldsymbol{u}}_E^* \tag{6.54}$$

Here, $\tilde{\boldsymbol{K}}$ represents a transformation matrix from the force/torque of the both feet to the one of the body defined in (3.48). This solution is the optimal in a sense that $\boldsymbol{f}_A$ has minimum square norm $||\boldsymbol{W}^{-1}\boldsymbol{f}_A||^2$. $\boldsymbol{W}$ is a weighting matrix defined by

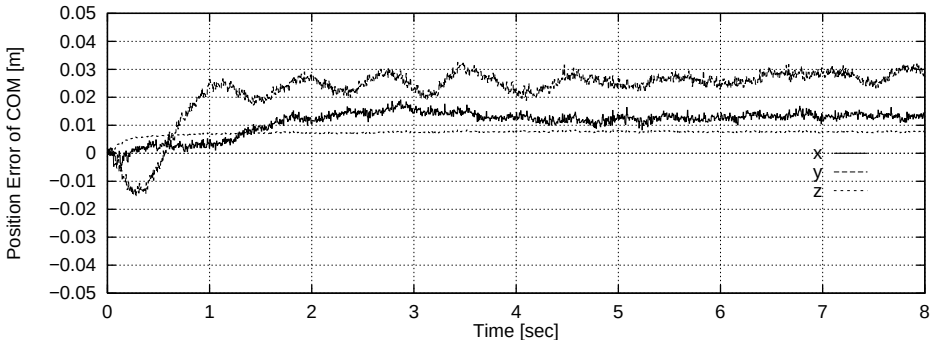
$$\boldsymbol{W} = \mathrm{diag}\{w_1, w_2, \ldots, w_{12}\}. \tag{6.55}$$

The stable posture control of the robot is shown as Fig. 6.7. The position of COM and the attitude of the body is well controlled within ± 0.03 [m] and ± 0.04 [rad] ($= \pm$ 2.3 [deg]) errors. Also ZMPs are directly controlled as shown in Fig. 6.8 through the force controller where the ZMPs control the COM of the robot. Here, the length of foot is 0.18 [m]. The upper bound of the foot is $+0.09$ [m] and the lower bound is -0.09 [m] in sagittal plane Fig. 6.8 (a). In lateral plane Fig. 6.8 (b), the upper bound of the support domain is $+0.155$ [m] and the lower bound is -0.155 [m]. (Note that this case is in the double support.) Both in sagittal plane and in lateral plane the ZMPs are located within the support domain, i. e. this method guarantees the stability in a sense of the ZMP criterion.

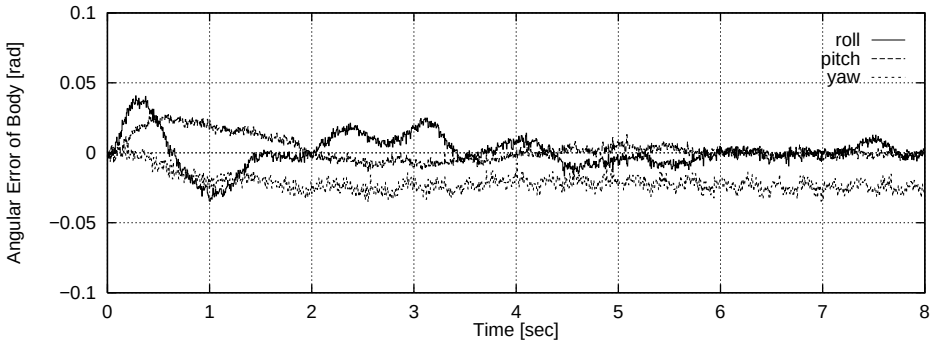
The parameters of the posture controller in (6.40)–(6.47) and (6.55) are shown in Table 6.4.

Table 6.4: Parameters of posture controller.

variables	values	variables	values
m	21.446 [kg]		
J_x	1.0 [kg·m^2]		
J_y	1.5 [kg·m^2]		
J_z	1.5 [kg·m^2]		
K_{11}	1.0	K_{12}	2.0
K_{21}	4.0	K_{22}	4.0
K_{31}	100.0	K_{32}	20.0
K_{41}	100.0	K_{42}	20.0
K_{51}	10.0	K_{52}	20.0
K_{61}	25.0	K_{62}	10.0
ZMP_x^l	-0.071 [m]	ZMP_x^u	0.071 [m]
ZMP_y^l	-0.148 [m]	ZMP_y^u	0.148 [m]
w_1, w_2, w_3, w_7, w_8, w_9	1.0		
w_4, w_5, w_{10}, w_{11}	0.25		
w_6, w_{12}	0.01		

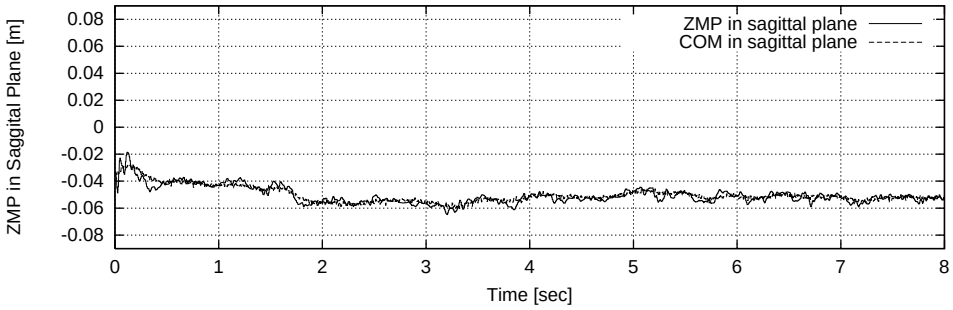


(a) Error of COM position.

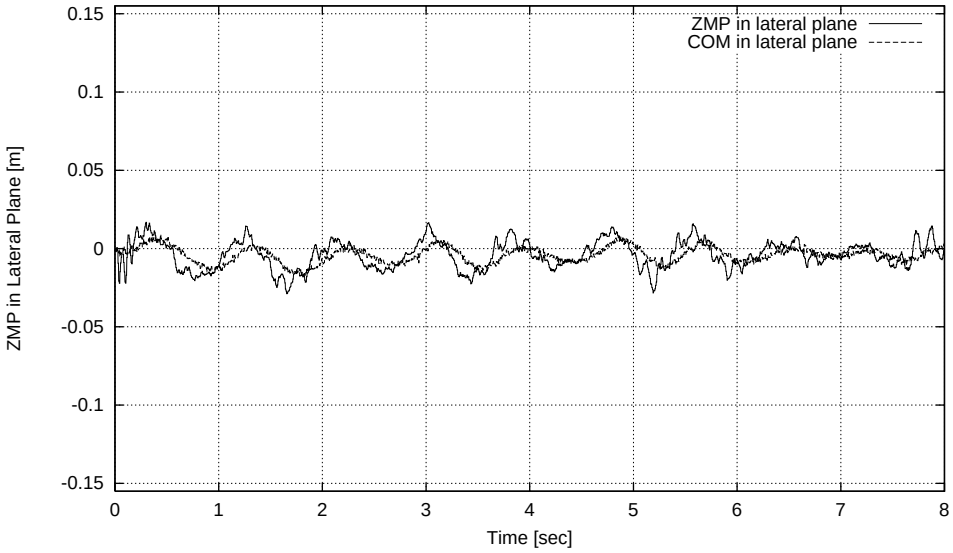


(b) Error of rotation of body.

Figure 6.7: Error of COM position and body rotation.



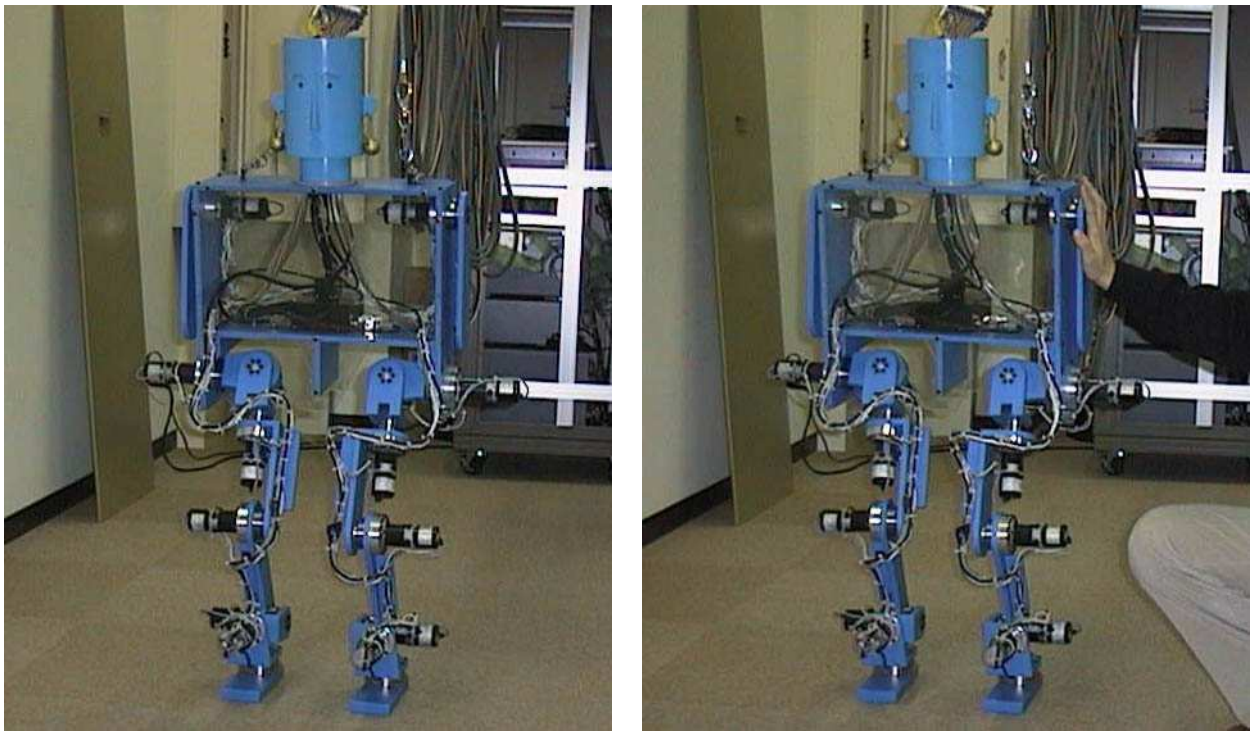
(a) In sagittal plane



(b) In frontal plane

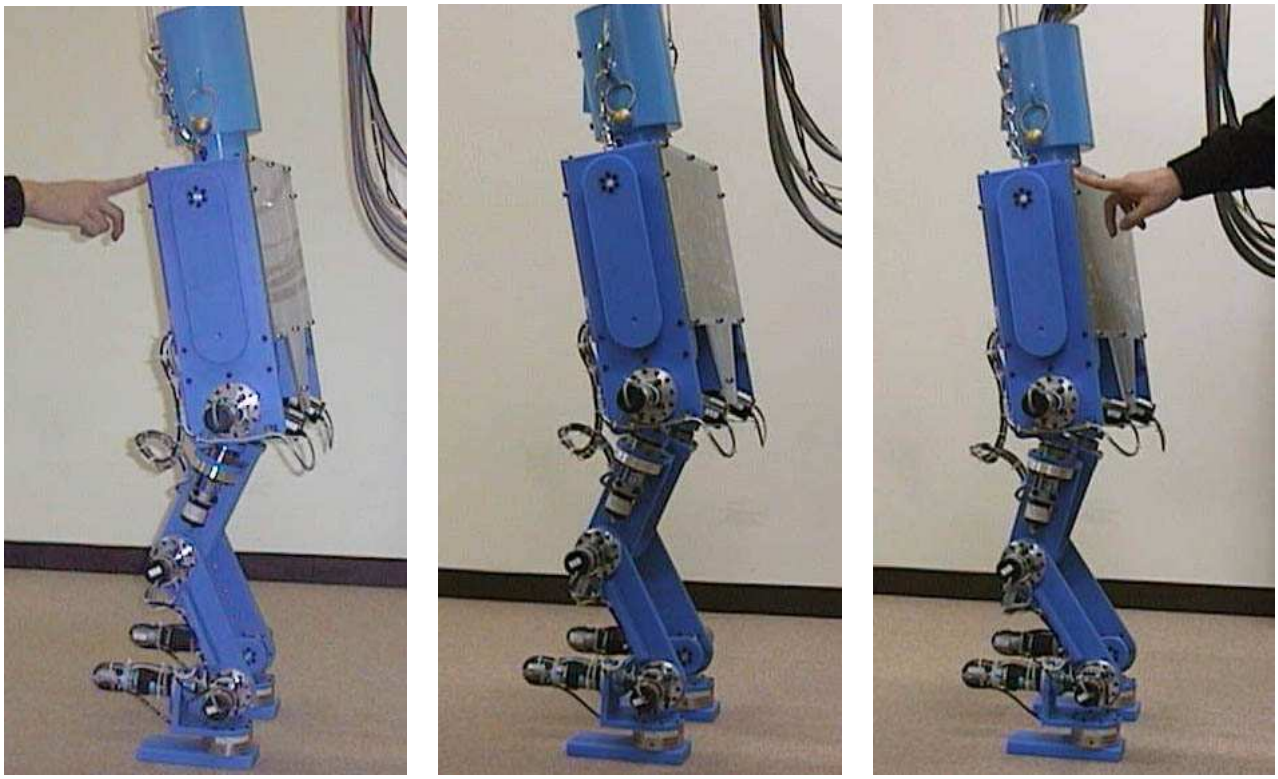
Figure 6.8: Measured ZMP trajectories.

Fig. 6.9 and Fig. 6.10 show snapshots of the experiment of posture control in frontal plane and sagittal plane, respectively. The robot slightly changes its posture when the disturbances are imposed.



(a) Without disturbance. (b) With disturbance.

Figure 6.9: Snapshots of posture control in frontal plane.



(a) With disturbance on the front.

(b) Without disturbance.

(c) With disturbance on the back.

Figure 6.10: Snapshots of posture control in sagittal plane.

Fig. 6.11 and Fig. 6.12 show the trajectories of the center of mass of the robot and the zero moment point in sagittal plane, where the ZMPs are used to control the COMs. The disturbance was imposed on the front of the robot at $t = 5.0$[sec] in Fig. 6.11 and on the back at $t = 4.0$[sec[] in Fig. 6.12. The COMs recovered by the proposed control.

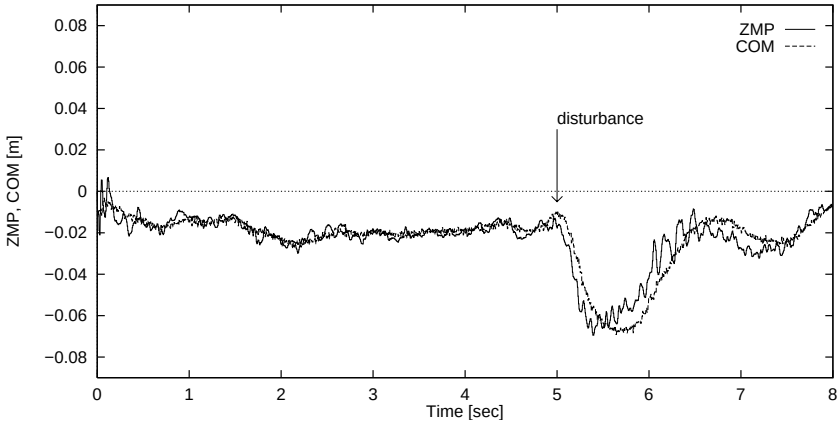


Figure 6.11: Trajectories of ZMP and COM in sagittal plane under disturbance on the front.

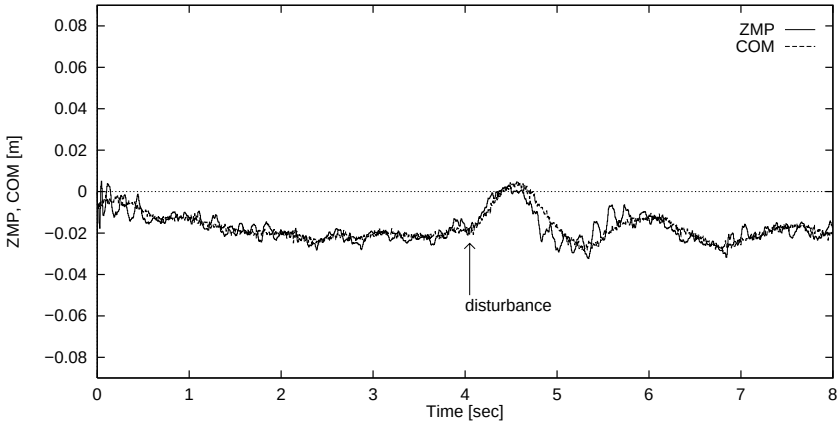


Figure 6.12: Trajectories of ZMP and COM in sagittal plane under disturbance on the back.

Chapter 7

Discussion

7.1 Feedback or Feedforward?

The proposed control is basically based on feedback controls. Animals including human beings, however, seems to do feedforward control such as adapting, learning, etc. My opinion is as follows. The animals also have hierarchical control system and the lower layers will be feedback controls and the upper layers will be feedforward. The motion controls including walking, running, manipulating, etc. must be feedback because there are fast interactions to and from environments. The feedforward approach will do not work well in such cases.

However, to make the system more autonomous and intelligent, the adaptive approach is needed upon the control system proposed in this paper. That is a future work.

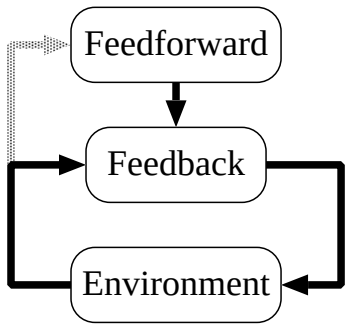


Figure 7.1: Role of feedback and feedforward.

7.2 On the Honda's Humanoid Robot

On Dec. 20, 1996 Honda Co. Ltd. announced officially their humanoid robot. The movement is very smooth and the control seems to be efficient. However, the technical details has not been opened to the public, then we cannot discuss the differences between the control algorithms proposed in this paper and the Honda's algorithms.

The originality of this paper is the direct and robust interaction control on the environment. There is no information that the Honda's robot does such a control or not.

Chapter 8

Conclusion

The summary of the proposals in this paper is as follows.

1. proposal of new exact simulator of legged robots with environmental interaction

2. proposal of new hierarchical control based on robust interaction control on environment

The proposed control system is applied to the 20 axes simulation model, and the stable biped locomotion with a velocity 0.25 m/sec and a stepping time 0.5 sec/step is realized, even if there is unknown slopes (5 [deg]).

In the experiments of the 14 axes biped robot, the attitude control of the body of the biped robot is realized. The position of COM and the attitude of the body is well controlled within ± 0.03 [m] and ± 0.04 [rad] ($= \pm$ 2.3 [deg]) errors.

Since the computing ability of recent micro processor is not enough to implement all of the proposed control algorithm in real-time, the experiments has not been done completely. For the future work, the proposed control will be fully investigated using much fast computers. The introduction of the adaptive methods to motion planners is also the future work.

Bibliography

[1] R. Y. Chiang and M. G. Safonov, *MATLAB Robust Control Toolbox User's Guide*, The Math Works, 1992.

[2] R. W. Cottle and G. B. Dantzig, "Complementary Pivot Theory of Mathematical Programming," *Linear Algebra and Its Applications*, vol. 1, pp. 103–125, 1968.

[3] P. S. Freeman and D. E. Orin, "Efficient Dynamic Simulation of a Quadruped Using a Decoupled Tree-Structure Approach," Int. J. Robotics Research, vol. 10, no. 6, pp. 619–627, 1991.

[4] T. Fukuda, Y. Komata, and T. Arakawa, "Stabilization Control of Biped Locomotion Robot based Learning with GAs having Self-adaptive Mutation and Recurrent Neural Networks," *Proc. IEEE Int. Conf. on Robotics and Automation*, pp. 217–222, 1997.

[5] J. Furusho and A. Sano, "Sensor-Based Control of a Nine-Link Biped," *Int. J. Robotics Research*, vol. 9, no. 2, pp. 83–98, 1990.

[6] P. E. Gill and W. Murray, "Numerically Stable Methods for Quadratic Programming," *Mathematical Programming*, vol. 14, pp. 349–372, 1978.

[7] C. L. Golliday, JR. and H. Hemami, "An Approach to Analyzing Biped Locomotion Dynamics and Designing Robot Locomotion Controls," IEEE Trans. Automatic Control, vol. 22, no. 6, pp. 962–972, 1977.

[8] H. Hashimoto, K. Maruyama, and F. Harashima, "A Microprocessor-Based Robot Manipulator Control with Sliding Mode," *IEEE Trans. Industrial Electronics*, vol. IE-34, no. 1, 1987.

[9] K. Hashimoto, M. Kawabata, and H. Kimura, "H_∞ Controller Design for Robust Manipulator Control," ASME JAPAN/USA Symposium on Flexible Automation, vol. 1, pp. 591–594, 1992.

[10] H. Hemami and R. L. Farnsworth, "Postural and Gait Stability of a Planar Five Link Biped by Simulation," IEEE Trans. Automatic Control, vol. 22, no. 3, pp. 452–458, 1977.

[11] H. Hemami and B. F. Wyman, "Modeling and Control of Constrained Dynamic System with Application to Biped Locomotion in Frontal Plane," IEEE Trans. Automatic Control, vol. 24, no. 4, pp. 526–535, 1979.

[12] J. K. Hodgins, "Three-Dimensional Human Running," *Proc. IEEE Int. Conf. on Robotics and Automation*, pp. 3271–3276, 1996.

[13] S. Kajita, T. Yamaura, and A. Kobayashi, "Dynamic Walking Control of a Biped Robot Along a Potential Energy Conserving Orbit," *IEEE Trans. on Robotics and Automation*, vol. 8, no. 4, pp. 431–438, 1992.

[14] S. Kajita and K. Tani, "Adaptive Gait Control of a Biped Robot based on Realtime Sensing of the Ground Profile," *Proc. IEEE Int. Conf. on Robotics and Automation*, pp. 570–577, 1996.

[15] S. Kawaji, N. Matsunaga and M. Arao, "Hierarchical Control of Biped Locomotion Robot," in *Proc. IEEE Int. Workshop on Advanced Motion Control*, pp. 421-430, 1994.

[16] A. Kawamura, H. Itoh, and K. Sakamoto, "Chattering Reduction of Disturbance Observer Based Sliding Mode Control," *IEEE Trans. Industry Applications*, vol. 30, pp. 456–461, 1994.

[17] S. Komada and K. Ohnishi, "Force Feedback Control of Robot Manipulator by Acceleration Tracing Orientation Method," *IEEE Trans. Industrial Electronics*, vol. IE-37, no. 1, pp. 6–12, 1989.

[18] A. Kun and W. T. Miller, III, "Adaptive Dynamic Balance of a Biped Robot using Neural Networks," *Proc. IEEE Int. Conf. on Robotics and Automation*, pp. 240–245, 1996.

[19] P. Lötstedt, "Numerical Simulation of Time-Dependent Contact and Friction Problems in Rigid Body Mechanics," *SIAM J. Scientific and Statistical Computing*, vol. 5, no. 2, pp. 370–393, 1984.

[20] J. Y. S. Luh, M. W. Walker, and R. P. C. Paul, "On-line Computational Scheme for Mechanical Manipulators," *ASME J. Dynamic Systems, Measurement, Control*, vol. 102, pp. 69–76, 1980.

[21] J. Y. S. Luh, M. W. Walker, and R. P. C. Paul, "Resolved-Acceleration Control of Mechanical Manipulators," IEEE Trans. Automatic Control, vol. 25, no. 3, pp. 468–474, 1980.

[22] H. Minakata and Y. Hori, "Development of Biped Bike Prototype 'Ostrich-I&II'," *Proc. Asian Control Conference*, vol. 3, pp. 319–322, 1997.

[23] T. Mita, M. Hirata, and S. B. Villas-Boas, "Design of H_∞ Controllers for Plants Having Poles on the $j\omega$ Axis, — H_∞ Motion Control —," *Trans. of IEE of Japan*, vol. 115-D, no. 10, pp. 1253–1262, 1995.

[24] T. Murakami, F. Yu, and K. Ohnishi, "Torque Sensorless Control in Multidegree-of-Freedom Manipulator," *IEEE Trans. Industrial Electronics*, vol. IE-40, pp. 259–265, 1993.

[25] M. Nakao, K. Ohnishi, and K. Miyachi, "A Robust Decentralized Joint Control Based on Interference Estimation", *Proc. IEEE Int. Conf. on Robotics and Automation*, pp. 326–331, 1987.

[26] B. Nobel, *Applied Linear Algebra*, Prentice-Hall, Englewood Cliffs, N.J., 1969.

[27] K. Ohishi, M. Nakao, K. Ohnishi, and K. Miyachi, "Microprocessor-Controlled DC Motor for Load-Insensitive Position Servo System," *IEEE Trans Industrial Electronics*, vol. IE-34, pp. 44–49, 1987.

[28] K. Ohnishi and T. Murakami, "Advanced Motion Control in Robotics," *Proc. IEEE Int. Conf. on Industrial Electronics, Control, and Instrumentation*, pp. 356–359, 1989.

[29] J. Pratt, P. Dilworth, and G. Pratt, "Virtual Model Control of a Bipedal Walking Robot," *Proc. IEEE Int. Conf. on Robotics and Automation*, pp. 193–198, 1997.

[30] M. H. Raibert, *Legged Robots That Balance*, Cambridge, MA, MIT Press, 1986.

[31] A. W. Salatian, K. Y. Yi, and Y. F. Zheng, "Reinforcement Learning for a Biped Robot to Climb Sloping Surfaces," *J. Robotic Systems*, vol. 14, no. 4, pp. 283–296, 1997.

[32] L. Shih, A. A. Frank and B. Ravani, "Dynamic Simulation of Legged Machines Using a Compliant Joint Model," Int. J. Robotics Research, vol. 6, no. 4, pp. 33–46, 1987.

[33] J. H. Shin, K. Fujiune, T. Suzuki, S. Okuma, and K. Yamada, "Positioning Control of Direct Drive Robot with Two-Degree-of-Freedom Compensator," *Trans. of IEE of Japan*, vol. 115-D, no. 4, pp. 444–451, 1995.

[34] J. J. E. Slotine and W. Li, "On the Adaptive Control of Robot Manipulators", *Int. J. Robotics Research*, vol. 6, no. 3, 1987.

[35] J. J. Slotine and W. Li, *Applied Nonlinear Control*, Prentice Hall, Englewood Cliffs, New Jersey, 1991.

[36] K. Sorao, T. Murakami, and K. Ohnishi, "A Unified Approach to ZMP and Gravity Center Control in Biped Dynamic Stable Walking," *Proc. IEEE/ASME Int. Conf. on Advanced Intelligent Mechatronics*, CD-ROM, 1997.

[37] T. Umeno, T. Kaneko, and Y. Hori, "Robust Servosystem Design with Two Degrees of Freedom and its Application to Novel Motion Control of Robot Manipulators", *IEEE Trans. on Industrial Electronics*, vol. IE-40, pp. 473–485, 1993.

[38] M. W. Walker and D. E. Orin, "Efficient Dynamic Computer Simulation of Robotic Mechanisms," *ASME J. Dynamic Systems, Measurement, Control*, vol. 104, pp. 205–211, 1982.

[39] D. E. Whitney, "Resolved Motion Rate Control of Mechanical Manipulators and Human Prosthesis," IEEE Trans. Man-Machine Systems, vol. 10, no. 2, pp. 47–53, 1969.

[40] S. Wolfram, *Mathematica™ — A System for Doing Mathematics by Computer,* Wolfram Research, Inc., Illinois, Addison-Wesley Publishing, 1992.

[41] J. Yamaguchi, A. Takanishi and I. Kato, "Development of a Biped Walking Robot Compensating for Three-Axis Moment by Trunk Motion," *J. Robotics Society of Japan*, vol. 11, no. 4, pp.581–586, 1993. (in Japanese)

[42] J. Yamaguchi, A. Takanishi, and I. Kato, "Experimental Development of a Foot Mechanism with Shock Absorbing Material for Acquisition of Landing Surface Position Information and Stabilization of Dynamic Biped Walking," *IEEE Int. Conf. Robotics and Automation*, pp. 2892–2899, 1995.

[43] J. Yamaguchi, N. Kinoshita, A. Takanishi, and I. Kato, "Development of a Dynamic Biped Walking System for Humanoid, — Development of a Biped Walking Robot Adapting to the Human's Living Floor —," *Proc. IEEE Int. Conf. on Robotics and Automation*, pp. 232–239, 1996.

List of Figures

1.1 An illustration of a legged system. · · · 2
1.2 A block diagram of the legged system. · · · 2

2.1 Representation of link-fixed coordinates. · · · 6

3.1 Biped walking control system. · · · 7
3.2 Plant system in joint space. · · · 9
3.3 Virtual plant. · · · 10
3.4 Equivalent system of virtual plant. · · · 10
3.5 Control system. · · · 11
3.6 Augmented system. · · · 12
3.7 Frequency characteristics of cost functions, sensitivity function, and complementary sensitivity function · · · 13
3.8 Global control system. · · · 14
3.9 Kinematics of legged robot. · · · 15
3.10 Force control system. · · · 16
3.11 An illustration of the kinematic property. · · · 18
3.12 Model of arms. · · · 22

4.1 Rotation of coordinates. · · · 26
4.2 Indices of links. · · · 27
4.3 Coordinates and parameters of links. · · · 28
4.4 Contact points. · · · 30

5.1 Snapshots of biped walking simulation. (More details are found in the web, http://www.kawalab.dnj.ynu.ac.jp/) · · · 38
5.1 Snapshots of biped walking simulation. (continued.) · · · 39
5.2 Trajectory of zero moment point. · · · 39
5.3 Trajectory of center of mass of the robot. · · · 40
5.3 Trajectory of center of mass of the robot (magnified.) · · · 41
5.4 Trajectory of zero moment point. · · · 42
5.5 Trajectory of center of mass of the robot. · · · 43
5.5 Trajectory of center of mass of the robot (magnified.) · · · 44
5.6 Required yaw moment and realized one. (in a case of no compensation) · · · 45
5.7 Rotation around yaw axis. · · · 46
5.8 Snapshots of biped walking simulation with arm swing motion. (More details are found in the web, http://www.kawalab.dnj.ynu.ac.jp/) · · · 47

5.8 Snapshots of biped walking simulation with arm swing motion. (continued.) · · 48

6.1 A photo of 14-axis biped robot. · 50
6.2 Dimension of 14-axis biped robot. · 51
6.3 System configuration. · 52
6.4 Frequency characteristics of cost functions, sensitivity function, and complementary sensitivity function in experiments · 54
6.5 H_∞ joint servo control. · 56
6.6 Force control on foothold. · 58
6.7 Error of COM position and body rotation. · 62
6.8 Measured ZMP trajectories. · 63
6.9 Snapshots of posture control in frontal plane. · 64
6.10 Snapshots of posture control in sagittal plane. · · · · · · · · · · · · · · · · · · · 65
6.11 Trajectories of ZMP and COM in sagittal plane under disturbance on the front. 66
6.12 Trajectories of ZMP and COM in sagittal plane under disturbance on the back. 66

7.1 Role of feedback and feedforward. · 67

List of Tables

4.1 Examples of a inner link and outer links in Fig. 4.2 case. 27

5.1 Parameters of biped robot. 37

6.1 Estimated rated power and torque of joints from simulations. 53
6.2 Selected motors and gears. 53
6.3 Parameters of force controller. 57
6.4 Parameters of posture controller. 61

APPENDIX

Appendix A

Related Publications

Papers

1. Y. Fujimoto and A. Kawamura, "Considerations on Disturbance Suppressions of Sliding Mode Control with Disturbance Observer and Two-Degree-of-Freedom Control," *Trans. of IEE of Japan*, vol. 114-D, no. 3, pp. 306–314, 1994. (in Japanese)

2. Y. Fujimoto and A. Kawamura, "Robust Servo-system Based on Two-Degree-of-Freedom Control with Sliding Mode," *IEEE Trans. on Industrial Electronics*, vol 42, no 3, pp. 272–280, June 1995.

3. Y. Fujimoto and A. Kawamura, "Three Dimensional Digital Simulation of Legged Robots," *Journal of Robotics and Mechatronics, (Fuji Technology Press Ltd.)*, vol. 8, no. 3, pp. 266–271, 1996.

4. Y. Fujimoto and A. Kawamura, "An Inertia Fluctuation Insensitive Robust Control of Robot Manipulators Based on a Combination of Inertia Torque Computation Filter and H_∞ Control," *Trans. of IEE of Japan*, vol. 117-D, no. 4, pp. 493–500, 1997. (in Japanese)

5. Y. Fujimoto and A. Kawamura, "A Three Dimensional Dynamic Simulation of Biped Walking Robot Considering Collision and Friction between Foot and Ground," *Journal of the Robotics Society of Japan*, vol. 15, no. 6, pp. 857–863, 1997. (in Japanese)

6. Y. Fujimoto and A. Kawamura, "Autonomous Control and 3D Dynamic Simulation of Biped Walking Robot Including Environmental Force Interaction," *IEEE Robotics and Automation Magazine*, vol. 5, no. 2, 1998. (to appear)

International Conference

1. Y. Fujimoto and A. Kawamura, "Robust Servo-system Based on Two-Degree-of-Freedom Control with Sliding Mode," *Proc. IEEE Int. Conf. on Industrial Electronics, Control, and Instrumentation (IECON'93)*, pp. 85–90, Maui, November 1993.

2. Y. Fujimoto and A. Kawamura, "Comparison between Sliding Mode Control with Disturbance Observer and Two-degree-of-freedom Control," *Proc. Japan Industry Applications Society Conference (JIASC'93)*, pp. 145–150, Tokyo, August 1993.

3. Y. Fujimoto and A. Kawamura, "Three Dimensional Digital Simulation and Autonomous Walking Control for Eight-axis Biped Robot," *Proc. IEEE Int. Conf. on Robotics and Automation (ICRA'95)*, pp. 2877–2884, Nagoya, May 1995.

4. Y. Fujimoto and A. Kawamura, "Proposal of Biped Walking Control Based on Robust Hybrid Position/Force Control," *Proc. IEEE Int. Conf. on Robotics and Automation (ICRA'96)*, pp. 2724–2730, Minneapolis, April 1996.

5. Y. Fujimoto and A. Kawamura, "Robust Biped Walking with Active Interaction Control between Robot and Environment," *Proc. IEEE Int. Workshop on Advanced Motion Control (AMC'96)*, pp. 247–252, Mie, March 1996.

6. Y. Fujimoto and A. Kawamura, "Biped Walking Control with Optimal Foot Force Distribution by Quadratic Programming," *Proc. IEEE/ASME Int. Conf. on Advanced Intelligent Mechatronics (AIM'97)*, CD–ROM, Tokyo, June 1997.

7. Y. Fujimoto and A. Kawamura, "Robust Control of Biped Walking Robot with Yaw Moment Compensation by Arm Motion," *Proc. Asian Control Conference (ASCC'97)*, vol. 3, pp. 327–330, Seoul, July 1997.

8. Y. Fujimoto and A. Kawamura, "Robust Biped Walking with Force Interaction Control between Foot and Ground," *Proc. IEEE Int. Conf. on Robotics and Automation (ICRA'98)*, Leuven, Belgium, May 1998. (to appear)

9. Y. Fujimoto and A. Kawamura, "Attitude Control Experiments of Biped Walking Robot based on Environmental Force Interaction," *Proc. IEEE Int. Workshop on Advanced Motion Control (AMC'98)*, Coimbra, Portugal, June 1998. (to appear)

Domestic Conference

1. 藤本, 坂本, 河村, "外乱オブザーバを併用したスライディングモード制御と2自由度制御の比較に関する一考察," 電気学会産業計測制御研究会, IIC-92-26, pp. 77–86, 1992.

2. 藤本, 河村, "DSP制御ロバストサーボ系に対する一考察," 電気学会電子回路研究会, ECT-93-41, pp. 17–22, 1993.

3. 藤本, 河村, "DSP制御ロバストサーボのディジタル再設計に関する一考察," 電気学会産業計測制御研究会, IIC-94-3, pp. 21–26, 1994.

4. 藤本, 河村, "2足歩行ロボットの精密なシミュレーションモデルの確立," 第*12*回日本ロボット学会学術講演会, pp. 1183–1184, 1994.

5. 藤本, 河村, "2足歩行ロボットの精密な3次元シミュレーションと自律的歩行パターン生成法の提案," 電気学会産業計測制御研究会, IIC-95-13, pp. 109–116, 1995.

6. 藤本, 河村, “計算トルク法と H ∞制御を組み合わせた慣性変動に不感なロボットマニピュレータのロバスト制御,” 平成 *7*年電気学会産業応用部門全国大会, vol. 3, pp. 277–280, 1995.

7. 藤本, 河村, “位置と力のハイブリッド制御に基づく 2 足歩行制御法の提案,” 第 *13*回日本ロボット学会学術講演会, pp. 199–200, 1995.

8. 藤本, 河村, “2 足ロボットの床反力を考慮した安定化制御と自律的歩行パターン生成システムの提案,” 電気学会産業計測制御研究会, IIC-96-20, pp. 103–110, 1996.

9. 藤本, 河村, “床反力を考慮した 2 足歩行ロボットの安定化制御,” 平成 *8*年電気学会全国大会, no. 3, pp. 104–105, 1996.

10. 藤本, 河村, “床反力制御による 2 足歩行ロボット,” 平成 *8*年電気学会産業応用部門全国大会, vol. 3, pp. 299–302, 1996.

11. 藤本, 河村, “環境との相互作用力を考慮した 2 足歩行制御,” 第 *14* 回日本ロボット学会学術講演会, pp. 197–198, 1996.

12. 藤本, 河村, “剛体モデルを用いた 2 足歩行ロボット制御,” 電気学会産業計測制御研究会, IIC-97, pp. 1–7, 1997.

13. 藤本, 河村, “腕の運動によりヨー軸モーメントを実時間補償する 2 足歩行制御法,” 第 *15*回日本ロボット学会学術講演会, pp. 97–98, 1997.

14. 藤本, 河村, “環境との相互作用に基づく 2 足歩行ロボットの制御： 3 次元姿勢制御実験,” 平成 *10*年電気学会全国大会.