SSI-DT79115601

Doctoral Thesis

Cooperative Object Transportation With Multiple Humanoid Robots

(複数ヒューマノイドロボットによる協調物体搬送)

Meng-Hung Wu

November, 2015

Division of Systems Science and Informatics
Graduate School of Information Science and Technology
Hokkaido University

Doctoral Thesis
submitted to Graduate School of Information Science and Technology,
Hokkaido University
in partial fulfillment of the requirements for the degree of
Doctor of Philosophy.

Meng-Hung Wu

Thesis Committee: 近野 敦 教授
小笠原 悟司 教授
山下 裕 教授

Cooperative Object Transportation With Multiple Humanoid Robots*

Meng-Hung Wu

Abstract

There are more and more robots appearing in factory or our daily life due to development of recent technology. Among all types of robots, humanoid robots have the potential to perform multiple tasks and walk on uneven terrain like human beings. Hence, it is expected that humanoid robots work instead of human beings at dangerous zones such as plant facilities. In such dangerous zones, humanoid robots must cooperate with each other in order to carry heavy and large objects.

Although there is a large amount of ongoing research about multiple robot cooperation, most of them focus on multiple manipulators or wheeled robots, and there is almost no research about cooperative object transportation by multiple humanoid robots. A few attempts have been made to achieve cooperation between a humanoid robot and human. However, such cooperation will not be possible in disaster zones or dangerous areas.

In this thesis, two frameworks of cooperative object transportation by multiple humanoid robots are proposed and evaluated. Multiple robot cooperation can be classified into two types in general: leader-follower type and symmetry type. Both types of cooperation control are implemented in real humanoid robots and experimentally evaluated in this thesis.

Two humanoid robots HRP-2 were used in experiments. In Chapter 2, the details of software and hardware system used in this research are described.

When a humanoid walks around a real environment, walking pattern must be generated online. In Chapter 3, an online walking pattern generation method used in this research is described.

In Chapter 4, a leader-follower type cooperation control is proposed. In leader-follower type control, the leader robot controls the position and attitude of the object, while the follower robot follows the leader robot based on the force sensor information measured on its hands. In this type of cooperation, the follower robots start planning after the leader robot moves, and hence this time-lag may cause excessive internal forces. The generated excessive internal force acts as disturbance for stabilizing controller of

*Doctoral Thesis, Division of Systems Science and Informatics, Graduate School of Information Science and Technology, Hokkaido University, SSI-DT79115601, November 19, 2015.

robots. Hence such excessive internal force may result in low responsiveness and unexpected tilting of the humanoid robots. There are two ways to suppress the internal force: by changing the relative position of robots or by changing the position of their hands. By combining position-based impedance control with online walking pattern generation, it is possible to generate walking command according to the displacement of the hands. This method is implemented in the follower robot. The proposed leader-follower type control is verified by experiments with two humanoid robots HRP-2 along a sagittal axis.

In Chapter5, a symmetry type cooperation control for arbitrary numbers of humanoid robots is proposed. In symmetry type cooperation control, all robots synchronously move, which achieves high responsiveness and safety when transporting an object. A method to define the external and internal force of an arbitrary number of humanoid robots is proposed. Furthermore, a symmetric hybrid position/force control law is derived. In order to achieve stable object transportation, external position and internal force are controlled. Internal forces are eased by adjusting the positions of all hands. The proposed symmetry type control is validated both in steady state (without walking) and in object transportation (with walking). In steady state, the reference object position is set as initial position, and reference internal forces are given online. The robots can track the reference internal forces while keeping in initial position. In cooperative object transportation experiments, the humanoid robots are able to generate its whole body motion to transport an object and to control the internal force. The cooperative object transportation among four HRP-2 humanoid robots is simulated and the cooperation between two HRP-2 humanoid robots is experimented. Both results show the effectiveness of the proposed symmetry cooperation control.

This work is summarized in Chapter 6.

Keywords: humanoid robots, force control, symmetric control, cooperation moving

Contents

List of Figures

List of Tables

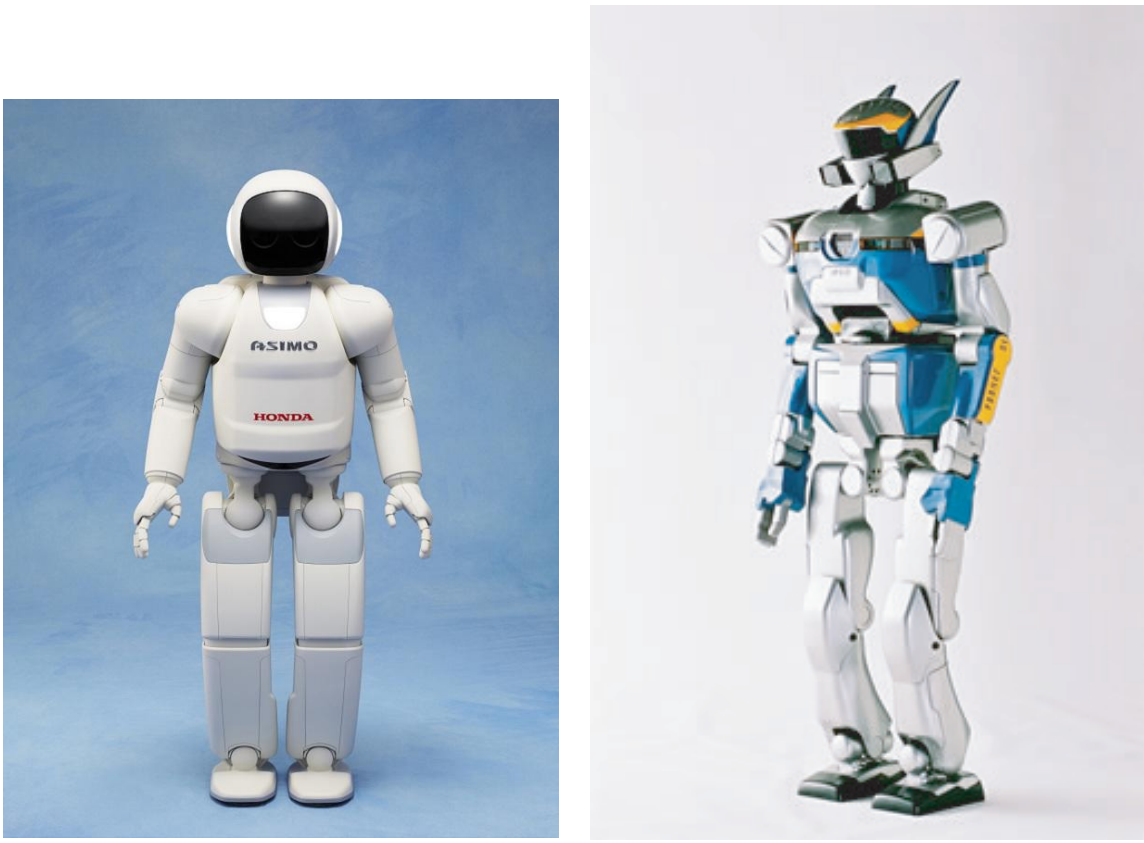
Chapter 1. Introduction

1.1 Real-World Robots

Robot used to be fiction in the past age, but there are more and more robots appear nowadays. In order to free people from tedious work and dangerous environment, and since robot can move more efficient and precise than humans, industry robots take over human position gradually. The first industry robot was Unimate, which is developed by a American company "unimation". Unimate is able to play back a motion taught by a operator. Unimate was imported to Japan in 1967 and Kawasaki heavy industry started to build domestic robots. In recent years, there is a new trend of robots. Robots not only play an active role in industry field, but also appear in service, entertainment and welfare field. This kind of robots features high mobility and intelligent. Among all of this robots, humanoid robots have the shape of human beings, and have the potential to handle multiple tasks and walk on uneven terrain like human beings. Therefore, general versatility of humanoid robots is expected.

1.2 Research of Humanoid Robots

The first humanoid robot is WABOT-1 [1] developed by Kato Lab. in Waseda University. WABOT-1 has the a full element set of humanoid robot, and is able to perform static walking, object handling, recognize object by stereoscopic camera, and communicate with human through acoustic sense and artificial vocal cord. Kato Lab. then developed WL-12, which robot use a weight dumbbell as upper body, and utilizes the inertia of upper body to get balance while walking aggressively. The importance of upper body is verified from this research, and robot with upper body becomes majors after that. Kato Lab. develops WABIAN series from 1997 [2]. WABIAN features a additional roll joint in hip joint to avoid singular point problem. By adding this joint, WABIAN is able to walk with knee unbending. Honda start their humanoid robot research in 1986. They develops P1, P2 and p3 during 1993∼1997 [3]. Furthermore, the advanced model of asimo is revealed in 2001 [4](Figure 1.1(a)), with running ability in 3 [km/h]. Japan government also starts to plan Humanoid Robotics Project in 1996. In 1998, the project starts with Japan's Ministry of Economy, Trade and Industry (METI) and New Energy and Industrial Technology Development Organization (NEDO) as sponsors, Kawada Industries, National Institute of Advanced Industrial Science and Technology (AIST) and Kawasaki Heavy Industries, Inc. as spearheads [6]. The project is divided



(a) ASIMO [5] (b) HRP-2 [8]

Figure 1.1: HRP-2 and Asimo

into prior term and later term. In prior term, virtual platform and remote operating cockpit is developed. And the application of humanoid is studied with robot hardware. HRP-2 [7](Figure 1.1(b)) was revealed as final accomplishment of the project in 2003. HRP-2 is recently rented to some university or laboratory for researching use. After that, HRP-3 [9] with spec of water and dust proof, and female humanoid HRP-4c [10] are revealed in 2007. JSK Lab. in Tokyo University develops H5, H6, H7 humanoid in 1998, 2000 and 2001, respectively [11]. JSK recently develops musculoskeletal humanoid robots [12]. Uchiyama Lab. in Tohoku University develops Saika series [13, 14]. Additionally, SDR series developed by sony [15], HOAP series developed by Fujitsu, partner robot developed by Toyota, there are lots of humanoid robots has been developed so long.

1.3 Robot Capability Enhanced by Cooperation Work

As the power-to-weight ratio of a robot's servo motor is severely lower than human being's muscle, a human-size robotic manipulator is unable to perform dexterously and

effectively as human beings, hence it is difficult to carry a long or heavy object by one single robotic manipulator. It is difficult for a single robotic manipulator to carry a long or heavy object. Even human beings do not move a long or heavy object by only one hand. assemble parts or twist a container's cap The importance of cooperation of multiple robot arms has been reported so far. For cooperation of multiple immobile manipulators, Nakano et al. proposed a master-slave control scheme for dual-arm manipulators [16]. Munawar and Uchiyama proposed a distributed event-based control strategy for a non-autonomous under-actuated multiple manipulator system [17]. Babazadeh and Sadati proposed a method to cooperate two robot manipulators with optimal torques minimizing a relative cost function [18]. Williams and Khatib proposed a concept of virtual linkage that can control internal forces among multiple robotic arms [19]. Position/force hybrid control [20, 21, 22, 23], and impedance-based control schemes for cooperating manipulators [24, 25, 26] have also been objects of study for a long time.

For cooperative object transportation by multiple mobile robots, a leader-follower type control scheme was proposed [27, 28]. Some researches focused on distributing motion commands based on the desired movement of a holding rigid body, and their control scheme facilitated a desired compliant interaction by impedance control [29, 30].

Although there are lots of ongoing research about cooperation of multiple robots, most of them focus on multiple manipulators or wheeled robots, and there is almost no research about cooperation object transportation by multiple humanoid robots. A few attempts have been made on cooperation between a humanoid robot and a human [31, 32]. However, such cooperation will not be possible in disaster zones or in dangerous area. Humanoid robots have similar shapes to human beings, and hence they have the potential to perform various tasks and walk on uneven terrain as human beings do. Therefore, they are expected to execute 3D (Dull, Dirty, or Dangerous) tasks instead of human. Especially, as the DARPA robotics challenge [33] aims, humanoid robots are expected to perform tasks in disaster zones such as removing the debris. Most of debris in a disaster zone may be too large to remove for a single humanoid robot. For such cases, cooperation of multiple humanoid robots will be effective.

Cooperation of multiple robots can be classified into two types in general.

1. Leader-follower type: there is one autonomous robot in the system, which is called leader robot. The leader robot autonomously generates its motion, or is operated directly by a human operator. Then the other robots, which are called follower robots, just follow the leader robot.The controllers of the robots in this system are independent.

2. Symmetry type: there is no apparent leader robot, and a central controller controls all the robot simultaneously. All information of the controlled robots is required in this type.

Both this two types of cooperation control is adapted to multiple humanoid robots and

compared to each other in this paper.

1.4 Purpose of This Research

In order to implement cooperation object transport by multiple humanoid robots, there are two main targets in this research.

1. Position control of the holding object.
2. Internal force control between the humanoid robots.

In this research, a method to control the position of holding object is developed. Operator just give the humanoid robots the reference velocity of the holding object, then the robots will generate their whole body motion autonomously. Besides, there is a moving error between the robots, and when there is a moving error among robots, internal force arises. This unexpected internal force may result in falling down. The internal force should be eased in some way. In order to control the internal force, the internal force is defined in this thesis. Then a hybrid position/internal force control law is proposed, and verified with both dynamic simulator and robot existed humanoid robots.

1.5 Composition of This thesis

The composition of this paper is structured as follows.

chapter 1 Introduction
The background and purpose of this research is describe.

chapter 2 System
How the joint angle is controlled is explained from the motion equation of humanoid robot. Also the robot hardware and software architecture used in this research is introduced.

chapter 3 Online Walking Generator
The online walking pattern generate method is described, and verified with dynamic simulator.

chapter 4 Leader-follower Type Cooperation Control
Leader-follower type cooperation control is adapted to two humanoid robots, and verified with existed humanoid robot HRP-2.

chapter 5 Symmetry Type Cooperation Control
The method of symmetry type cooperation control adapting to arbitrary numbers of humanoid robots is explained. Proposed method is validated with 4 HRP-2 robots in simulator, also validated with 2 HRP-2 robots experiment. Then make a appraisal to purposed method by simulation and experiment data.

chapter 6 Conclusion A short conclusion of this thesis, and prospects of this research is given.

Chapter 2. System

2.1 Introduction

First, a motion equation of a humanoid is described in this chapter. A humanoid robots is consists of multiple rigid links, and the dynamics of a humanoid robots is represented by the motion equation. Next, the method(torque control mode) is explained. And at last of this chapter, the overview of the dynamic simulation OpenHRP3 [34] and the robot HRP-2, which is used in this research, are introduced.

2.2 The Motion Equation of Humanoid Robot

A humanoid robot consists of multiple links, and is not fixed to environment so it can move freely. It is proper to treat the humanoid robots as a manipulator floating in the air. In order to describe the state equation, further 6 degree of freedom(DOF) is added. The 6 DOF represent the position and attitude of base link, to N DOF of joints. This 6 DOF also represents the position and attitude of robot coordinate, which is fixed on the robot, and is called waist coordinate in this paper.

When a N-DOF humanoid robot's position and attitude is represented as the position and attitude of the waist coordinate, the state variables $\mathbf{x}$, velocity vector $\mathbf{v}$, force and torque vector $\mathbf{u}$ are as follows:

$$\boldsymbol{x}^T = \begin{bmatrix} \boldsymbol{p}_{\mathrm{w}}^T & \Psi_{\mathrm{w}}^T & \boldsymbol{\theta}^T \end{bmatrix} \tag{2.1}$$

$$\boldsymbol{v}^T = \begin{bmatrix} \boldsymbol{v}_{\mathrm{w}}^T & \Omega_{\mathrm{w}}^T & \dot{\boldsymbol{\theta}}^T \end{bmatrix} \tag{2.2}$$

$$\boldsymbol{u}^T = \begin{bmatrix} 0 & 0 & \boldsymbol{\tau}^T \end{bmatrix} \tag{2.3}$$

The definition of the variables are:

$\boldsymbol{p}_{\mathrm{w}}$	:	position of waist coordinate(3×1 vector)
Ψ_{w}	:	Euler angles of waist coordinate(3×1 vector)
$\boldsymbol{\theta}$	:	joint angles(N× 1 vector)
$\boldsymbol{v}_{\mathrm{w}}$	:	velocity of waist coordinate(3×1 vector)
Ω_{w}	:	angular velocity of waist coordinate(3×1 vector)
$\dot{\boldsymbol{\theta}}$	:	joint velocity(N×1 vector)
$\boldsymbol{\tau}$	:	joint torque(N×1 vector)

The motion equation of a humanoid is represented as:

$$\mathbf{H}(\boldsymbol{x})\dot{\boldsymbol{v}} + C(\boldsymbol{x}, \boldsymbol{v}) + g(\boldsymbol{x}) = \boldsymbol{u} + \boldsymbol{u}_{\mathrm{E}} \tag{2.4}$$

in this equation,

$\mathbf{H}(\boldsymbol{x})$	:	inertia matrix($(N+6) \times (N+6)$ matrix)
$C(\boldsymbol{x}, \boldsymbol{v})$	:	centrifugal force and Coriolis force $\in \mathbb{R}^{(\mathrm{N}+6)\times(\mathrm{N}+6)}$
$g(\boldsymbol{x})$	:	gravity $\in \mathbb{R}^{(\mathrm{N}+6)\times 1}$
$\boldsymbol{u}_{\mathrm{E}}$	:	external force $\in \mathbb{R}^{(\mathrm{N}+6)\times 1}$

Torque Control Mode

When a humanoid robot's motion is simulated in the simulator, $\boldsymbol{v}$ is solved from (2.4), and then by integrated $\boldsymbol{v}$, $\boldsymbol{x}$ is obtained. $\mathbf{H}(\boldsymbol{x})$, $C(\boldsymbol{x}, \boldsymbol{v})$, $g(\boldsymbol{x})$ is calculated from $\boldsymbol{x}$, $\boldsymbol{v}$. $\boldsymbol{u}_{\mathrm{E}}$ is depend on the contact condition between the foot sole and the ground. $\boldsymbol{u}$ is independent from $\boldsymbol{x}$ and $\boldsymbol{v}$. (2.4) is rewritten as follows:

$$\begin{bmatrix} \mathbf{H}_{11} & \mathbf{H}_{12} & \mathbf{H}_{13} \\ \mathbf{H}_{11} & \mathbf{H}_{12} & \mathbf{H}_{13} \\ \mathbf{H}_{11} & \mathbf{H}_{12} & \mathbf{H}_{13} \end{bmatrix} + \begin{bmatrix} \dot{\boldsymbol{v}}_{\mathrm{w}} \\ \dot{\boldsymbol{\Omega}}_{\mathrm{w}} \\ \ddot{\boldsymbol{\theta}} \end{bmatrix} + \begin{bmatrix} \boldsymbol{b}_1 \\ \boldsymbol{b}_2 \\ \boldsymbol{b}_3 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ \boldsymbol{\tau} \end{bmatrix} + \begin{bmatrix} \boldsymbol{u}_{E1} \\ \boldsymbol{u}_{E2} \\ \boldsymbol{u}_{E3} \end{bmatrix} \tag{2.5}$$

$$\begin{bmatrix} \boldsymbol{b}_1 \\ \boldsymbol{b}_2 \\ \boldsymbol{b}_3 \end{bmatrix} = C(\boldsymbol{x}, \boldsymbol{v}) + g(\boldsymbol{x}) \tag{2.6}$$

In order to simulate the motion of a humanoid robot, the torque vector $\boldsymbol{\tau}$ in $\boldsymbol{u}$ should be determined. In general, the joints of a humanoid are controlled by a servo system, which involves PD control. $\boldsymbol{\tau}$ is calculated as:

$$\boldsymbol{\tau} = \mathbf{K}_p(\boldsymbol{\theta}_{\mathrm{d}} - \boldsymbol{\theta}) + \mathbf{K}_{\mathrm{v}}(\dot{\boldsymbol{\theta}}_{\mathrm{d}} - \dot{\boldsymbol{\theta}}) \tag{2.7}$$

$\boldsymbol{\theta}_d$	:	target joint angle$\in \mathbb{R}^{N\times 1}$
$\dot{\boldsymbol{\theta}}_d$	:	target joint angular velocity$\in \mathbb{R}^{N\times 1}$
$\mathbf{K}_p$	:	proportional gain$\in \mathbb{R}^{N\times N}$ diagonal matrix
$\mathbf{k}_v$	:	derivative gain$\in \mathbb{R}^{N\times N}$ diagonal matrix

A simulation with this kind of joint torque calculation is called torque control mode in dynamic simulator OpenHRP3. Because the torque control mode features a servo characteristic more likely happens in real world, it is suitable to estimate the joint torque with torque control mode. The simulations in this thesis are all simulated under torque control mode. The motion of a humanoid robot indicates a time sequential data. i. e. motion generator generates the time sequential data of joint angles and joint velocities.

2.3 Life-size Humanoid Robot HRP-2

The humanoid robot, the dynamic simulation, and the control software used in this research are described in this section.

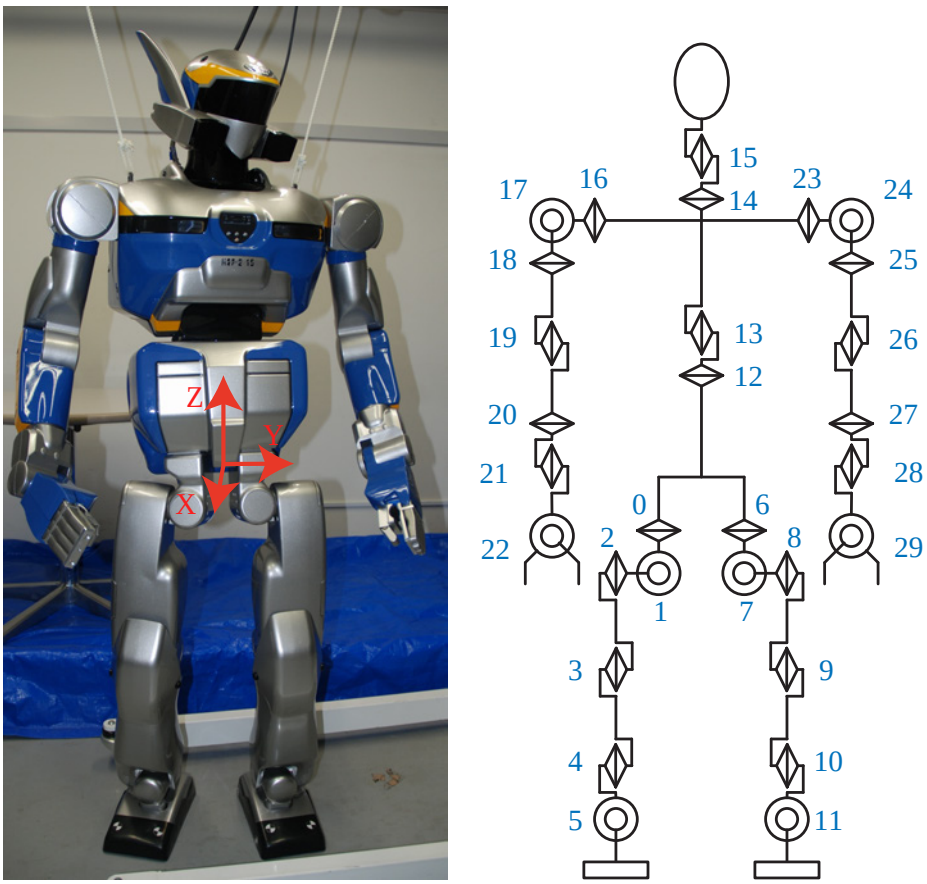
Table 2.1: Specifications : HRP-2

Size		Height: 1.54 [m], Width: 0.62 [m]
Total Weight (include Battery)		58 [kg]
Degree of Freedom	Total	30 DOF(32 DOF in HRP-2 no. 20)
	Head	2 DOF
	Arm	6 DOF(7 DOF in HRP-2 no. 20)×2 Arms
	Hand	1 DOF×2 Hands
	Waist	2 DOF
	Leg	6 DOF×2 Legs
Max Walking Speed		2.0 [km/h]
Holding Power by a Hand		20 [N]
Sensor	Joint Axis	Incremental Encoder
	Vision Sensor	3-Eyes Stereo Camera
	Body	3-Axes Vibratory Gyroscope, 3-Axes Acceleration Sensor
	Arm	6-Axes Force Sensor
	Leg	6-Axes Force Sensor
Motor Driver		48 [V], 20 [A] (Max), 2-Axes/Driver×16
Power Supply		NiMH Battery DC 48 [V], 18 [Ah]

HRP-2 is a part of humanoid robot project, which is sponsored by Japan's Ministry of Economy, Trade and Industry (METI) and New Energy and Industrial Technology Development Organization (NEDO), spearheaded by Kawada Industries and supported by the National Institute of Advanced Industrial Science and Technology (AIST) and Kawasaki Heavy Industries, Inc. HRP-2 is the final platform of this project.

HRP-2 heights 154 [cm], weights 58 [kg], and with 30 degree of freedom (DOF). The DOF of each arm is in 6-DOF, and 1-DOF hand-gripper with maximum grip strength in of 20 [N]. A Ni-MH battery is mounted. Which battery allows HRP-2 activates with no cable plugs on it for a short while. Additionally, force 6-axis sensors are mounted in two wrists and ankles, also there are a gyro and a g-sensor mounted in the body. HRP-2 is suitable to activate in a human space due to this design. The spec of HRP-2 is shown in Table 2.1.

The HRP-2 NO. 15 and No. 20 are used in this research. The appearance and the DOF of HRP-2 NO. 15 and No. 20 are shown in Figure 2.1∼Figure 2.2. Also the position of their waist coordinate are shown in Figure 2.1 and Figure 2.2. Except for the wrist joint, this two humanoid robots is in the same spec. Compare the wrist joint of HRP-2 NO.20 and HRP-2 NO.15, a *roll* joint is added in HRP-2 NO.20. The roll joint makes the arm redundant, robust in avoiding singular point, and ease the consumption energy. Furthermore, the roll joint allows the humanoid robot to move more like with human beings, and increases friendliness.

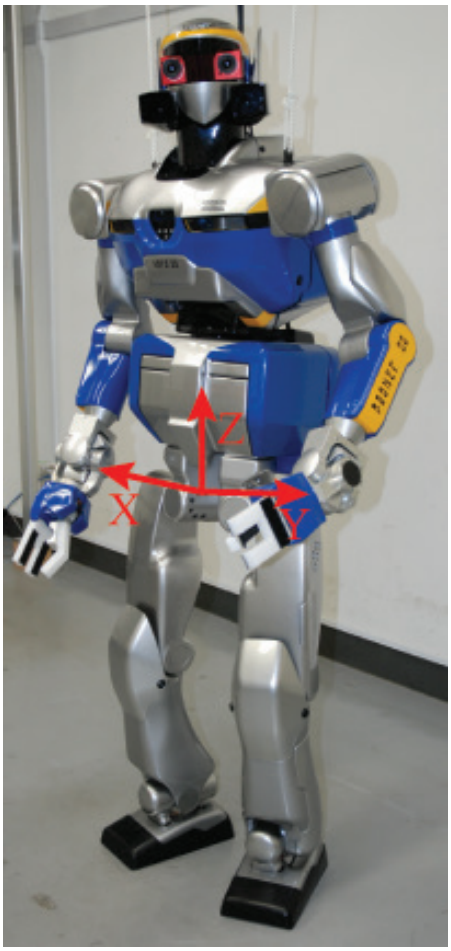


(a) Overview of HRP-2 no. 15. (b) DOF of HRP-2 no. 15.

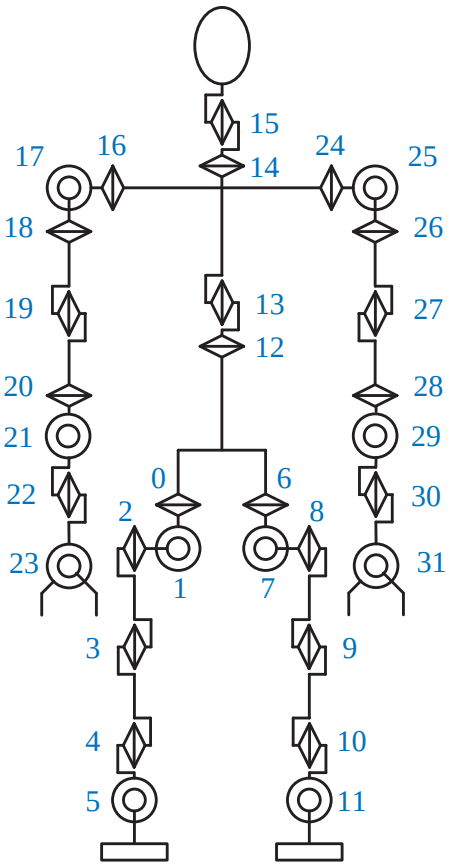
Figure 2.1: HRP-2 NO. 15.

Software Architecture

The software for controlling HRP-2, hrpsys, is based on OpenHRP(Open architecture Humanoid Robot Platform) and OpenRTM (Open robot technology middle ware) [35]. OpenHRP is introduced first in this section. OpenHRP, which is developed by AIST, is a develop environment for developing robots. OpenHRP includes dynamic simulator, and essential library API for programing. It is released as open source resource. This research use OpenHRP (OpenHRP version-3. 1. 0). OpenHRP3 consists of graphic user interface, dynamic engine (aist-dynamics), collision detector, model loader for loading robot model (Model Loader), and robot controller, which represents the robot in the simulator. The communication between each component is conducted by CORBA(Common Request Broker Architecture) [36], and by ORB(Object Request Broker) the component can communicate to each other through TCP/IP Internet. OpenHRP can load the robot models in format of VRML (Virtual Reality Modeling Language) or COLLADA (COLLAborative Design Activity) data type. Additionally, a user can build his/her own controller by use the OpenHRP library in the format of OpenRTM. The basic unit of



(a) Overview of HRP-2 no. 20.



(b) DOF of HRP-2 no. 20.

Figure 2.2: HRP-2 NO. 20.

OpenRTM is called RTC (rt component). The whole controlling system is divided into several RTCs, and each RTC represents one specific function.

Next, hrpsys is introduced, which is the control software of HRP-2. hrpsys is a software developed by General Robotics Inc.. General Robotics also participate the developing of OpenHRP, mainly in charge of the GUI part. hrpsys provides some essential control program for controlling HRP-2, which is built in RTC format. And the command is transfer to HRP-2 by hrpsys. Figure 2.5 shows the whole control system consists of RTCs.

A RTC can be developed by C++, python, and JAVA. When a user want to build a new algorithm, which is not provided by hrpsys, the user can make a new RTC to implement his/her own algorithm. The robot controller used in this research are all built in RTC format. The control cycle of HRP-2 is 5 [ms]. The motor command is transfer to the robot every 5 [ms]. And the robot controls the joint angles by running a 1 [ms] servo control loop. As a result, if all the RTCs can not finish their calculation within

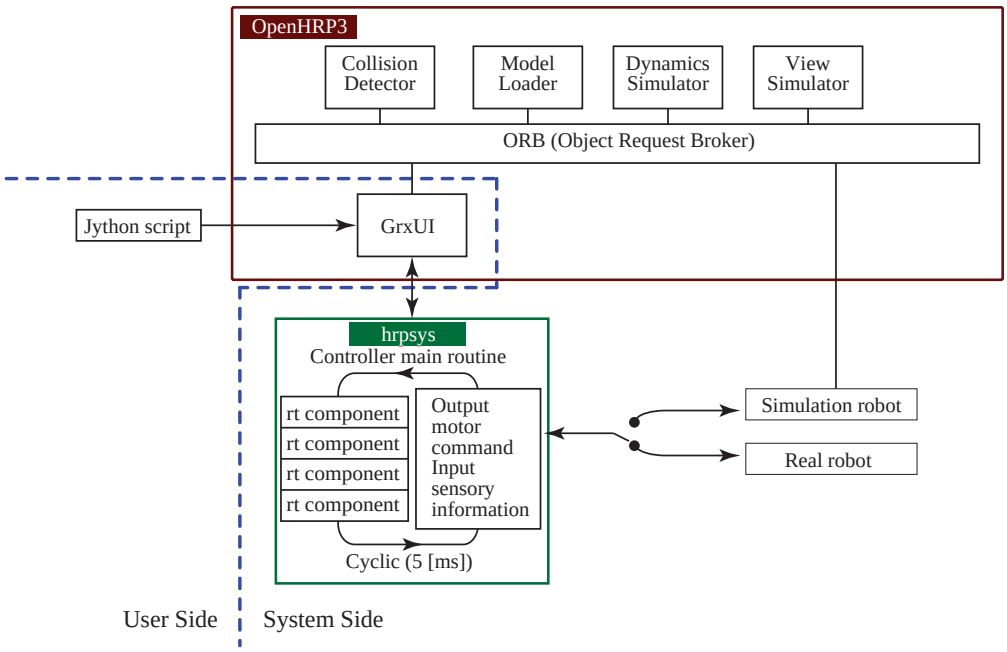


Figure 2.3: Control system software of HRP-2 (hrpsys) with OpenHRP3.

5 [ms], the robot will lose its real time performance.

Next, the relationship between OpenHRP3, hrpsys, and the robot hardware are introduced. The relationship is organized in Figure 2.3. A user can operate RTM system in console computer(local computer) or in the robot(remote computer) through a GUI of jython script. The overview of OpenHRP3 is shown in Figure 2.4. Usually if the develop environment in simulator system is different from real robot system, the controller should be rebuilt in some way. But in this kind of system there is no necessary to rebuild it. When running a simulation, a user connects the data ports of controller RTCs to the simulation robot. And when the user want to operate the real robot, all that the user have to do is reconnect the data ports to real robot instead of simulation robot. Each RTCs is called every 5 [ms] in the simulation world, and also in real world. Basically, if the simulator can imitate the experiment environment well, there is no necessary to change the program of controllers. This character features high develop efficiency because there is no needs to rebuild the controllers.

An user can use a function defined in the RTC, or give a command to the robot through GUI of Jython script [37]. Jython is an implementation of the Python programming language written in Java. Jython can import Java class seamlessly, and load/unload the RTC dynamically. Also user can change the state of a RTC, or connect the data ports between RTCs. An user also can achieve simple robot motion by the RTC provided by hrpsys through Jython.

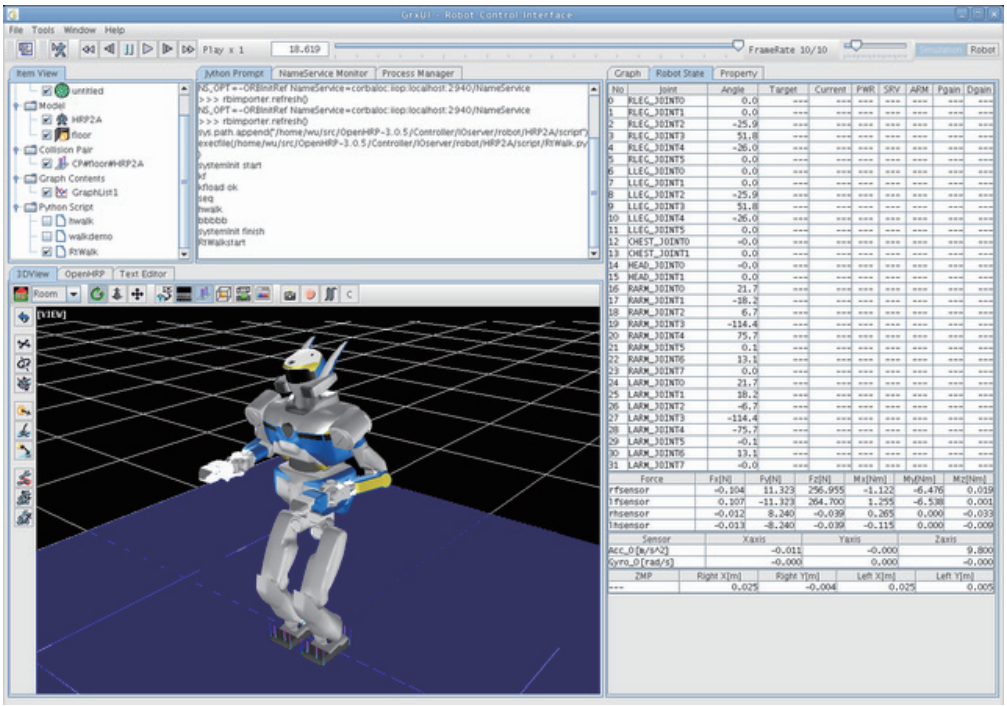


Figure 2.4: Overview of OpenHRP3.

2.4 Summary

In this chapter, the dynamic simulator is introduced, and the hardware also the software used in this research.

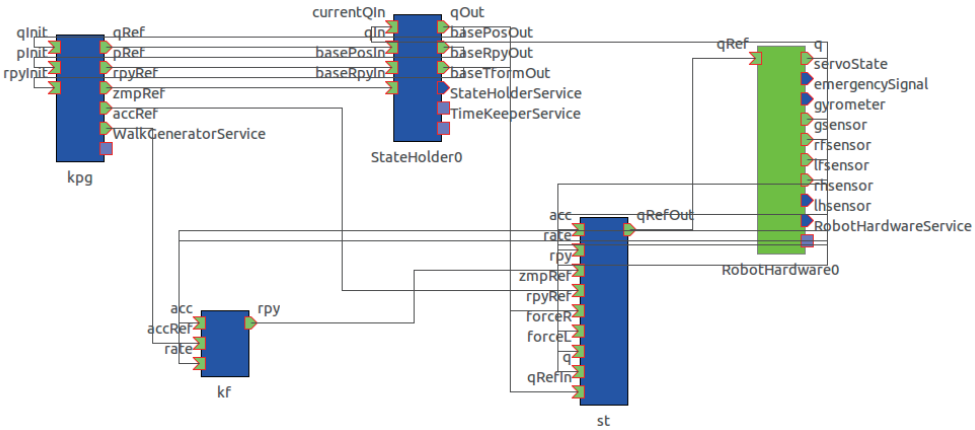


Figure 2.5: Overview of RT components of HRP-2.

Chapter 3. Online Walking Generator

3.1 Introduction

In this chapter, the online walking pattern generator method is described. Online walking pattern generator is a element technology of this research. In order to achieve high responsibility during cooperative move, a method to change the gait motion during a stepping motion is also proposed.

Although online walking consists of pattern generator and walking stabilizer, only pattern generator is discussed in this paper. Stabilizer proposed in [38] is applied in this research.

3.2 Derivation of ZMP Equation [41]

Almost all walking pattern generator method are based on ZMP control. The ZMP is introduced in this section. ZMP is propose by Miomir Vukobratovic in 1972 [39]. Vukobratovic design the ZMP trajectory within support polygon in advance, and then calculate the coincided walking motion by numerical convergence calculation. By applying ZMP theorem, it is possible to design a walking pattern that takes influence of upper body into consideration strictly. ZMP is a point in which point the moment around horizontal axis equal 0, and ZMP position can be computed from the robot motion and motion equation. The position of ZMP point is derived as follow.

When the whole momentum of a robot $\boldsymbol{P}$ and the floor reaction force $\boldsymbol{f}$ is given, the relationship between angular momentum of a robot and floor reaction moment is as

$$\dot{\boldsymbol{P}} = M\boldsymbol{g} + \boldsymbol{f} \tag{3.1}$$

$$\dot{\boldsymbol{L}} = \boldsymbol{c} \times M\boldsymbol{g} + \boldsymbol{\tau} \ , \tag{3.2}$$

where

M	:	mass.
$\boldsymbol{c} = \begin{bmatrix} x & y & z \end{bmatrix}^{\mathrm{T}}$	:	position of CoM (center of mass) ,
$\boldsymbol{P} = \begin{bmatrix} P_x & P_y & P_z \end{bmatrix}^{\mathrm{T}}$	:	linear momentum ,
$\boldsymbol{g} = \begin{bmatrix} 0 & 0 & -g \end{bmatrix}^{\mathrm{T}}$	:	gravity ,
$\boldsymbol{L} = \begin{bmatrix} L_x & L_y & L_z \end{bmatrix}^{\mathrm{T}}$	:	angular momentum .

It is assumed that the external forces act on the robot are the floor reaction force and moment generate from the ZMP only. Let $\boldsymbol{p}$ be ZMP, $\boldsymbol{\tau}_p$ be the moment acts at ZMP, the relationship between them is

$$\boldsymbol{\tau}_p = \boldsymbol{p} \times \boldsymbol{f} + \boldsymbol{\tau}_p \ . \tag{3.3}$$

from (3.2), (3.3), (3.1), τ_p is resolved as

$$\dot{\boldsymbol{\tau}}_p = \dot{\boldsymbol{L}} - \boldsymbol{c} \times M\boldsymbol{g} + (\dot{\boldsymbol{P}} - M\boldsymbol{g}) \times \boldsymbol{p} \tag{3.4}$$

Then the x and y segment of $\boldsymbol{\tau}_p$ are

$$\tau_{px} = \dot{\boldsymbol{L}}_x + M\boldsymbol{g}y + \dot{\boldsymbol{P}}_y p_z - (\dot{\boldsymbol{P}}_z + M\boldsymbol{g})p_y = 0 \ , \tag{3.5}$$

$$\tau_{py} = \dot{\boldsymbol{L}}_y + M\boldsymbol{g}x + \dot{\boldsymbol{P}}_x p_z - (\dot{\boldsymbol{P}}_z + M\boldsymbol{g})p_x = 0 \ , \tag{3.6}$$

Because the x and y segment of moment at ZMP equal 0, τ_{px} and τ_{py} are 0. With this condition, p_x and p_y can be solve from (3.5) and (3.6) as follow

$$p_x = \frac{Mgx + p_z\dot{P}_x - \dot{L}_y}{Mg + \dot{P}_z} \ , \tag{3.7}$$

$$p_y = \frac{Mgy + p_z\dot{P}_y + \dot{L}_x}{Mg + \dot{P}_z} \ , \tag{3.8}$$

where p_z is the height of ground. z equals to 0 on a flat surface.

Furthermore, if the mass of the robot is replaced with a single mass model, the liner momentum and the angular momentum about the origin are

$$\boldsymbol{P} = M\dot{\boldsymbol{c}} \ , \tag{3.9}$$

$$\boldsymbol{L} = \boldsymbol{c} \times M\dot{\boldsymbol{c}} \ . \tag{3.10}$$

Derivative of (3.9), (3.10) are

$$\begin{bmatrix} \dot{P}_x \\ \dot{P}_y \\ \dot{P}_z \end{bmatrix} = \begin{bmatrix} M\ddot{x} \\ M\ddot{y} \\ M\ddot{z} \end{bmatrix} \ , \tag{3.11}$$

$$\begin{bmatrix} \dot{L}_x \\ \dot{L}_y \\ \dot{L}_z \end{bmatrix} = \begin{bmatrix} M(y\ddot{z} - z\ddot{y}) \\ M(z\ddot{x} - x\ddot{z}) \\ M(x\ddot{y} - y\ddot{x}) \end{bmatrix} \ . \tag{3.12}$$

From (3.11), (3.12), (3.7), and (3.8), ZMP is computed as

$$p_x = x - \frac{(z - p_z)\ddot{x}}{\ddot{z} + g} \tag{3.13}$$

$$p_y = y - \frac{(z - p_z)\ddot{y}}{\ddot{z} + g} \tag{3.14}$$

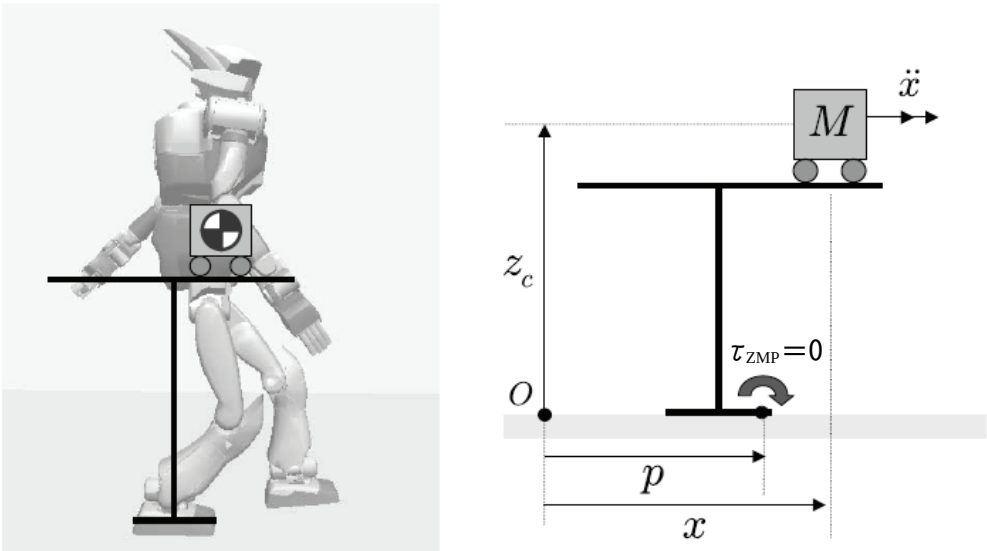


Figure 3.1: A cart-table model [41]

3.3 Preview Control Pattern Generator

Applying preview control theory to generate the trajectory of center of mass (CoM) that follows desired ZMP is proposed by Kajita et al. [40]. This method can generate a trajectory of CoM that corresponds desired zero moment point (ZMP) at the starting point of the trajectory, so the method is suitable for highly frequent repetitive pattern generation. Since the pattern generation cycle time is 5 [ms] in this research, it is proper to use the method. The method will be reviewed for further explanation.

A humanoid robot is modeled by a cart whose mass is M, and a mass-less table (Figure 3.1). p and x in Figure 3.1 are x coordinates of the ZMP and CoM,respectively. ZMP equation can be led as follows:

$$p = x - \frac{z_c}{g}\ddot{x} \ , \tag{3.15}$$

where g is gravity acceleration. Then (3.15) can be translated into a strictly proper dynamical system as:

$$\frac{d}{dt}\boldsymbol{x} = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{bmatrix} \boldsymbol{x} + \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} u \ ,$$

$$p = \begin{bmatrix} 1 & 0 & -z_c/g \end{bmatrix} \boldsymbol{x}, \ u \equiv \dddot{x} \ , \tag{3.16}$$

where $\boldsymbol{x} \equiv \begin{bmatrix} x & \dot{x} & \ddot{x} \end{bmatrix}^{\mathrm{T}}$. Let input u of (3.16) be the time derivative of $\ddot{x}$, then (3.16) is discretized with sampling time of Δt (5 [ms]) as follows:

$$\boldsymbol{x}_{k+1} = \mathbf{A}\boldsymbol{x}_k + \boldsymbol{b}u_k \ ,$$

$$p = \boldsymbol{c}\boldsymbol{x}_k \ , \tag{3.17}$$

where

$$\mathbf{A} \equiv \begin{bmatrix} 1 & \Delta t & \frac{\Delta t^2}{2} \\ 0 & 1 & \Delta t \\ 0 & 0 & 1 \end{bmatrix} , \ \boldsymbol{b} \equiv \begin{bmatrix} \frac{\Delta t^3}{6} \\ \frac{\Delta t^2}{2} \\ \Delta t \end{bmatrix} , \ \boldsymbol{c} \equiv \begin{bmatrix} 1 & 0 & \frac{-z_c}{g} \end{bmatrix} ,$$

$$\boldsymbol{x}_k \equiv \begin{bmatrix} x(k\Delta t) & \dot{x}(k\Delta t) & \ddot{x}(k\Delta t) \end{bmatrix}^{\mathrm{T}} .$$

In order to treat this state equation as error system, the ZMP error is defined as $e_k \equiv p_k^{ref} - p_k$, and the follow extended state equation is led

$$\begin{bmatrix} e_{k+1} \\ \Delta \boldsymbol{x}_{k+1} \end{bmatrix} = \begin{bmatrix} 1 & -\boldsymbol{c}\mathbf{A} \\ 0 & \mathbf{A} \end{bmatrix} \begin{bmatrix} e_k \\ \Delta \boldsymbol{x}_k \end{bmatrix} + \begin{bmatrix} -\boldsymbol{c}\boldsymbol{b} \\ \boldsymbol{b} \end{bmatrix} \Delta u_k + \begin{bmatrix} 1 \\ 0 \end{bmatrix} \Delta p_{k+1}^{ref} , \tag{3.18}$$

where $\Delta \boldsymbol{x}_k \equiv \boldsymbol{x}_k - \boldsymbol{x}_{k-1}$, $\Delta u_k \equiv u_k - u_{k-1}$, $\Delta p_k^{ref} \equiv p_k^{ref} - p_{k-1}^{ref}$. (3.18) is rewritten as

$$\boldsymbol{X}_{k+1} = \Phi X_k + \boldsymbol{G}\Delta u_k + \boldsymbol{G}_R \Delta p_{k+1}^{ref} \tag{3.19}$$

$$\boldsymbol{X}_k \equiv \begin{bmatrix} e_k \\ \Delta \boldsymbol{x}_k \end{bmatrix}, \Phi \equiv \begin{bmatrix} 1 & -\boldsymbol{c}\mathbf{A} \\ \mathbf{0} & \mathbf{A} \end{bmatrix}$$

$$\boldsymbol{G} \equiv \begin{bmatrix} -\boldsymbol{c}\boldsymbol{b} \\ \boldsymbol{b} \end{bmatrix}, \boldsymbol{G}_R \equiv \begin{bmatrix} 1 \\ 0 \end{bmatrix}$$

In order to obtain the optimal input u_k, a performance index is given as:

$$J = \sum_{k=1}^{\infty} \left\{ Q \left(p_k^{ref} - p_k \right)^2 + R\Delta u_k^2 \right\} , \tag{3.20}$$

where p_k^{ref} is reference ZMP, and Q and R are positive values. When Q is lager than R gratefully, the convergence of ZMP to reference ZMP becomes faster, but it also result in a rapidly acceleration of waist link. And when R is gratefully larger than Q, the acceleration of waist link becomes slow but also the convergence of ZMP to reference ZMP.

When the ZMP reference can be previewed for N steps, the optimal input Δu_k that minimize the performance index is given by:

$$\Delta u_k = -\boldsymbol{K} \begin{bmatrix} p_k^{ref} - p_k \\ \Delta \boldsymbol{x}_k \end{bmatrix} + \sum_{j=1}^{N} f_j \Delta p_{k+j}^{ref} , \tag{3.21}$$

where $\Delta \boldsymbol{x}_k \equiv \boldsymbol{x}_k - \boldsymbol{x}_{k-1}$, $\Delta p_k^{ref} \equiv p_k^{ref} - p_{k-1}^{ref}$, and $\boldsymbol{K}$ and f_j are the gains calculated from Q, R and z_c as

$$\boldsymbol{K} = [R + \boldsymbol{G}^{\mathrm{T}}\mathbf{P}\boldsymbol{G}]^{-1}\boldsymbol{G}^{\mathrm{T}}\mathbf{P}\Phi$$

$$f_j = -[R + \boldsymbol{G}^{\mathrm{T}}\mathbf{P}\boldsymbol{G}]^{-1}\boldsymbol{G}^{\mathrm{T}}(\xi^{\mathrm{T}})^{j-1}\mathbf{P}\boldsymbol{G}_R$$

$$\xi \equiv [\mathbf{I} - \boldsymbol{G}[R + \boldsymbol{G}^{\mathrm{T}}\mathbf{P}\boldsymbol{G}]^{-1}\boldsymbol{G}^{\mathrm{T}}\mathbf{P}\Phi]$$

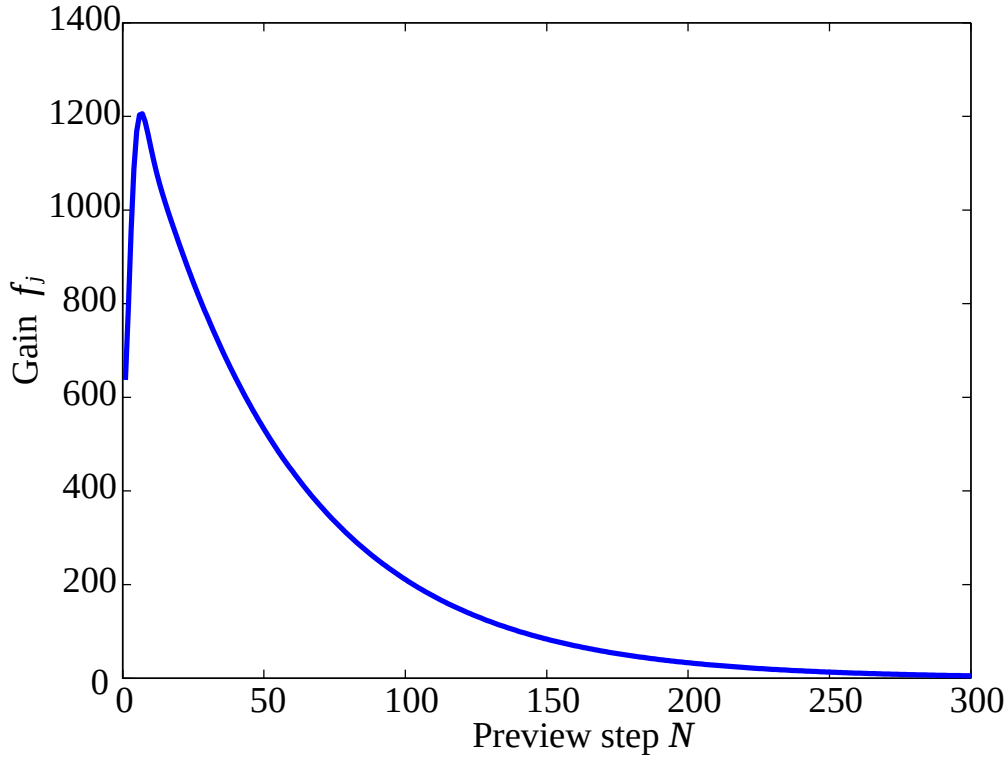


Figure 3.2: The value of preview gain f_j

$\mathbf{P}$ is solve from Ricatti equation as

$$\mathbf{P} = \Phi^{\mathrm{T}} P\phi + G_R^{\mathrm{T}} Q G_R - \Phi^{\mathrm{T}} \mathbf{P} G (R + G_R^{\mathrm{T}} \mathbf{P} G_R)^{-1} G_R^{\mathrm{T}} \mathbf{P} \Phi \ .$$

Ricatti equation has to be solved numerically in order to decide $\boldsymbol{K}$ and f_j. However, z_c is kept at constant in this paper, hence $\boldsymbol{K}$ and f_j can be computed in advance, and they are constant. Figure 3.2 shows the value of f_j obtained by solving the Ricatti equation. Here Q, R are set to 1 and 10^{-6} respectively, and f_j is calculated till 300 steps (i.e. 1.5 [s] since the sampling time Δt is 5 [ms]). [h]

3.4 Determination of Preview Span

The value f_j along j is illustrated in Figure 3.2. From this graph, the value converge to 0 about from $N = 300$. Since the sampling time of the system is 1.5 [s], it is sufficient to calculate optimal input u_k by using the future reference ZMP until 1.5 [s] later.

From above explanation, the input u_k only dependent on the state x_k at timing k and the incoming reference ZMP until N steps later. The CoM position of next step x_{k+1} is calculated from these inputs. The trajectory of y direction also can be computed in the same way. The preview control theorem is organized in Figure 3.3. By Solving these discrete equation along time line, the reference CoM trajectory is obtained.

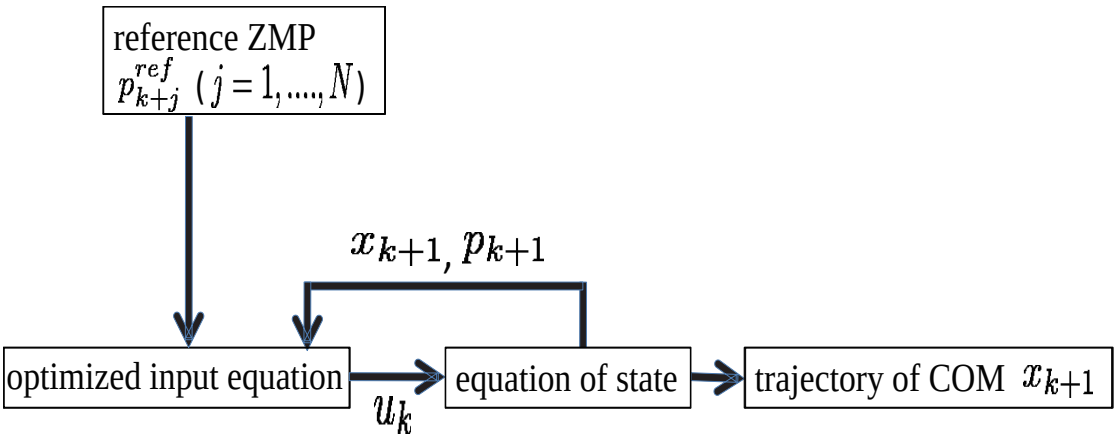


Figure 3.3: Summation of preview control.

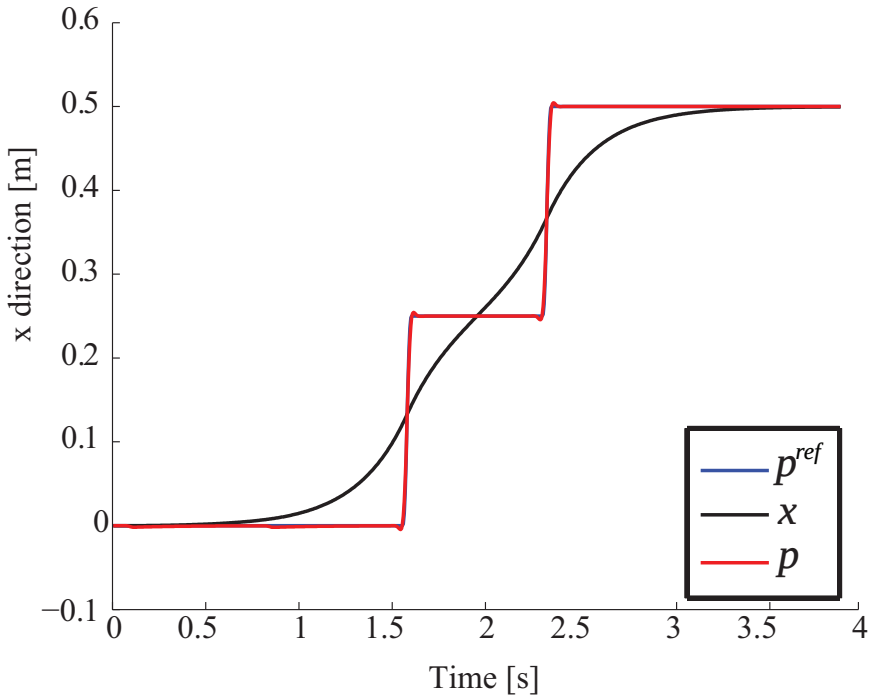


Figure 3.4: Trajectory of x direction.

The ZMP tracking effect is confirm in Figure 3.4 and Figure 3.5. In this simulation, $Q = 1$, $R = 10^{-6}$, respectively. There is a huge oscillation of ZMP in y direction at the beginning. This is due to the feed forward member in (3.21), and this is a characteristic of preview control. In order to ease this oscillation, a steady motion is added in front of the first step as illustrated in Figure 3.6.

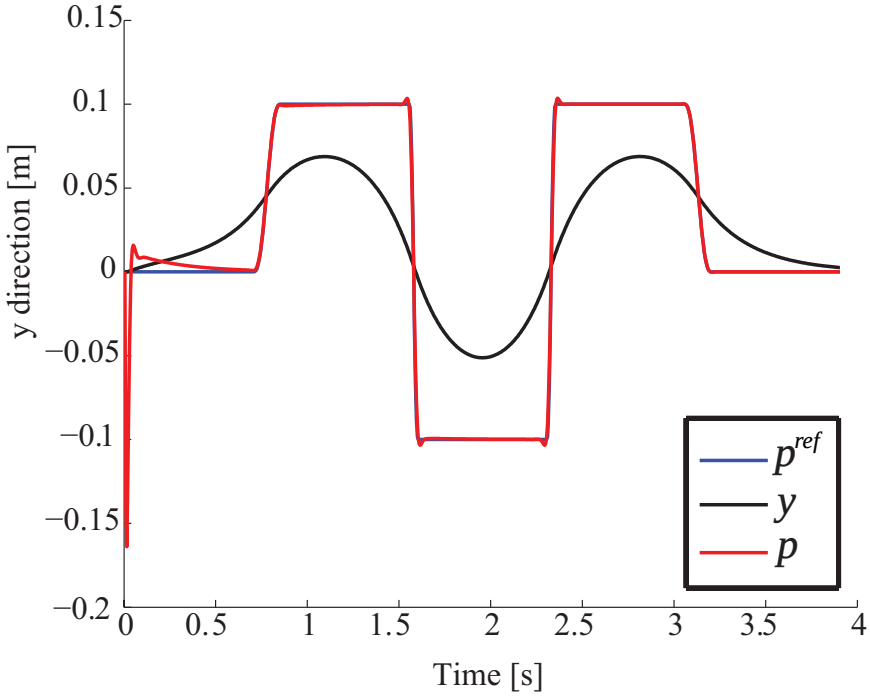


Figure 3.5: Trajectory of y direction.

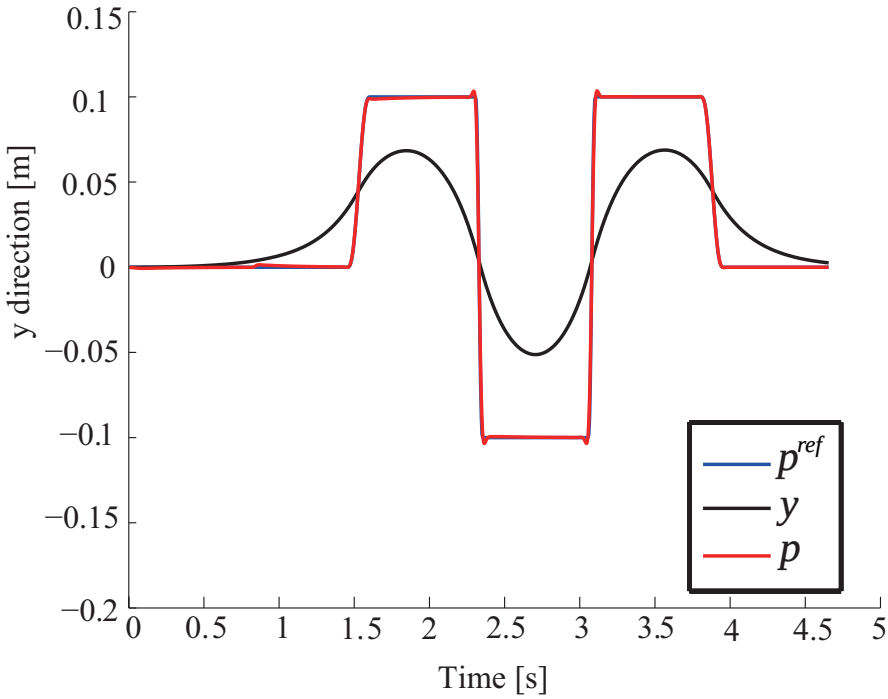


Figure 3.6: Trajectory of y direction(Initial position is extended).

3.5 Walking Planning

The procedure of walking planning is as follow.

1. Set the CoM state, ZMP position.
2. Plan the reference ZMP trajectory.
3. Compute CoM trajectory by preview control.
4. Compute joint angles.
5. Transform the reference ZMP in robot coordinate for applying stabilization control.

The CoM state is defined as a 3×1 vector which includes position, velocity, and acceleration. In order to generate walking motion of one step, the follow five parameters are required.

1. T_d: The interval of double support phase.
2. T_s: The interval of single support phase.
3. z_c:Height of CoM
4. Z_{up}:Height of swing leg.
5. $prmx, prmy, prm\theta$:Walking parameters.

Walking parameter is a walking command includes the relative distance between the next landing place and the supporting foot, and the rotation motion of upper body. One step is decomposed to

- $\frac{T_d}{2}$[s] of double supporting phase (first half).
- T_s[s] of single supporting phase.
- $\frac{T_d}{2}$[s] of double supporting phase (last half).

The overall walking motion is to repeat $(T_s + T_d)$ [s] motion n times. But due to the characteristic of preview control, the double supporting phase span of first step is extended to $(T_s + \frac{3}{2}T_d)$ [s]. The data length of one step is illustrated in Figure 3.7. The boundary of one step in programming is the middle of double supporting phase. Except the first step, the length of one unit step is $\frac{1}{4}T_d + T_s$. In this paper, k-step indicates this data structure. k-step start at the middle of double support phase, and end at the middle of next double support phase. The origin of world coordinate of one step is at the middle point of a supporting foot. After a step is done, the origin of world coordinate is switched to the other foot. The origin is in left or right foot in turn.

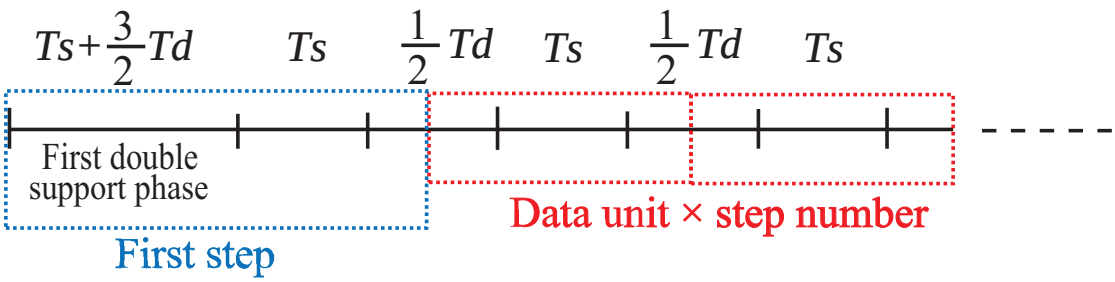


Figure 3.7: The time axis of pattern generation.

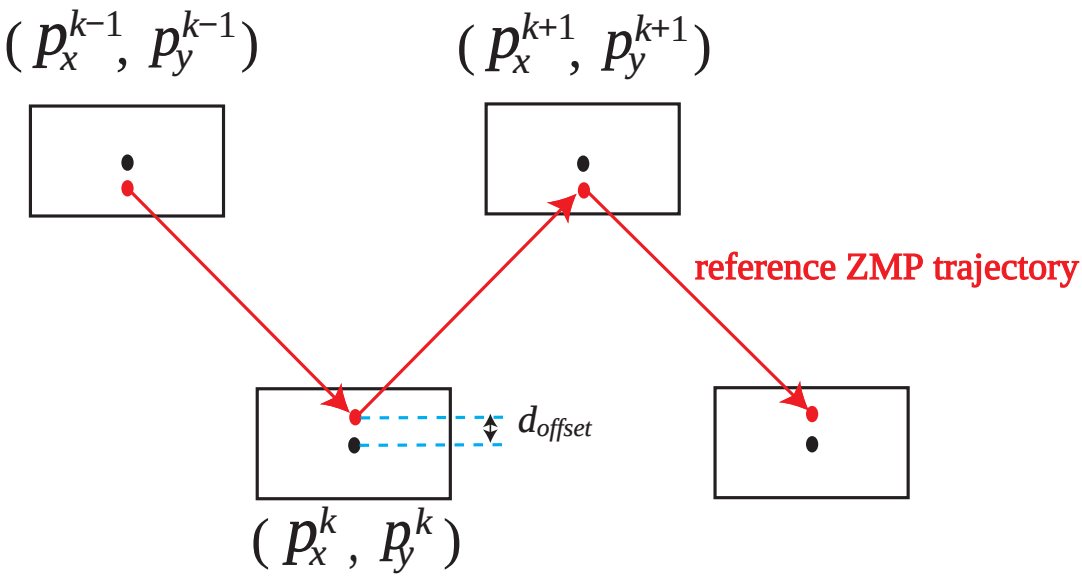


Figure 3.8: Reference ZMP trajectory.

Table 3.1: Length of reference ZMP.

1st step	1.575 [s]
Step other than 1st step	0.75 [s]
1st step(extended with preview span)	3.075 [s]
Step other than 1st step(extended with preview span)	2.25 [s]

Table 3.2: Parameters of sample motion

T_d [s]	0.1
T_s [s]	0.7
Z_{up} [m]	0.05
($prmx$ [m], $prmy$ [m], $prm\theta$ [°])	(0.10, 0.19, 0.0) (0.10, 0.19, 0.0) (0.10, 0.19, 0.0)

3.5.1 Generation of Reference ZMP Trajectory

In order to calculate the reference ZMP, first the center of supporting foot of k-step is calculated from the given walking parameter. The reference ZMP of k-step is set at a certain offset d_{offset} from the center of supporting foot (p_x^k, p_y^k). By setting the reference ZMP a little bit inside form the foot center in single supporting phase, displacement of upper body in y direction is reduced. d_{offset} is set to = 10 [mm].

The trajectory of reference ZMP is moving from (p_x^{k-1}, p_y^{k-1}) to (p_x^k, p_y^k) in double supporting phase (first half), and from (p_x^k, p_y^k) to (p_x^{k+1}, p_y^{k+1}) in double supporting (last half), respectively. Both trajectory is calculated in fifth polynomial. The reference ZMP trajectory is illustrated in Figure 3.8. The reference CoM is set to the initial ZMP position (p_x^0, p_y^0) for T_s [s] in double supporting phase of first step, and then move to (p_x^1, p_y^1) in $\frac{3}{2}T_d$ [s]. The velocity and acceleration at both begin and end of one trajectory is set to 0. This is also the boundary conditions of fifth polynomial.

For an instance, trajectory of a sample motion as described in Table 3.2 and Figure 3.9 is computed. In this sample, robot starts with right leg, so the ZMP trajectory of first step is computed in left foot coordinate. The reference ZMP position is set to CoM position in initial posture. In order to apply preview control, expect the reference ZMP of first step, additional reference ZMP trajectory for 1.5 [s] is needed. The current walking parameter is treated as next walking parameter for calculate the future reference ZMP trajectory. The length of reference ZMP is organized in Table 3.1.

The reference ZMP and reference CoM trajectory of sample motion is shown in Figure 3.10 and Figure 3.11, respectively.

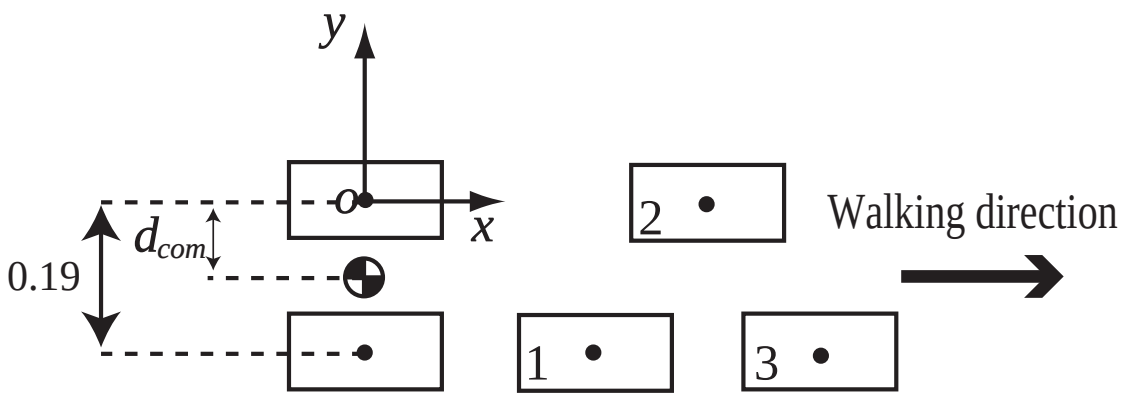
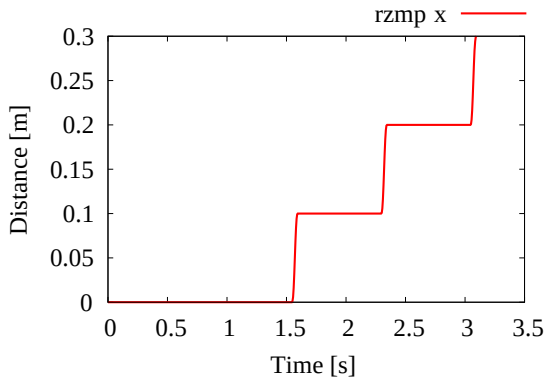
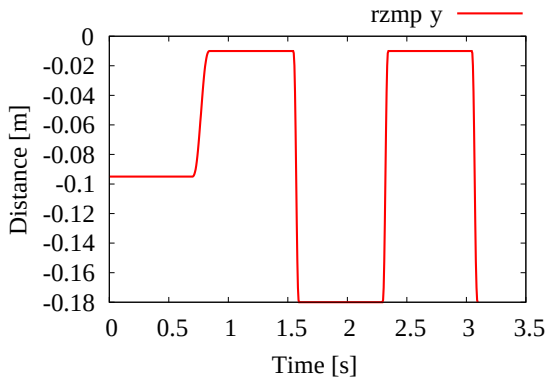


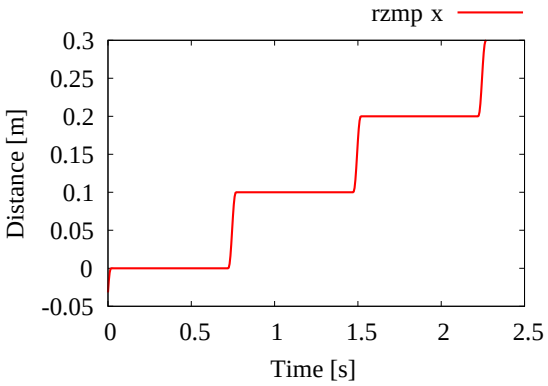
Figure 3.9: Overview of sample motion.



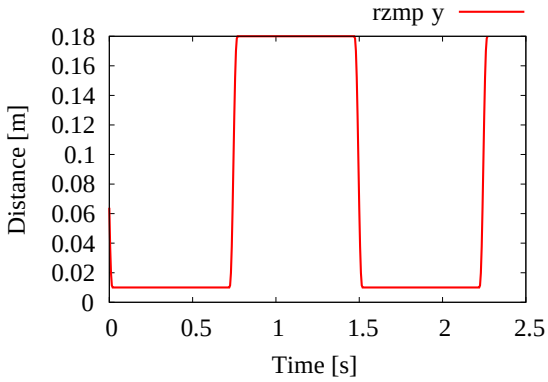
(a)Reference ZMP of 1st step(x).



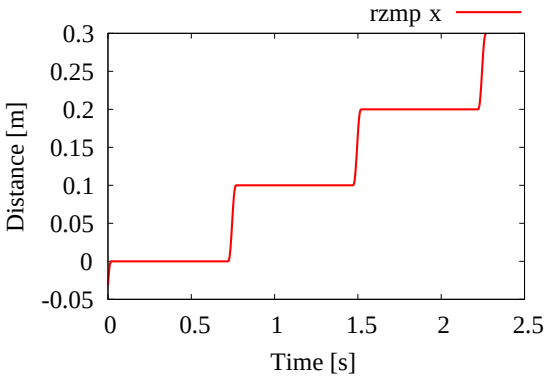
(b)Reference ZMP of 1st step(y).



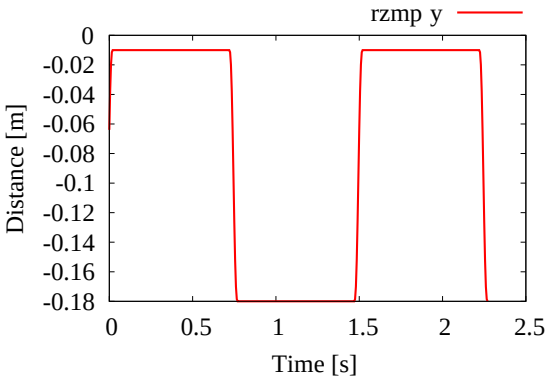
(c)Reference ZMP of 2nd step(x).



(d)Reference ZMP of 2nd step(y).

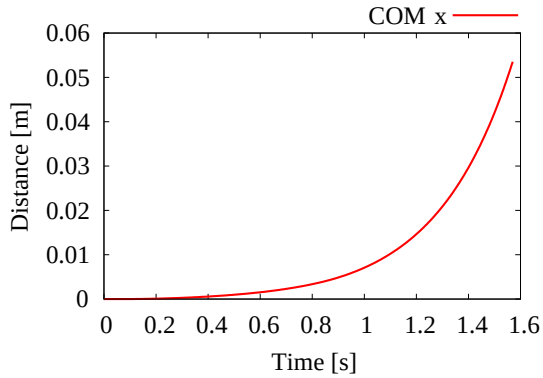


(e)Reference ZMP of 3rd step(x).

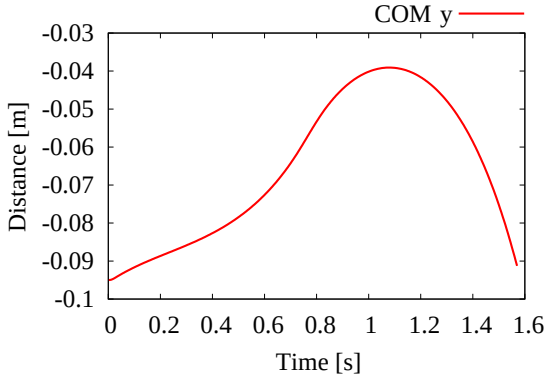


(f)Reference ZMP of 3rd step(y).

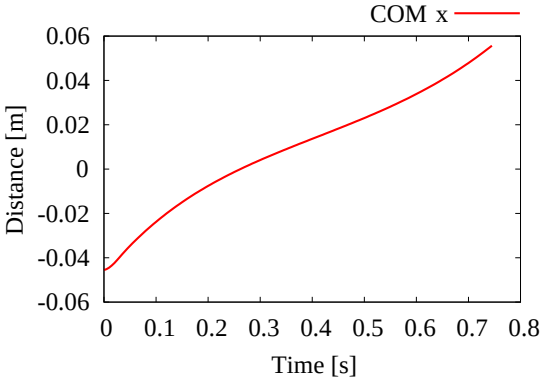
Figure 3.10: The reference ZMP of test motion.



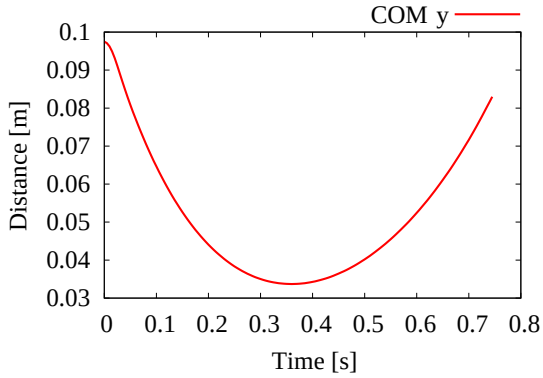
(a)The COM trajectory of 1st step(x).



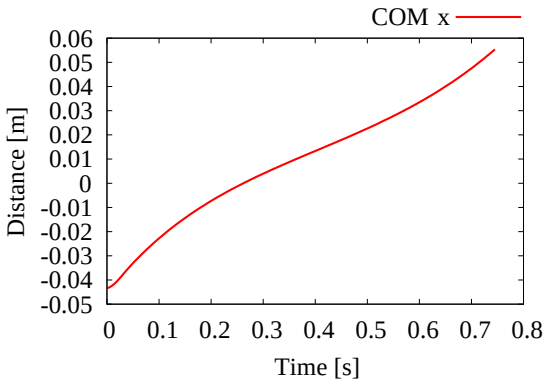
(b)The COM trajectory of 1st step(y).



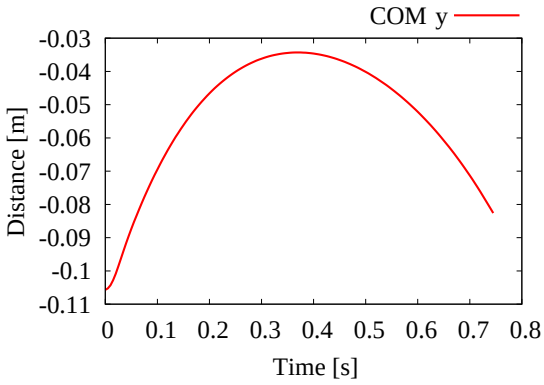
(c)The COM trajectory of 2nd step(x).



(d)The COM trajectory of 2nd step(y).



(e)The COM trajectory of 3rd step(x).



(f)The COM trajectory of 3rd step(y).

Figure 3.11: The center of mass trajectory of test motion.

3.5.2 Calculation of Swing Foot Trajectory

In this subsection, the trajectory of a swing foot based on walking parameter is discussed.

x, y, z, and the rotation angle about vertical axis θ of upper body is calculated independently. Walking parameter $prmx$ and $prmy$ are treated as relative displacement with respect to supporting foot coordinate. The trajectory of x, y direction is computed as (3.22) and (3.23).

$$p_{x,swing} = \begin{cases} p^o_{x,swing} + prmx^1(12t^5 - 30t^4 + 20t^3) & (a) \\ p^o_{x,swing} + (prmx^{k-1} + prmx^k)(12t^5 - 30t^4 + 20t^3) & (b) \end{cases} \quad (t = 0 \sim T_s - 2 \times T_{LD}) \tag{3.22}$$

$$p_{y,swing} = \begin{cases} p^o_{y,swing} + (0.19 - prmy^1)(12t^5 - 30t^4 + 20t^3) & (a) \\ p^o_{y,swing} + (-0.19 + prmy^1)(12t^5 - 30t^4 + 20t^3) & (b) \\ p^o_{y,swing} + (-prmy^{k-1} + prmy^k)(12t^5 - 30t^4 + 20t^3) & (c) \\ p^o_{y,swing} + (prmy^{k-1} - prmy^k)(12t^5 - 30t^4 + 20t^3) & (d) \end{cases} \quad (t = 0 \sim T_s - 2 \times T_{LD}) \tag{3.23}$$

$p^o_{x,swing}$ and $p^o_{y,swing}$ are swing foot position in world coordinate before lifting off. T_{LD} is a time span for vertical lift off and landing. In (3.22), (a) and (b) are trajectory computing of first step and k-step. Except first step, the calculation includes the previous walking parameter.

(3.23) is calculated in the same way, but is classified into two way in case of right foot and left foot. In case of right foot swinging, $(a)(d)$ is adapted, and $(b)(c)$ is adapted in case of left. 0.19 is displacement between right and left foot in initial posture.

The trajectory in z is consisted of two curve calculated by third-polynomial. Both velocity is 0 at start and end point. The swing foot height is set in advance, and then the trajectory in z direction is computed as (3.24).

$${}^{w}p_{z,swing} = \begin{cases} Z_{up}(\frac{12}{T_s^2}t^2 - \frac{16}{T_s^3}t^3) & (when\ 0 < t \leq \frac{T_s}{2}) \\ Z_{up}(\frac{-12}{T_s^2}t^2 + \frac{16}{T_s^3}t^3) & (when\ \frac{T_s}{2} < t \leq T_s) \end{cases} \quad (t = 0 \sim T_s) \tag{3.24}$$

The attitude of upper body (waist link) is always the middle of supporting foot and swinging foot in yaw direction. That is,

$$\phi_{waist} = (\phi_{swing} + \phi_{supporting})/2 \ . \tag{3.25}$$

With the reference motion trajectory, the reference joint angles of two legs are calculated based on the following kinematic equations.

$$\begin{bmatrix} \dot{\boldsymbol{r}}_{\mathrm{CoM}} \\ \dot{\boldsymbol{v}}_{\mathrm{waist}} \\ \dot{\boldsymbol{p}}_{\mathrm{swingFoot}} \end{bmatrix} = \begin{bmatrix} \mathbf{J}_{\mathrm{CoM}} \\ \mathbf{J}_{\mathrm{waist}} \\ \mathbf{J}_{\mathrm{swingFoot}} \end{bmatrix} \dot{\boldsymbol{\Theta}} \ . \tag{3.26}$$

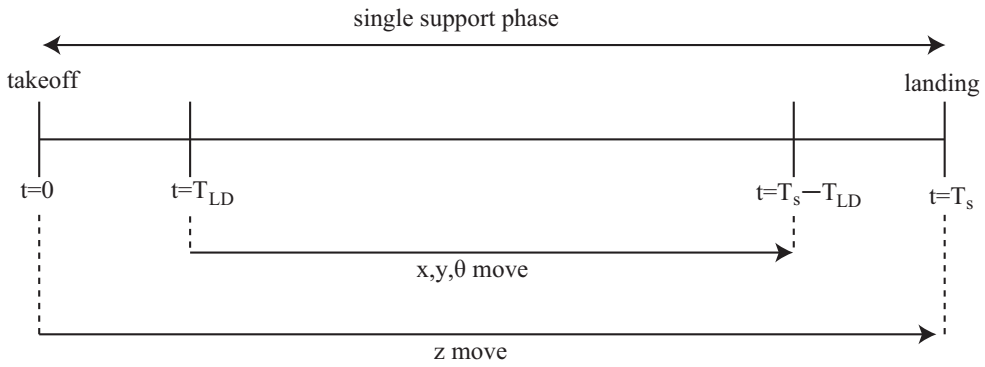


Figure 3.12: Schedule of single support phase.

Table 3.3: Parameters of sample motion2

T_d [s]	0.7
T_s [s]	0.3
Z_{up} [m]	0.05
($prmx$ [m], $prmy$ [m], $prm\theta$ [°])	(0.10, 0.19, 5)×5

The control variables includes

$\boldsymbol{r}_{\mathrm{CoM}} \in \mathbb{R}^3$: position of center of mass,

$\boldsymbol{v}_{\mathrm{waist}} \in \mathbb{R}^3$: attitude of waist,

$\boldsymbol{p}_{\mathrm{swingFoot}} \in \mathbb{R}^6$: position and attitude of swing foot,

$\dot{\boldsymbol{\Theta}}$: joint angular velocity of two legs,

$\mathbf{J}_{\mathrm{CoM}}$: Jacobian matrix mapping $\dot{\boldsymbol{\Theta}}$ to $\dot{\boldsymbol{r}}_{\mathrm{CoM}}$,

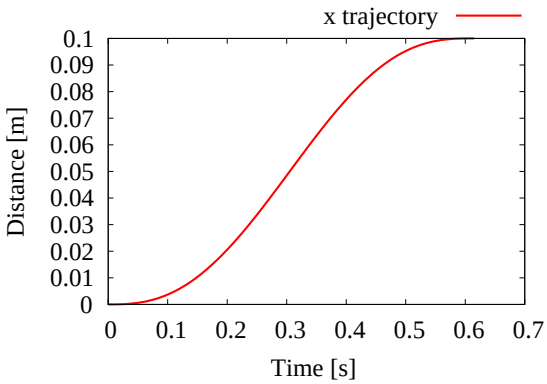
$\mathbf{J}_{\mathrm{waist}}$: Jacobian matrix mapping $\dot{\boldsymbol{\Theta}}$ to $\dot{\boldsymbol{v}}_{\mathrm{waist}}$,

$\mathbf{J}_{\mathrm{swingFoot}}$: Jacobian matrix mapping $\dot{\boldsymbol{\Theta}}$ to $\dot{\boldsymbol{p}}_{\mathrm{swingFoot}}$,

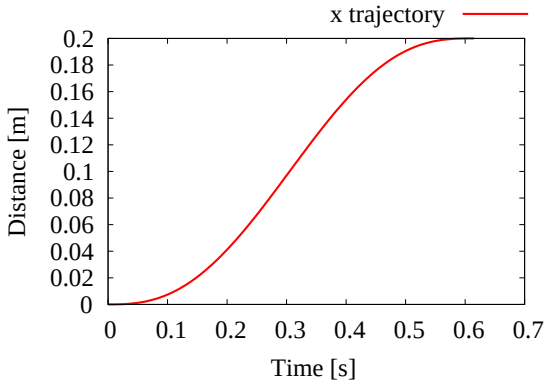
The swinging foot trajectory is shown in Figure 3.13 ($T_{LD} = 0.04$). This is a strait walking motion, so the displacement in y direction is always 0.

The walking simulation with walking parameter Table 3.2 is shown in Figure 3.14.

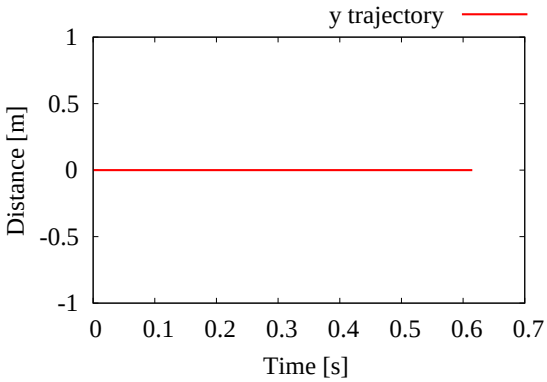
Another walking simulation with parameter Table 3.3 is conducted. The simulation is shown in Figure 3.15. The robot turn 5 [°] every step.



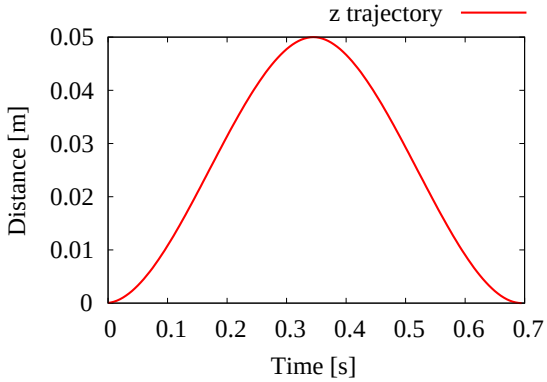
(a)x trajectory of swing leg of 1st step.



(b)x trajectory of swing leg other than 1st step.

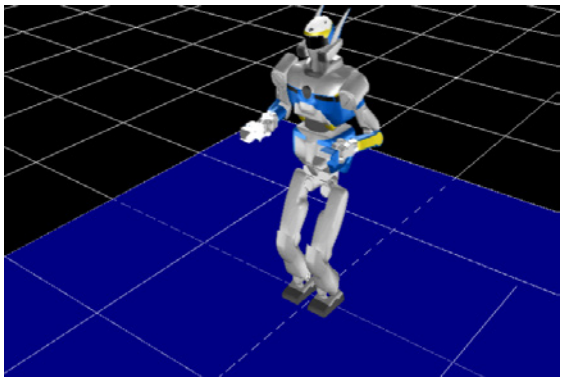


(c)y trajectory of swing leg.

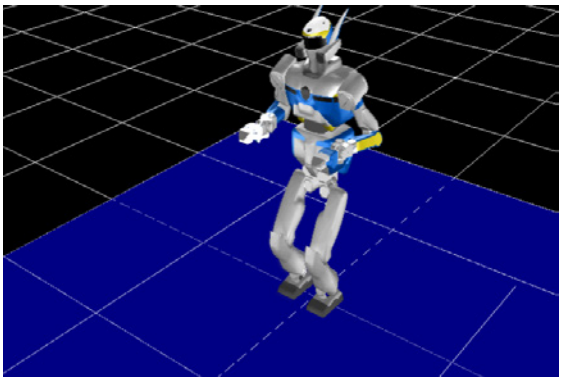


(d)z trajectory of swing leg.

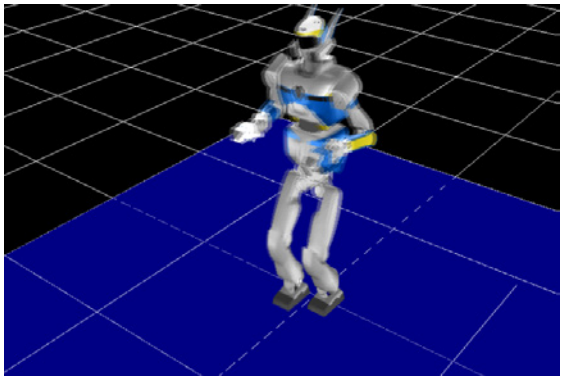
Figure 3.13: x,y,z trajectory in test motion.



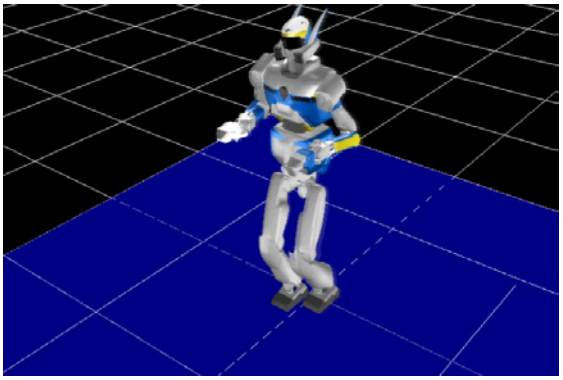
(a)t=9.67 [s]



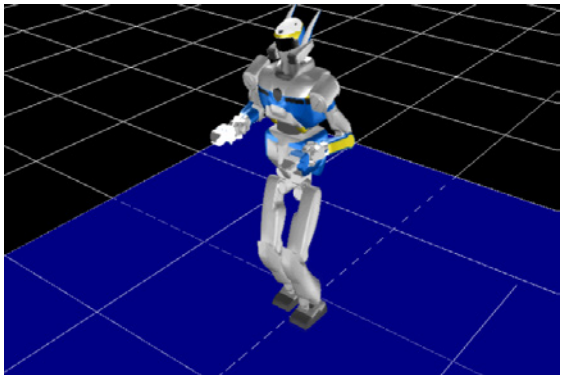
(b)t=10 [s]



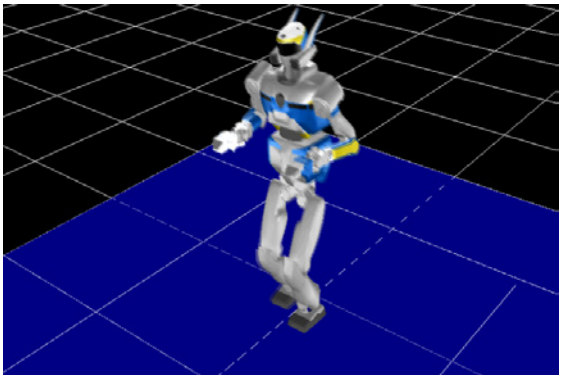
(c)t=10.33 [s]



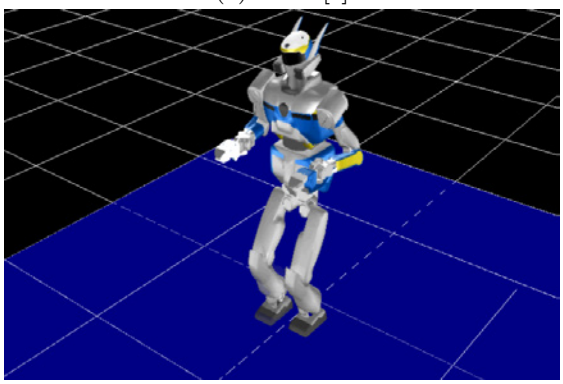
(d)t=10.67 [s]



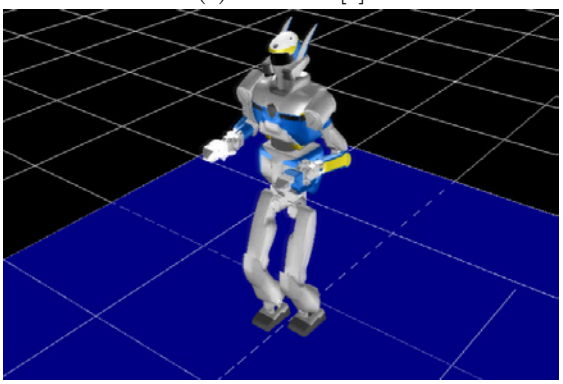
(e)t=11 [s]



(f)t=11.33 [s]

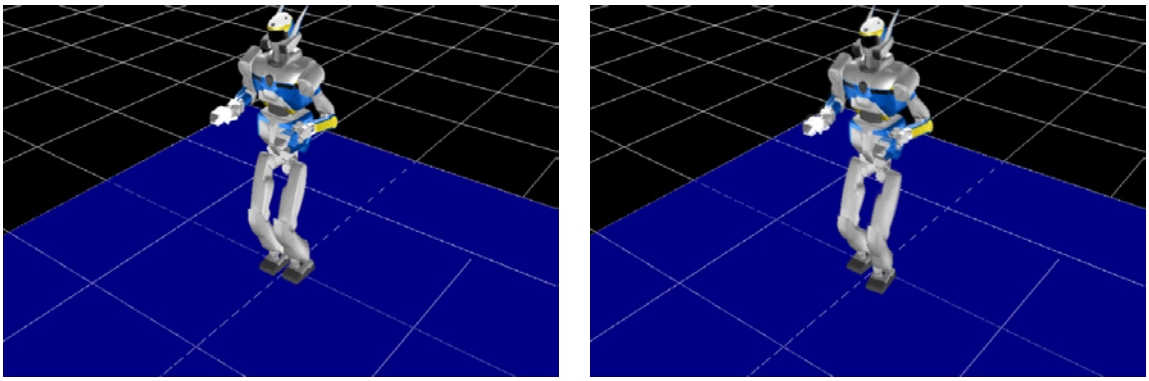


(g)t=11.67 [s]



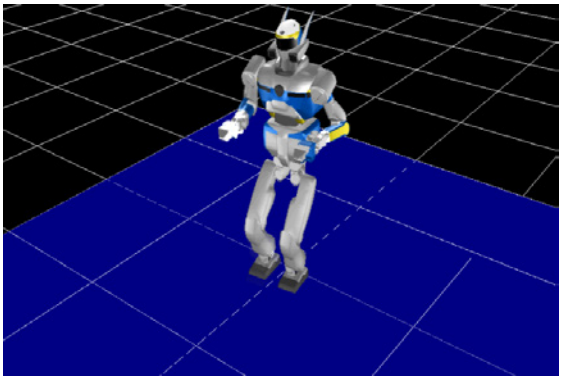
(h)t=12 [s]

Figure 3.14: Simulation of walking in line.

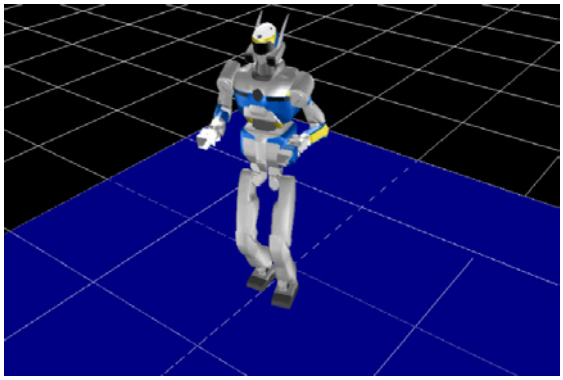


(a)t=12 [s]

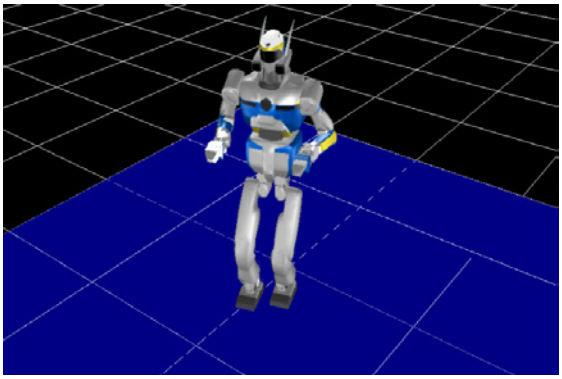
(b)t=12.67 [s]



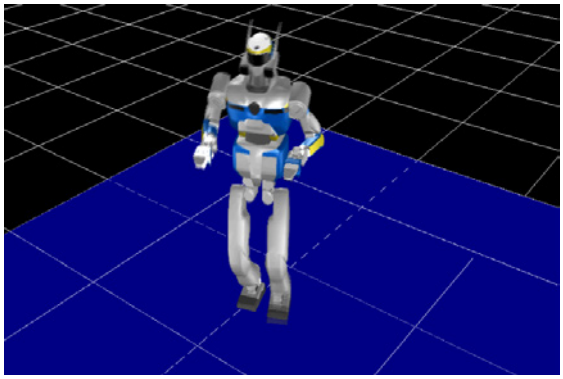
(c)t=13.33 [s]



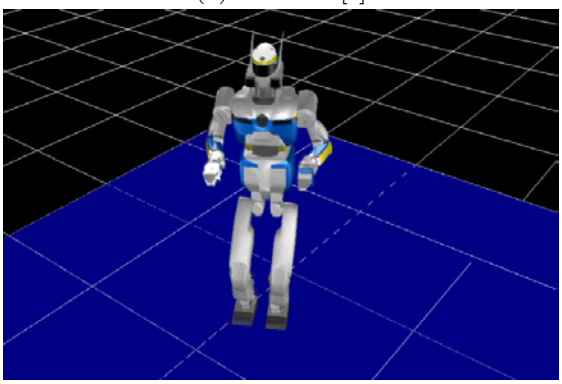
(d)t=14.00 [s]



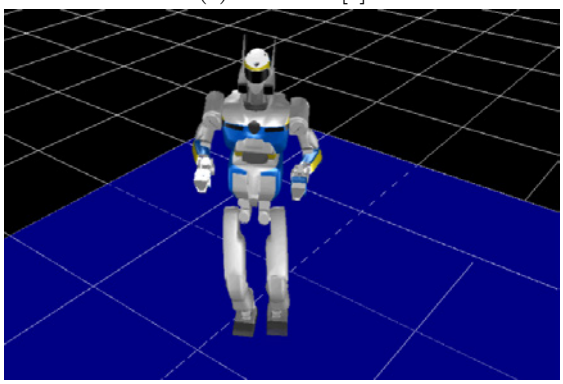
(e)t=14.67 [s]



(f)t=15.33 [s]



(g)t=16.00 [s]



(h)t=16.67 [s]

Figure 3.15: Simulation of walking with turn.

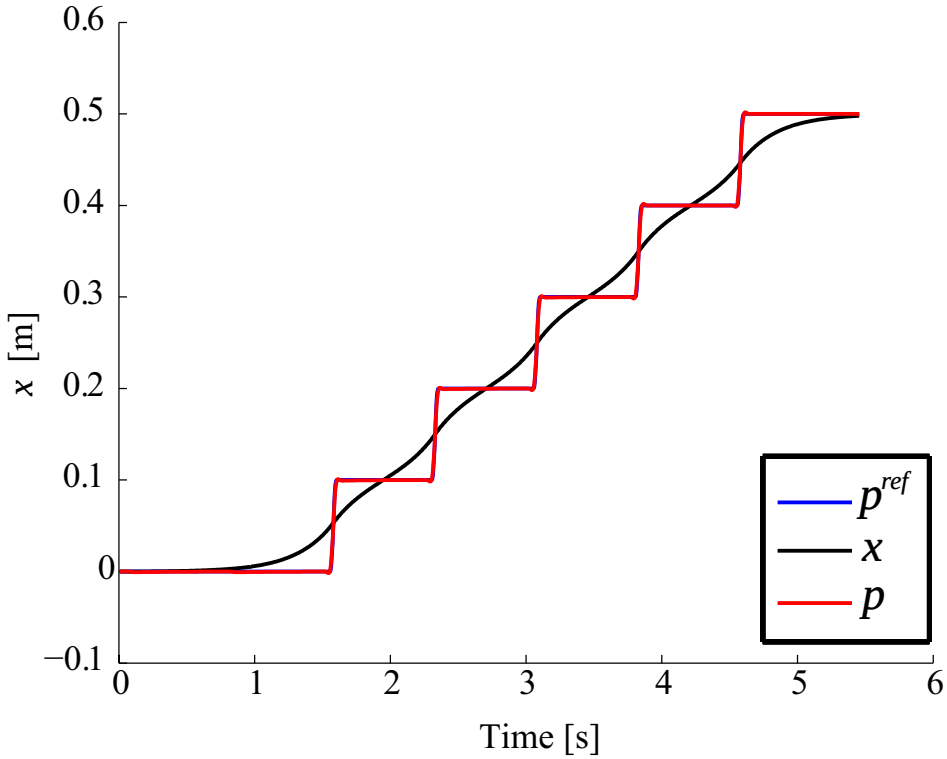


Figure 3.16: x direction data (without offset)

3.5.3 CoM Trajectory Change during Single Support Phase

In order to achieve quick response to the change of input commands, changing the trajectory of CoM during single support phase is desirable. A method proposed by Nishiwaki et al. [42] is adopted. CoM trajectory calculated by applying preview control of ZMP theory depends on:

1. the present position, velocity, acceleration of CoM,
2. reference ZMP given from now to next 1.5 [s].

According to this feature, CoM trajectory can be changed by offsetting the given reference ZMP at arbitrary timing. In order to keep imaginary ZMP [43] inside the footprint of support leg in single support phase, landing foot placement, that is pace should be changed with the same offset of reference ZMP.

For example, Figure 3.16 shows trajectories of reference ZMP, CoM and ZMP in x direction. The ZMP almost follows the reference ZMP when there is no offset in reference ZMP. Figure 3.17 shows a result of offsetting reference ZMP at 2.6 [s] with 0.05 [m]. That is, the pace is enlarged with 0.05 [m]. Here, 2.6 [s] is the moment when the trajectory change is commanded. But the offset begins from the next starting point of single support phase. ZMP leaves form reference ZMP since 2.6 [s] and moves backward due

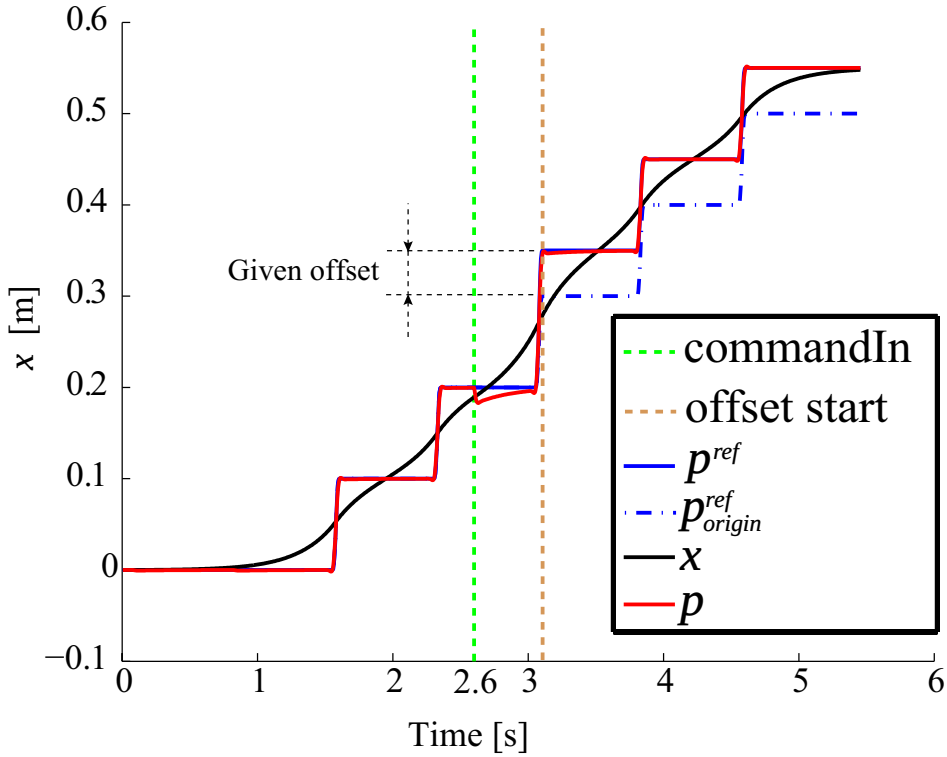


Figure 3.17: x direction data (with offset)

to the sudden acceleration, then converge to reference ZMP again. However, excessive displacement of ZMP is not desirable, and it would result in falling down. According to (3.17), ZMP at discrete time $k+1$ can be calculated as:

$$p_{k+1} = \boldsymbol{c}\mathbf{A}\boldsymbol{x}_k + \boldsymbol{cb}u_k \ . \tag{3.27}$$

(3.21) is rewritten as:

$$u_k = -\mathbf{K}\begin{bmatrix} p_k^{ref} - p_k \\ \Delta\boldsymbol{x}_k \end{bmatrix} + \sum_{j=1}^{N} f_j \Delta p_{k+j}^{ref} + u_{k-1} \ . \tag{3.28}$$

Substituting (3.28) into (3.27), following equation is obtained:

$$p_{k+1} = \boldsymbol{c}\mathbf{A}\boldsymbol{x}_k + \boldsymbol{cb}u_{k-1} - \boldsymbol{cbK}\begin{bmatrix} p_k^{ref} - p_k \\ \Delta\boldsymbol{x}_k \end{bmatrix} + \boldsymbol{cb}\sum_{j=1}^{N} f_j \Delta p_{k+j}^{ref} \ . \tag{3.29}$$

The change at discrete time $k+s$ influence p_{k+1} is given by:

$$\delta p_{k+1} = \boldsymbol{cb} f_s \delta \Delta p_{k+s}^{ref} \ . \tag{3.30}$$

According to (3.30) and Figure 3.2, the displacement of ZMP increases as trajectory change command is delayed. Simulation with the same condition of Figure 3.17 except

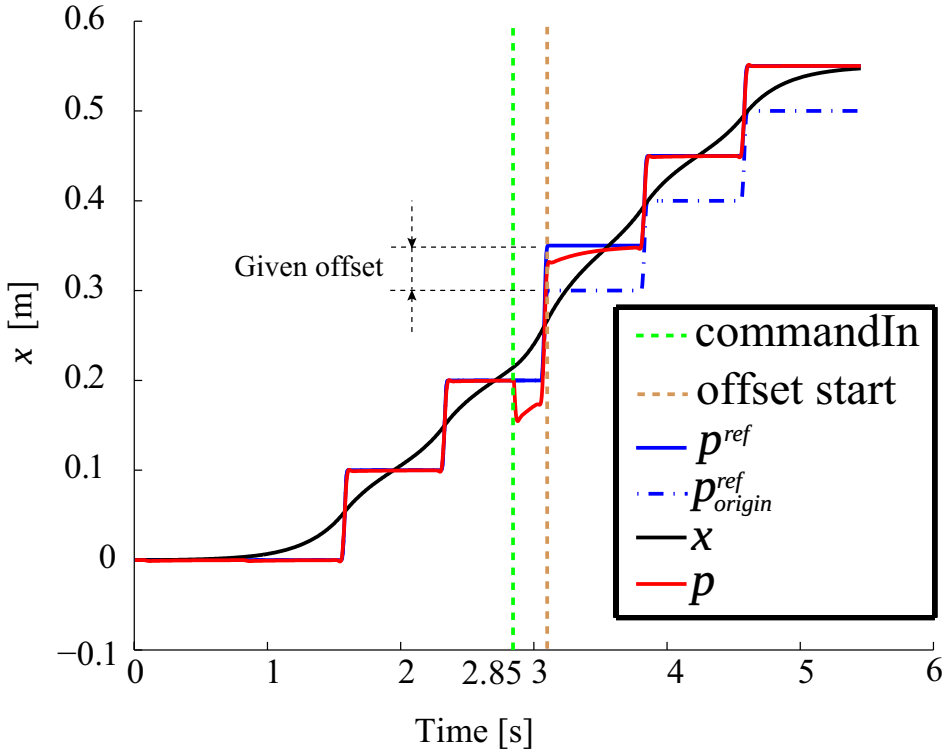


Figure 3.18: x direction data (a late command input)

input timing is shown in Figure 3.18. Therefore, there is a possibility that ZMP moves to the fringe of the support polygon due to a delayed input. In order to avoid a large displacement of ZMP , a threshold is set for restraining the displacement of ZMP. When the maximum displacement of ZMP is under the threshold, robots transit to new trajectory by given command. And when the maximum displacement of ZMP goes beyond the threshold, a new offset value that meets the threshold is computed as:

$$\text{offset}_{new} = \text{offset}_{origin} \times \frac{\delta p_{Threshold}}{\delta p} \ . \tag{3.31}$$

Also, foot placement is adjusted with the same displacement of reference ZMP as:

$$\text{pace}_{new} = \text{pace}_{origin} + \text{offset}_{new} \ . \tag{3.32}$$

3.6 Summary

In this chapter, the online walking method used in this research is introduced, which includes reference zmp planning, reference CoM planning, and swing leg trajectory planning. A method to change its trajectory during stepping motion without large zmp deviance is also proposed.

Chapter 4. Leader Follower Type Cooperation Control

4.1 Introduction

In this chapter, a leader-follower type cooperative object transportation method for two humanoid robots is proposed. In leader-follower type cooperative control, the leader is directly controlled by an operator, i. e. the walking parameter defined in chapter 3 is given by an operator online. The follower robot stretches/shrinks its hand based on impedance control, then reference walking velocity is generated proportional to the current hand position.

4.2 Position-based impedance control of hands

Since a leader-follower type cooperation is used in this research, the cooperative movement of robots is based on the relative displacement of their connecting portion, i.e. the position of hands. When there is a relative displacement, interactive force occurs between the leader and the follower robots. Therefore, position-based impedance control is used in order to ease the interactive force. The equation of an impedance is given as:

$$m\Delta\ddot{x} + d\Delta\dot{x} + k\Delta x = F \ , \tag{4.1}$$

where, m is a virtual mass of hand, d is a damping coefficient, k is a spring constant, Δx is the displacement from reference value, and F is the force measured from force sensor in the wrists of each robot. Then, the system of (4.1) is discretized with sampling time of ΔT as:

$$\Delta x_{i+1} = \frac{\Delta T^2}{m} F_i + \frac{2m - k\Delta T^2 - d\Delta T}{m} \Delta x_i + \frac{-m + d\Delta T}{m} \Delta x_{i-1} \ . \tag{4.2}$$

When the spring constant is set to zero, it will become damping control. With input F, the position of hand can be calculate as a reference.

4.2.1 Walking command generation with hand position

As shown in Figure 4.1, the hand position of the leader robot is controlled in the reference position by setting a spring constant in the impedance control of hand position. However, the spring constant of follower robot is set to zero. Therefore, follower robot

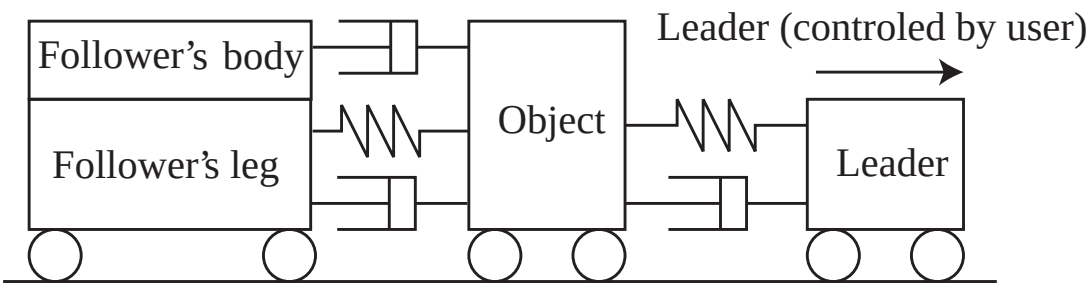
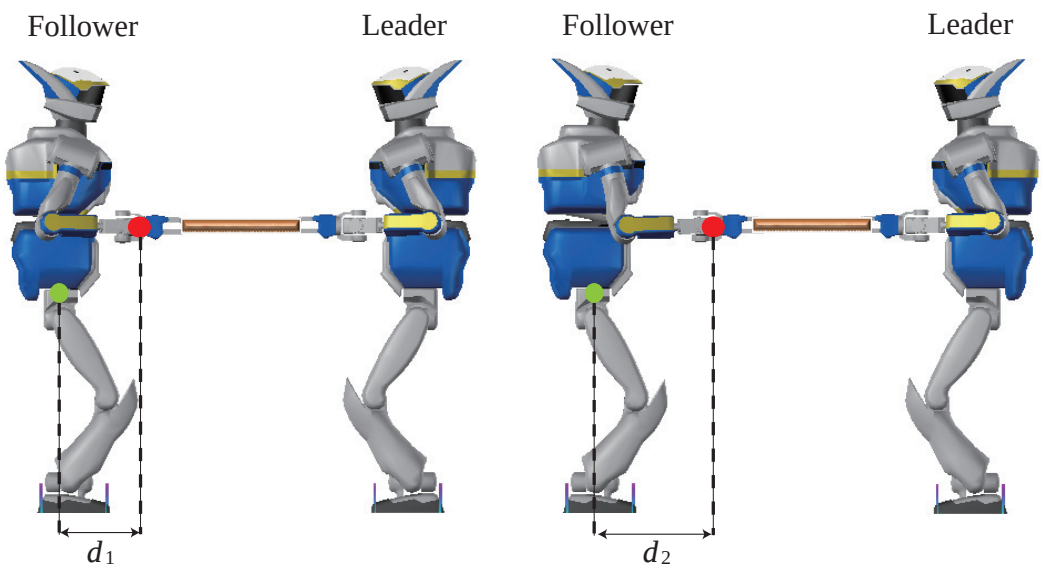


Figure 4.1: A concept of cooperative transfer system



(a) Initial position (b) After the leader robot moved

Figure 4.2: The principle of the framework

recover its hand position to the reference position by walking. The walking command generation corresponds to the spring and damper between the object and follower's leg, and the impedance control of hands corresponds to the damper between the object and follower's body in Figure 4.1 respectively. As the leader robot starts to move, the hands of follower robot start to stretch due to the impedance control, then d_1 becomes d_2 (Figure 4.2). PD control is used to generate walking command that recovers d_2 back to d_1. The walking command is generated as:

$$\text{pace}_x = P(d_2 - d_1) + D(\dot{d}_2 - \dot{d}_1) \ , \tag{4.3}$$

where pace_x is a walking command that is the relative distance between landing placement and support foot, and P and D act as spring and damper in Figure 4.1. Besides, only x direction (back and forward) is verified in this research. Reference ZMP is com-

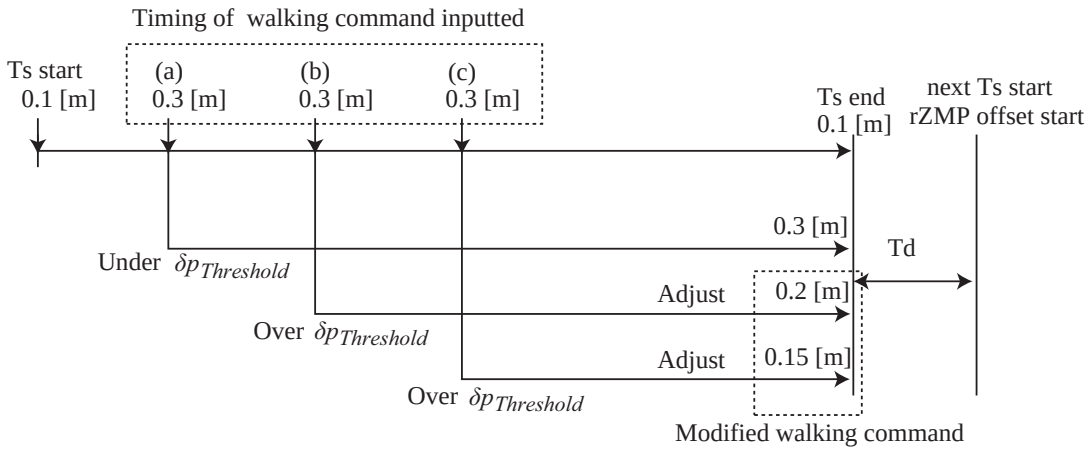


Figure 4.3: Same input in single support phase

puted with the walking command, then CoM trajectory is calculated by applying preview control theory. However, if the trajectory change method discussed in section 3.5.3 is applied without any extra processing, a problem as illustrated in Figure 4.3 occurs. Ts and Td in Figure 4.3 are period of single support phase and double support phase, respectively. The walking command pace_x in the beginning of single support phase is 0.1 [m] at first. A new walking command 0.3 [m] is inputted at timing (a), and the ZMP displacement from reference ZMP is assumed under the given threshold. Then, the new walking command is executed and the robot transits to new trajectory. As shown in Figure 3.17 and Figure 3.18, the ZMP displacement increases as trajectory change command is delayed. Assuming that the ZMP displacement goes beyond the threshold value since timing (b), then the walking command is adjusted downward in order to meet to the threshold. Here, the adjusted walking command is assumed to be 0.2 [m]. Similarly, the walking command is adjusted downward at timing (c). Because the ZMP displacement at timing (c) is larger than one at timing (b), the walking command 0.3 [m] is adjusted into a value smaller than one at timing (b) due to (3.30) and (3.32). As a result, the adjusted walking command becomes smaller and smaller. Finally, the pace degenerates even if a same command is inputted continuously. Therefore, in order to refer the walking command generated from (4.3) at every sampling time, it is necessary to add extra processing that avoids degeneration.

4.3 Walking Simulation of Walking Command Generated with Hand Position

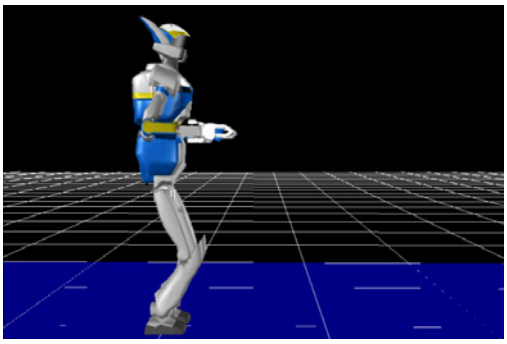
The walking generator is simulated with the method describe above, and the system is implemented on HRP-2. The procedure is shown as follows:

(I) Let the robot step right in the place.

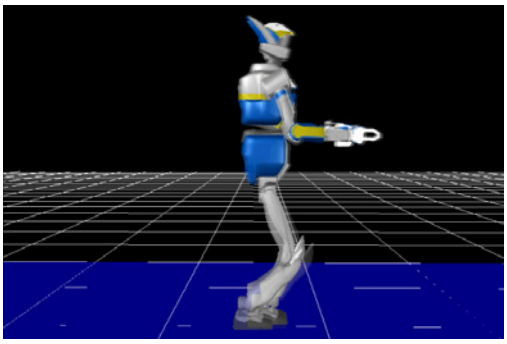
(II) Applying in a virtual force at the force sensor, then release.

(III) Applying in a virtual force in inverse direction, then release.

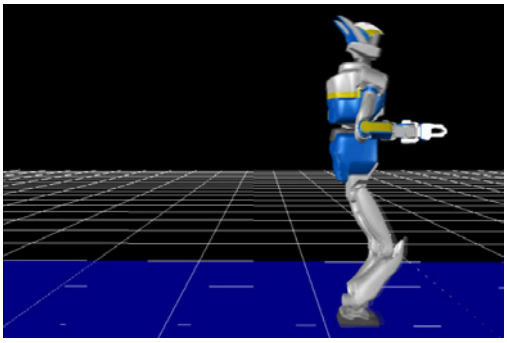
The hand position is limited within $-0.1 \sim 0.2$ [m] from the original position in order to avoid singular point. The walking command is restrained under 0.15 [m] in order to keep the height of CoM constant. The simulation result is shown in Figure 4.4. The robot generates walking command properly according to hand position. The hand position is plotted in Figure 4.5. Walking command with pace adjustment and without are shown in Figure 4.7 and Figure 4.6 respectively. The walking command without adjustment is proportional to hand position, and discontinuous levels are observed with adjustment (Figure 4.7). This discontinuous levels are due to the effect of walking command adjustment. Figure 4.8 shows a zone up of the first discontinuous level. At first, walking command with adjustment rises as robot stretches its hands. Because the ZMP displacement goes beyond the threshold from the timing circled in Figure 4.8, the walking command with adjustment is kept constant in order to avoid degeneration of pace. After that, the walking command is executed without change till the end of executing step even walking command keeps changing, and this is the reason of discontinuous level.



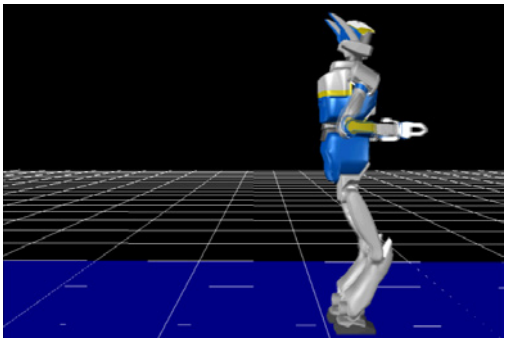
(a) 25 [s]



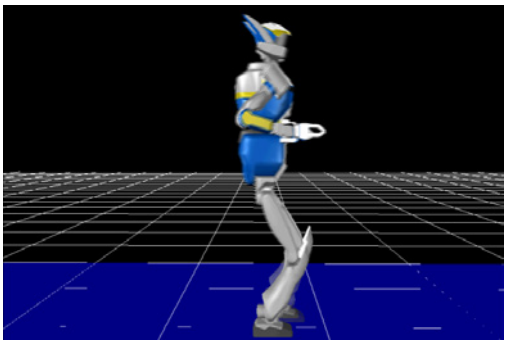
(b) 29 [s]



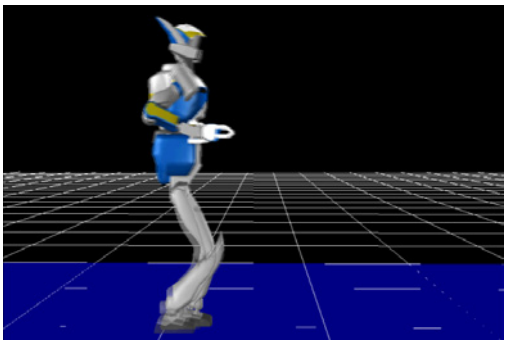
(c) 33 [s]



(d) 42 [s]



(e) 46 [s]



(f) 50 [s]

Figure 4.4: Walking simulation

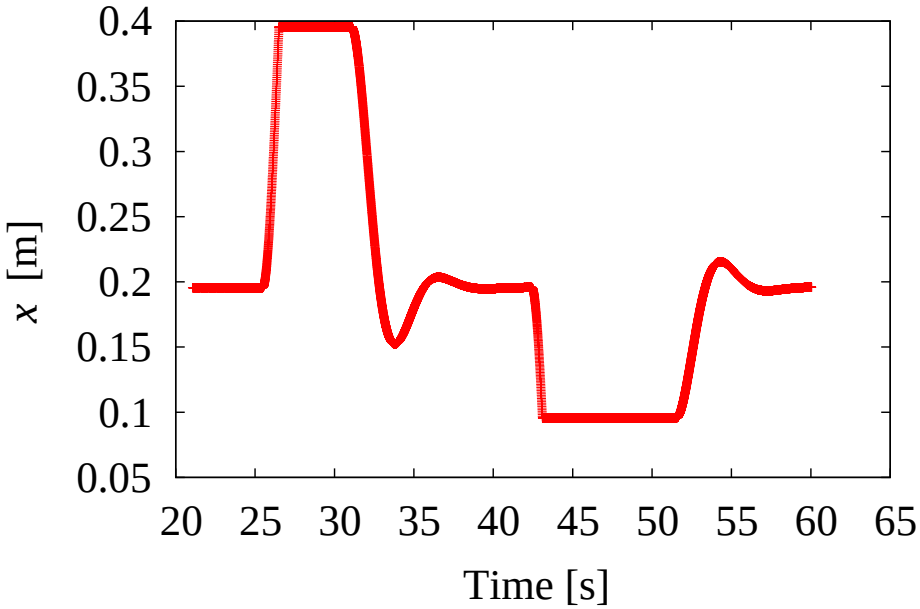


Figure 4.5: Hand position in waist coordinates

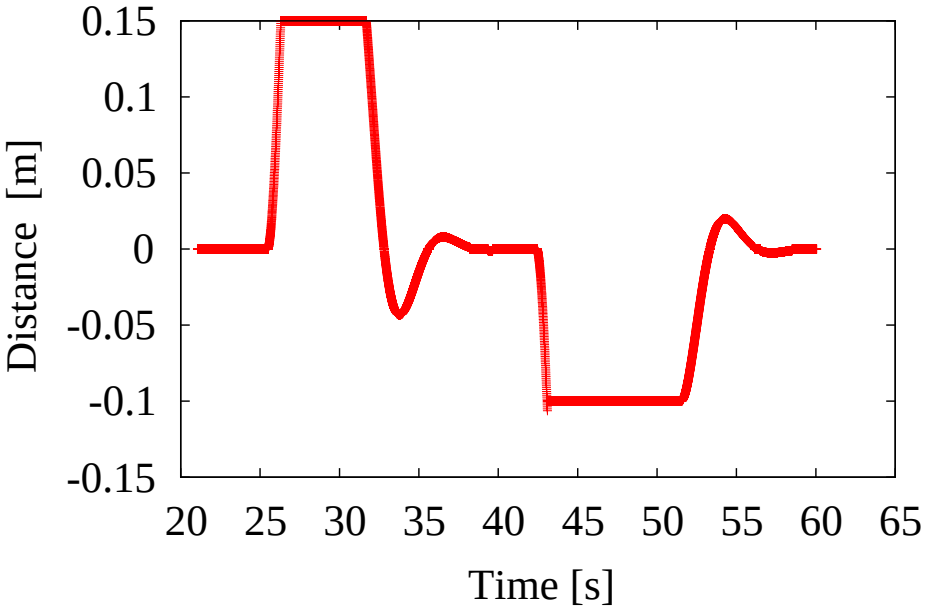


Figure 4.6: Walking command without adjustment

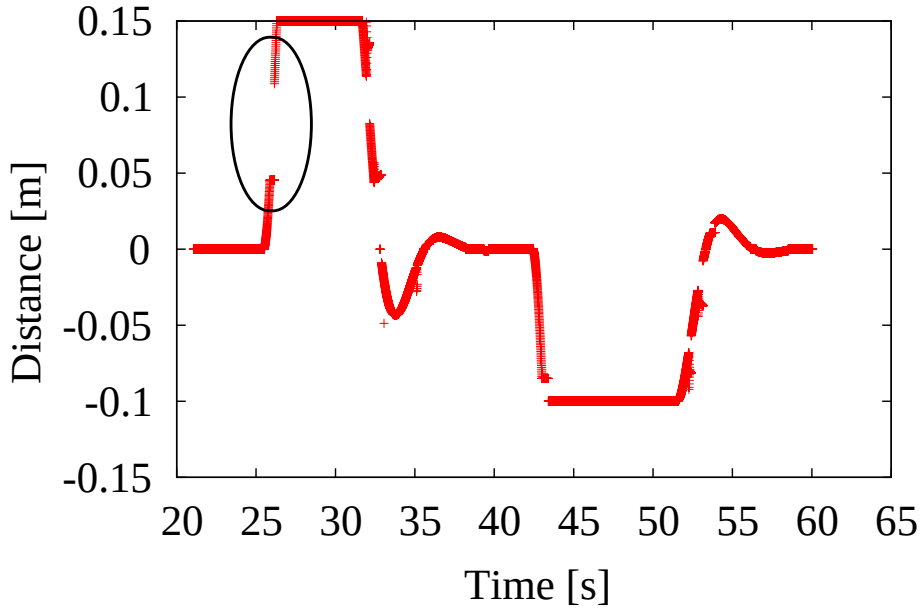


Figure 4.7: Walking command with adjustment

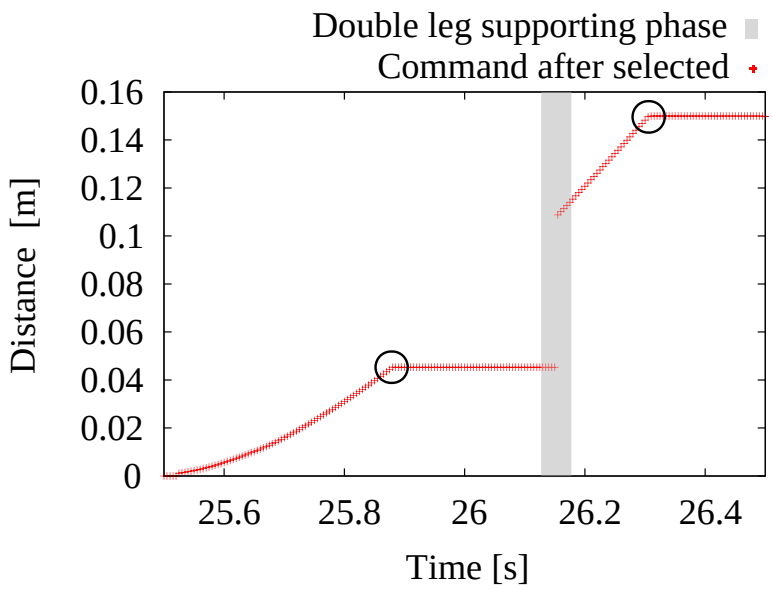


Figure 4.8: A scale-up of Figure 4.7

4.4 Experiment of Leader-Follower Type Cooperative Object Transportation

Since the interactive force between the two robots results from relative displacement of their connecting portion, the position of hands, it is effective to ease the force by eliminate the relative displacement. There are two ways to eliminate the relative displacement: change the position of robots or change the position of their hands. In order to keep hand position from entering singular point, the hand of follower robot is restrained within 0.194∼0.396 [m], and the hand of leader robot is restrained between 0.096∼0.396 [m]. Also, the right and left hands of robots move coordinately in x direction, and keep constant in y and z direction. The procedure of this experiment is shown as follows:

(I) Let the robots step right in the place in order of follower, leader robot.

(II) Let the leader robot walk backward, then step right in the place for a while.

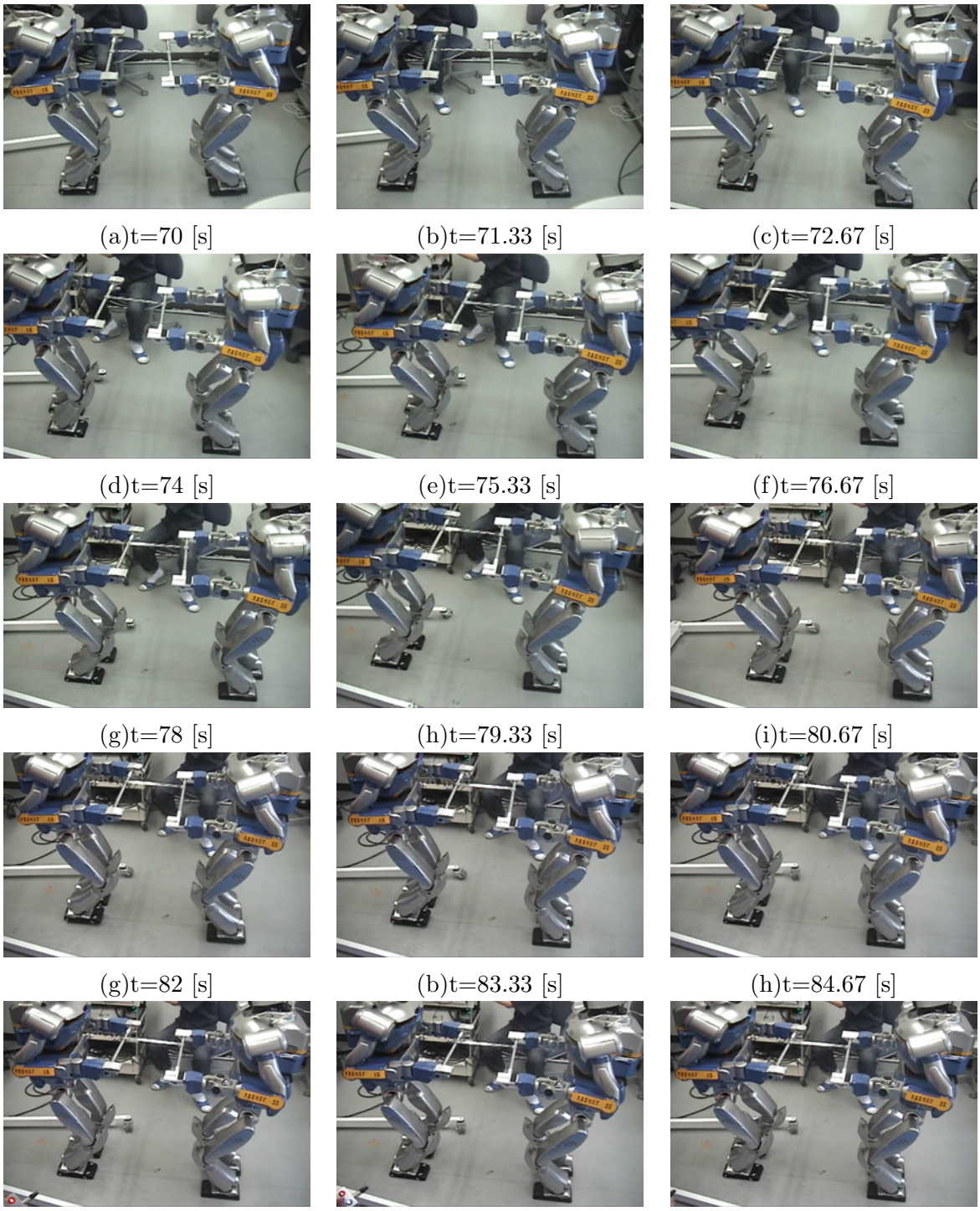
(III) Let the leader robot walk forward, then step right in the place for a while.

(IV) Stop the robots in order of leader, follower robot.

The experiment is shown in Figure 4.9 Figure 4.10, and the trend of hand position are shown in Figure 4.11 (a)(b). The hand position of follower is displaced from reference position at first in order to grip object with leader robot. Although there is a oscillation of hand position, both the robots move without falling down, and the follower robot follows leader robot properly before its hand reach its limit position. The oscillation cycle is about 3 [s] with maximum amplitude about 0.02 [m]. Additionally, the oscillation neither diverge nor converge. Whether the oscillation is a result of easing the interactive force, or a result of improper control, further study is necessary. Also, the hand position of leader robot always reaches limit position even the spring parameter does not equal zero while stepping. An insufficient k gain is considered as a reason. Finally, except some of the peaks, the interactive force is restrained under about 7 [N].

4.5 Summary

In this chapter, a framework of cooperative object transportation bases on impedance control is proposed, and cooperative object transportation by two humanoid robots in x direction is achieved.



(a)t=70 [s] (b)t=71.33 [s] (c)t=72.67 [s]

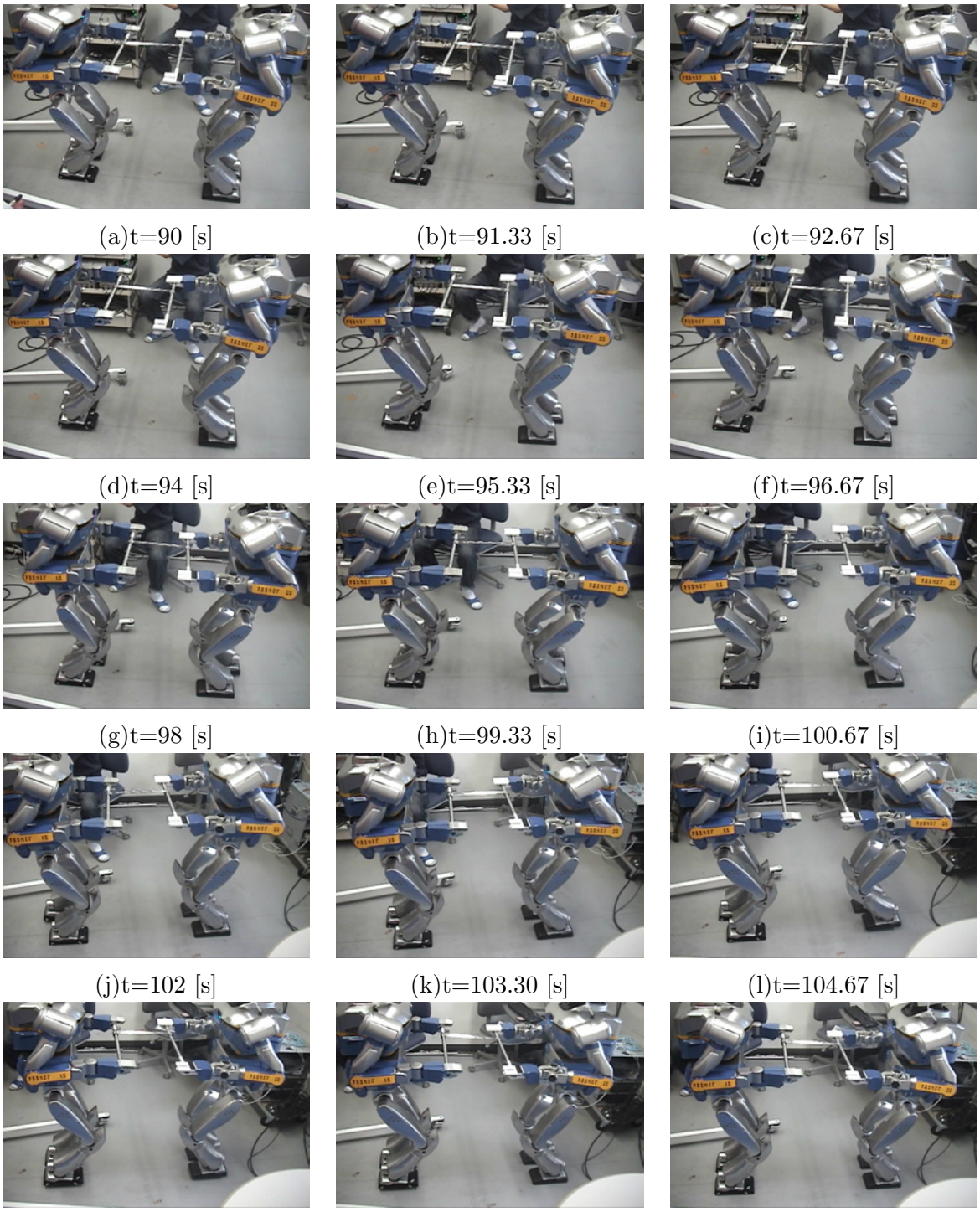
(d)t=74 [s] (e)t=75.33 [s] (f)t=76.67 [s]

(g)t=78 [s] (h)t=79.33 [s] (i)t=80.67 [s]

(g)t=82 [s] (b)t=83.33 [s] (h)t=84.67 [s]

(j)t=86 [s] (k)t=87.33 [s] (l)t=88,67 [s]

Figure 4.9: Cooperative move by two humanoid robots(70 [s]∼88.67 [s]).



(a)t=90 [s] (b)t=91.33 [s] (c)t=92.67 [s]

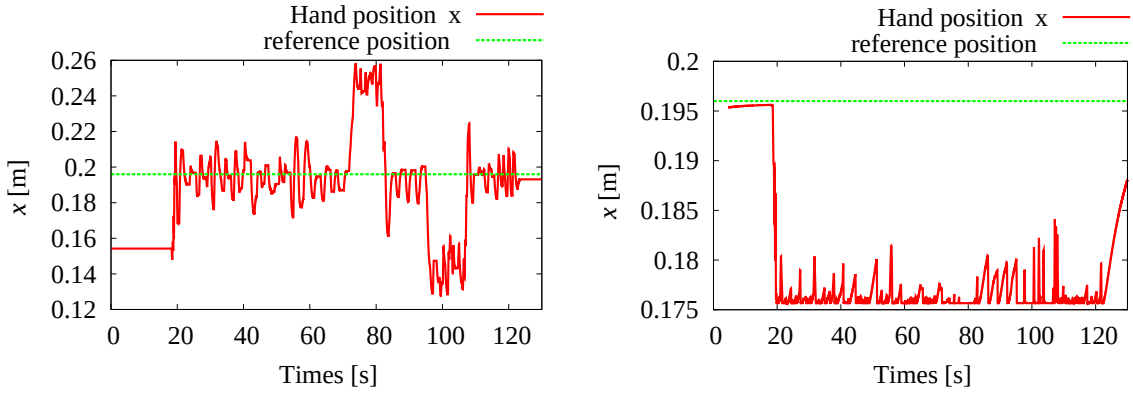
(d)t=94 [s] (e)t=95.33 [s] (f)t=96.67 [s]

(g)t=98 [s] (h)t=99.33 [s] (i)t=100.67 [s]

(j)t=102 [s] (k)t=103.30 [s] (l)t=104.67 [s]

(m)t=106 [s] (n)t=107.33 [s] (o)t=108.67 [s]

Figure 4.10: Cooperative move by two humanoid robots(90 [s]∼108.67 [s]).



(a) Position of follower's hand.

(b) Position of leader's hand.

Figure 4.11: Hand position of leader and follower robot.

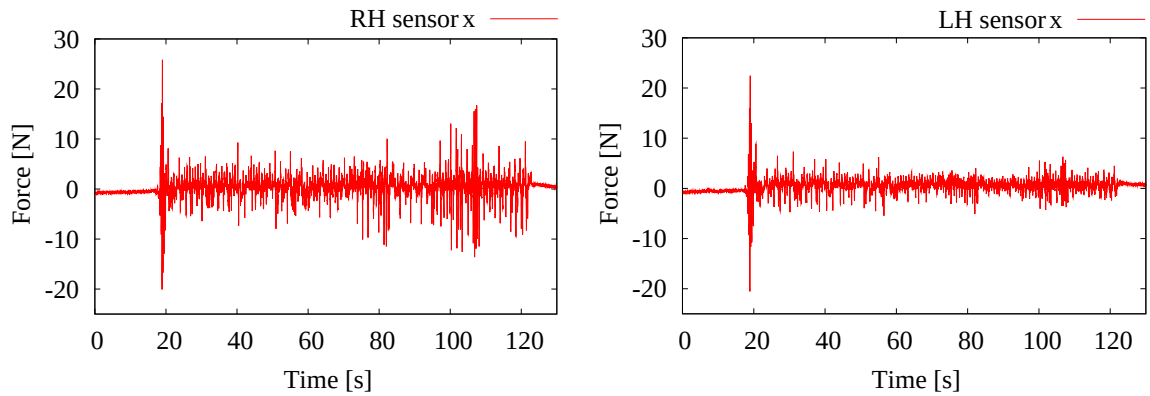


Figure 4.12: Measured Force at the wrist force sensor of follower robot

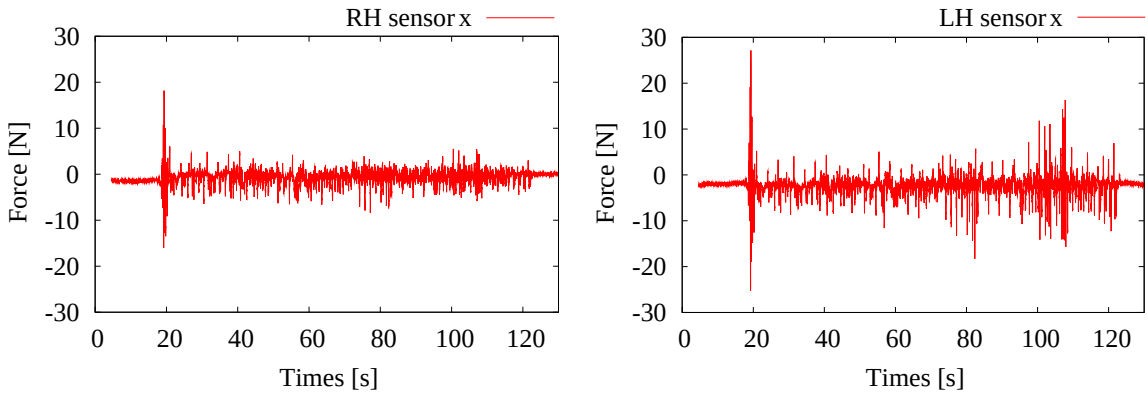


Figure 4.13: Measured Force at the wrist force sensor of leader robot

Chapter 5. Symmetry Type Cooperation Control

5.1 Introduction

In the leader-follower type, the movements of follower robots are generated base on the force controller, which calculates the movement of the leader robot by presumption, and decides the moving velocity. However, the moving velocity may not be the same as the leader robot's velocity. Moreover, the follower robots start planning after the leader robot moves as illustrated in Figure. 5.1 (a), and this time-lag may result in a low responsibility and an unexpected tilting. In symmetry type, the robots synchronously move as illustrated in Figure. 5.1 (b). This synchronous movement achieves high responsibility and safety in carrying an object. Therefore, a symmetry type cooperation for arbitrary number of humanoid robots to transport an object is proposed.

5.2 Workspace Vectors Describing Coordinated Tasks of n Robotic Arms System

5.2.1 External Force and Internal Force

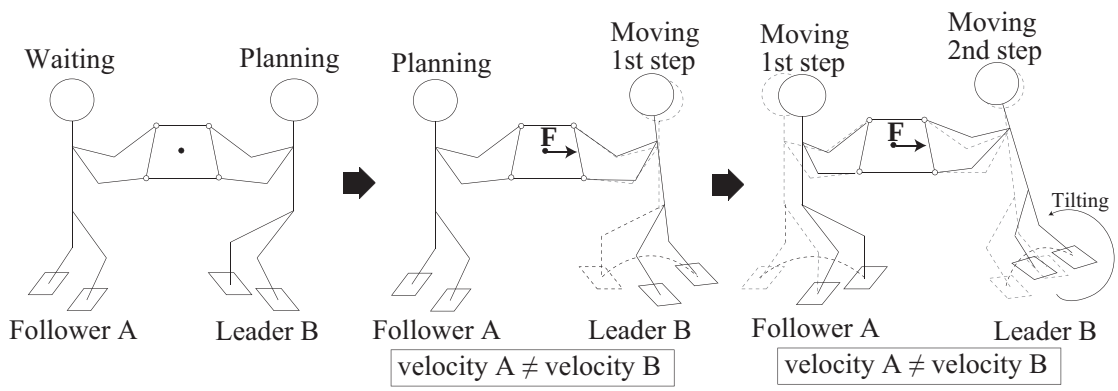
In order to define external and internal forces and moments in a multi-robot cooperation system, the concept of a virtual stick [21] is used. Let us consider n robotic arms that hold an object, as illustrated in Figure 5.2. Here, $\Sigma_{\mathrm{h}i}$ is a coordinate frame fixed to the hand i ($i = 1, \ \ldots, n$), Σ_{a} is a coordinate frame fixed to the object, Σ_{o} is the world coordinate frame, and $\mathrm{O}_{\mathrm{h}i}$, O_{a}, and O_{o} are their origins, respectively.

Origin O_{a} is a specific point fixed to the object. The position and orientation of the object are defined at O_{a}. Although O_{a} can be set anywhere, it is reasonable to set it at the center of mass of the object. The virtual sticks are defined by vectors ${}^{\mathrm{o}}\boldsymbol{l}_{\mathrm{h}i}$ from $\mathrm{O}_{\mathrm{h}i}$ to O_{a}, with respect to Σ_{o}. The vectors ${}^{\mathrm{o}}\boldsymbol{l}_{\mathrm{h}i}$ are determined when n robot arms grasp the object. As illustrated in Figure 5.2(b), Σ_{bi} is a coordinate frame fixed to the tip of virtual stick i. Initially, Σ_{bi} coincides with Σ_{a}, however, if the object deforms, they may no longer coincide.

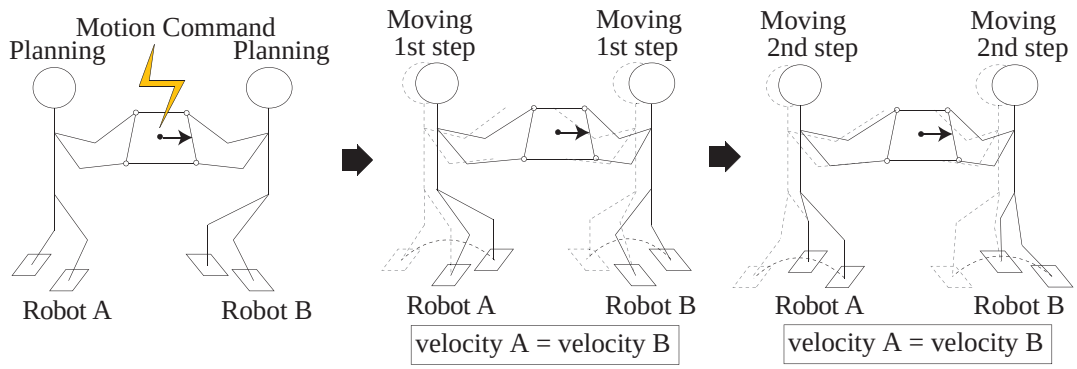
The force vector ${}^{\mathrm{o}}\boldsymbol{f}_{\mathrm{b}i}$ generated at the tip of virtual stick i is defined as

$$ {}^{\mathrm{o}}\boldsymbol{f}_{\mathrm{b}i} \equiv \begin{bmatrix} {}^{\mathrm{o}}\boldsymbol{F}_{\mathrm{b}i}^{\mathrm{T}} & {}^{\mathrm{o}}\boldsymbol{N}_{\mathrm{b}i}^{\mathrm{T}} \end{bmatrix}^{\mathrm{T}} , \tag{5.1} $$

where ${}^{\mathrm{o}}\boldsymbol{F}_{\mathrm{b}i}$ and ${}^{\mathrm{o}}\boldsymbol{N}_{\mathrm{b}i}$ are the force and moment exerted at the tip of virtual stick i.



(a) Leader-follower type.



(b) Symmetry type.

Figure 5.1: Conceptual difference of the two types of cooperation.

Suffix o indicates that the vector is defined with respect to world coordinate frame Σ_{o}. Vector ${}^{\mathrm{o}}\boldsymbol{f}_{\mathrm{b}i}$ is calculated from the force and moment applied to the object by hand i (see Appendix A). The external forces and moments applied to the object, ${}^{\mathrm{o}}\boldsymbol{F}_{\mathrm{a}}$ and ${}^{\mathrm{o}}\boldsymbol{N}_{\mathrm{a}}$, respectively, are given by the summation of ${}^{\mathrm{o}}\boldsymbol{f}_{\mathrm{b}i}$ as

$$ {}^{\mathrm{o}}\boldsymbol{f}_{\mathrm{a}} \equiv \begin{bmatrix} {}^{\mathrm{o}}\boldsymbol{F}_{\mathrm{a}}^{\mathrm{T}} & {}^{\mathrm{o}}\boldsymbol{N}_{\mathrm{a}}^{\mathrm{T}} \end{bmatrix}^{\mathrm{T}} = \mathbf{W}\ {}^{\mathrm{o}}\boldsymbol{q}_{\mathrm{b}} \ , \tag{5.2} $$

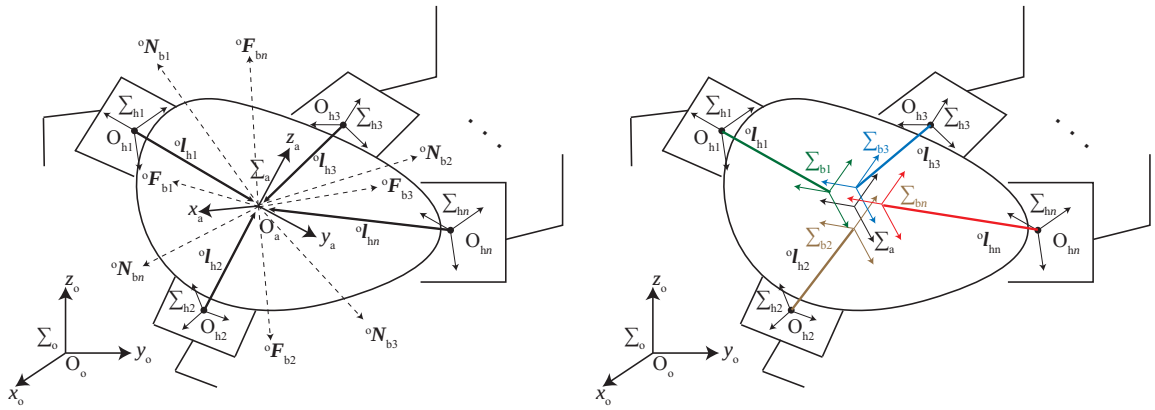
where

$$ \mathbf{W} \equiv \begin{bmatrix} \mathbf{I}_6 & \mathbf{I}_6 & \mathbf{I}_6 & \dots \end{bmatrix} \ , \mathbf{W} \in \mathbb{R}^{6\times 6n} \ , \tag{5.3} $$

$$ {}^{\mathrm{o}}\boldsymbol{q}_{\mathrm{b}} \equiv \begin{bmatrix} {}^{\mathrm{o}}\boldsymbol{f}_{\mathrm{b}1}^{\mathrm{T}} & {}^{\mathrm{o}}\boldsymbol{f}_{\mathrm{b}2}^{\mathrm{T}} & {}^{\mathrm{o}}\boldsymbol{f}_{\mathrm{b}3}^{\mathrm{T}} & \dots & {}^{\mathrm{o}}\boldsymbol{f}_{\mathrm{b}i}^{\mathrm{T}} \end{bmatrix}^{\mathrm{T}} \ . $$

Here, $\mathbf{I}_n$ is an $n \times n$ identity matrix. Because matrix $\mathbf{W}$ maps a $6n$-dimensional vector to a 6-dimensional vector, the rank is 6. Hence, the range of $\mathbf{W}$ is 6-dimensional, and the range of its null space is $(6n-6)$-dimensional. The null space of $\mathbf{W}$ is the set of all column vectors $\boldsymbol{v}$ that satisfy $\mathbf{W}\boldsymbol{v} = \mathbf{0}$. By choosing appropriate $(6n-6)$ independent vectors from this set, a null space basis of $\mathbf{W}$, $\mathbf{V}$, can be defined. Where $\mathbf{V}$ is a $6n \times (6n-6)$ matrix that satisfies

$$ \mathbf{W}\mathbf{V} = \mathbf{0}_{6\times(6n-6)} \ . \tag{5.4} $$



(a) Force and moment.

(b) Position and orientation.

Figure 5.2: An object is hold by n robotic arms.

The general solution of (5.2) is given by

$$ {}^{\mathrm{o}}\boldsymbol{q}_{\mathrm{b}} = \mathbf{W}^{-}\,{}^{\mathrm{o}}\boldsymbol{f}_{\mathrm{a}} + \mathbf{V}\,{}^{\mathrm{o}}\boldsymbol{f}_{\mathrm{m}} \ , \tag{5.5} $$

where ${}^{\mathrm{o}}\boldsymbol{f}_{\mathrm{m}}$ is an arbitrary $(6n-6)$-dimensional vector that corresponds to $\mathbf{V}$, and $\mathbf{W}^{-}$ is a generalized inverse matrix of $\mathbf{W}$ that satisfies

$$ \mathbf{W}\mathbf{W}^{-}\mathbf{W} = \mathbf{W} \ . \tag{5.6} $$

Note that $\mathbf{V}\,{}^{\mathrm{o}}\boldsymbol{f}_{\mathrm{m}}$ belongs to the null space of $\mathbf{W}$, and hence ${}^{\mathrm{o}}\boldsymbol{f}_{\mathrm{m}}$ does not affect the external force. Therefore, ${}^{\mathrm{o}}\boldsymbol{f}_{\mathrm{m}}$ corresponds to internal forces/moments. Further, note that $\mathbf{V}$ and ${}^{\mathrm{o}}\boldsymbol{f}_{\mathrm{m}}$ are not uniquely determined.

Equation (5.5) can be rewritten as

$$ {}^{\mathrm{o}}\boldsymbol{q}_{\mathrm{b}} = \begin{bmatrix} \mathbf{W}^{-} & \mathbf{V} \end{bmatrix} \begin{bmatrix} {}^{\mathrm{o}}\boldsymbol{f}_{\mathrm{a}} \\ {}^{\mathrm{o}}\boldsymbol{f}_{\mathrm{m}} \end{bmatrix} = \mathbf{U}\,{}^{\mathrm{o}}\boldsymbol{h} \ , \tag{5.7} $$

$$ \mathbf{U} \equiv \begin{bmatrix} \mathbf{W}^{-} & \mathbf{V} \end{bmatrix} \in \mathbb{R}^{6n\times 6n}, \ {}^{\mathrm{o}}\boldsymbol{h} \equiv \begin{bmatrix} {}^{\mathrm{o}}\boldsymbol{f}_{\mathrm{a}} \\ {}^{\mathrm{o}}\boldsymbol{f}_{\mathrm{m}} \end{bmatrix} \in \mathbb{R}^{6n}. $$

The force/moment vector ${}^{\mathrm{o}}\boldsymbol{h}$ is defined as a generalized force. The internal force ${}^{\mathrm{o}}\boldsymbol{f}_{\mathrm{m}}$ can be represented as $(n-1)$ sets of 6-dimensional force vectors as

$$ {}^{\mathrm{o}}\boldsymbol{f}_{\mathrm{m}} \equiv \begin{bmatrix} {}^{\mathrm{o}}\boldsymbol{f}_{\mathrm{r1}} \\ {}^{\mathrm{o}}\boldsymbol{f}_{\mathrm{r2}} \\ \vdots \\ {}^{\mathrm{o}}\boldsymbol{f}_{\mathrm{r},n-1} \end{bmatrix} \ . \tag{5.8} $$

The force/moment vector ${}^{\mathrm{o}}\boldsymbol{h}$ for a given ${}^{\mathrm{o}}\boldsymbol{q}_{\mathrm{b}}$ is obtained by solving (5.7) as

$$ {}^{\mathrm{o}}\boldsymbol{h} = \mathbf{U}^{-1}\,{}^{o}\boldsymbol{q}_{\mathrm{b}} \ . \tag{5.9} $$

Equation (5.9) can be expanded as

$$\begin{aligned}
{}^{\mathrm{o}}\boldsymbol{f}_{\mathrm{a}} &= {}^{\mathrm{o}}\boldsymbol{f}_{\mathrm{b1}} + {}^{\mathrm{o}}\boldsymbol{f}_{\mathrm{b2}} + \ \ldots \ + {}^{\mathrm{o}}\boldsymbol{f}_{\mathrm{b}n} \ , \\
{}^{\mathrm{o}}\boldsymbol{f}_{\mathrm{r1}} &= c_{1,1}{}^{\mathrm{o}}\boldsymbol{f}_{\mathrm{b1}} + c_{1,2}{}^{\mathrm{o}}\boldsymbol{f}_{\mathrm{b2}} + \ \ldots \ + c_{1,n}{}^{\mathrm{o}}\boldsymbol{f}_{\mathrm{b}n} \ , \\
{}^{\mathrm{o}}\boldsymbol{f}_{\mathrm{r2}} &= c_{2,1}{}^{\mathrm{o}}\boldsymbol{f}_{\mathrm{b1}} + c_{2,2}{}^{\mathrm{o}}\boldsymbol{f}_{\mathrm{b2}} + \ \ldots \ + c_{2,n}{}^{\mathrm{o}}\boldsymbol{f}_{\mathrm{b}n} \ , \\
&\vdots \\
{}^{\mathrm{o}}\boldsymbol{f}_{\mathrm{r},n-1} &= c_{n-1,1}{}^{\mathrm{o}}\boldsymbol{f}_{\mathrm{b1}} + c_{n-1,2}{}^{\mathrm{o}}\boldsymbol{f}_{\mathrm{b2}} + \ \ldots \ + c_{n-1,n}{}^{\mathrm{o}}\boldsymbol{f}_{\mathrm{b}n} \ .
\end{aligned} \tag{5.10}$$

Furthermore, (5.9) can be rewritten as

$${}^{\mathrm{o}}\boldsymbol{h} = \begin{bmatrix} {}^{\mathrm{o}}\boldsymbol{f}_{\mathrm{a}} \\ {}^{\mathrm{o}}\boldsymbol{f}_{\mathrm{m}} \end{bmatrix} = \begin{bmatrix} \mathbf{W} \\ \mathbf{C} \end{bmatrix} {}^{\mathrm{o}}\boldsymbol{q}_{\mathrm{b}} = \mathbf{U}^{-1}\ {}^{o}\boldsymbol{q}_{\mathrm{b}} \ , \tag{5.11}$$

where

$$\mathbf{C} \equiv \begin{bmatrix} c_{1,1}\mathbf{I}_6 & \ldots & c_{1,n}\mathbf{I}_6 \\ \vdots & \ddots & \vdots \\ c_{n-1,1}\mathbf{I}_6 & \ldots & c_{n-1,n}\mathbf{I}_6 \end{bmatrix} .$$

The $6n \times (6n-6)$ matrix $\mathbf{V}$ in (5.5) and the $(6n-6) \times 6n$ matrix $\mathbf{C}$ in (5.11) have the following relationship.

$$\mathbf{U} = \begin{bmatrix} \mathbf{W}^{-} & \mathbf{V} \end{bmatrix} = \begin{bmatrix} \mathbf{W} \\ \mathbf{C} \end{bmatrix}^{-1} . \tag{5.12}$$

Because n robot arms can apply $6n$-dimensional forces/moments, external force ${}^{\mathrm{o}}\boldsymbol{f}_{\mathrm{a}}$ and internal forces ${}^{\mathrm{o}}\boldsymbol{f}_{\mathrm{r}i}$ $(i = 1, \ \ldots, \ n-1)$ can be independently controlled.

The physical meanings of the matrices in this section are summarized as follows. Matrix $\mathbf{W}$ maps the force applied by multiple robot arms onto the external force that is applied to the holding object. Matrix $\mathbf{V}$ is a null space basis of $\mathbf{W}$. Matrix $\mathbf{U}$ maps a set of external and internal forces onto the forces that the robot arms should apply to the object. Matrix $\mathbf{C}$ maps the forces applied by the robot arms onto the internal forces.

5.2.2 Determination of Internal Force of n Robotic Arms System

In [21], $\mathbf{V}$ was first given, and by using the pseudo inverse matrix $\mathbf{W}^{\dagger}$ for $\mathbf{W}^{-}$, $\mathbf{C}$ was automatically determined by inverting $\begin{bmatrix} \mathbf{W}^{\dagger} & \mathbf{V} \end{bmatrix}$. However, when more than two multiple robotic arms are cooperating, it is difficult to determine $\mathbf{V}$, because $\mathbf{V}$ does not present an intuitive meaning for internal forces/moments. As shown in (5.10), the intuitive meaning for internal forces/moments is given by $\mathbf{C}$.

Therefore, in this paper, internal forces/moments ${}^{\mathrm{o}}\boldsymbol{f}_{\mathrm{r1}} \ \ldots \ {}^{\mathrm{o}}\boldsymbol{f}_{\mathrm{r},n-1}$ are first determined. Once ${}^{\mathrm{o}}\boldsymbol{f}_{\mathrm{r1}} \ \ldots \ {}^{\mathrm{o}}\boldsymbol{f}_{\mathrm{r},n-1}$ are given, $\mathbf{C}$ is automatically determined. Matrices $\mathbf{W}^{-}$ and $\mathbf{V}$ are given by solving (5.12). Note that $\mathbf{W}^{-}$ is not always $\mathbf{W}^{\dagger}$, this fact depends upon $\mathbf{C}$.

The internal force between two virtual sticks i and j, which is defined as ${}^{\mathrm{o}}\Delta\boldsymbol{f}_{\mathrm{b}i,j}$, can be represented as

$$ {}^{\mathrm{o}}\Delta\boldsymbol{f}_{\mathrm{b}i,j} \equiv \frac{1}{2}({}^{\mathrm{o}}\boldsymbol{f}_{\mathrm{b}i} - {}^{\mathrm{o}}\boldsymbol{f}_{\mathrm{b}j}) \ . \tag{5.13} $$

Note that (5.13) is one of the representation of internal forces when $n = 2$ and under the assumption of grasping a rigid body. In an n robotic arm cooperation system, ${}_n\mathrm{C}_2$ $(= \frac{1}{2}n(n-1))$ internal forces can be considered. These internal forces can be organized as

$$ {}_n\mathrm{C}_2 \left\{ \begin{bmatrix} {}^{\mathrm{o}}\Delta\boldsymbol{f}_{\mathrm{b}1,2} \\ {}^{\mathrm{o}}\Delta\boldsymbol{f}_{\mathrm{b}1,3} \\ \vdots \\ {}^{\mathrm{o}}\Delta\boldsymbol{f}_{\mathrm{b},n-1,n} \end{bmatrix} \right. = \frac{1}{2}\mathbf{G}\,{}^{\mathrm{o}}\boldsymbol{q}_{\mathrm{b}} \ , \tag{5.14} $$

where $\mathbf{G} \in (\mathbb{R}^{6\,{}_n\mathrm{C}_2 \times 6n})$ is given by

$$ \mathbf{G} \equiv \begin{array}{c} \overbrace{\hphantom{\mathbf{I}_6 \quad -\mathbf{I}_6 \quad \quad \quad \quad 0}}^{6n} \\ \left[\begin{array}{ccccc} \mathbf{I}_6 & -\mathbf{I}_6 & & & \\ \mathbf{I}_6 & & -\mathbf{I}_6 & & \\ \vdots & 0 & & \ddots & \\ \mathbf{I}_6 & & & & -\mathbf{I}_6 \\ 0 & \mathbf{I}_6 & -\mathbf{I}_6 & & \\ 0 & \mathbf{I}_6 & & -\mathbf{I}_6 & \\ \vdots & \vdots & 0 & & \ddots \\ 0 & \mathbf{I}_6 & & & -\mathbf{I}_6 \\ & & \vdots & & \\ 0 & 0 & \dots & \mathbf{I}_6 & -\mathbf{I}_6 \end{array} \right] \end{array} \begin{array}{l} \left.\vphantom{\begin{array}{c}1\\1\\1\\1\end{array}}\right\} 6(n-1) \\ \left.\vphantom{\begin{array}{c}1\\1\\1\\1\end{array}}\right\} 6(n-2) \\ \\ \left.\vphantom{\begin{array}{c}1\\1\\1\end{array}}\right\} 6 \end{array} = \left[\begin{array}{ccccc} \mathbf{I}_6 & & & & \\ \vdots & & -\mathbf{I}_{6(n-1)} & & \\ \mathbf{I}_6 & & & & \\ 0 & \mathbf{I}_6 & & & \\ \vdots & \vdots & -\mathbf{I}_{6(n-2)} & & \\ 0 & \mathbf{I}_6 & & & \\ & & \vdots & & \\ 0 & \dots & 0 & \mathbf{I}_6 & -\mathbf{I}_6 \end{array} \right] \tag{5.15} $$

Because the upper right minor matrix of $\mathbf{G}$ is $-\mathbf{I}_{6(n-1)}$ and the upper left block of $\mathbf{G}$ $(\begin{bmatrix}\mathbf{I}_6 & \dots & \mathbf{I}_6\end{bmatrix}^{\mathrm{T}})$ can be generated by a linear combination of the vectors in $-\mathbf{I}_{6(n-1)}$, the rank of $\mathbf{G}$ is $6(n-1)$, which indicates that ${}^{\mathrm{o}}\Delta\boldsymbol{f}_{\mathrm{b}i,j}$ are not linearly independent. Only $6(n-1)$ vectors of ${}^{\mathrm{o}}\Delta\boldsymbol{f}_{\mathrm{b}i,j}$ among ${}_n\mathrm{C}_2$ are linearly independent. Hence, the internal force of an n robotic arm cooperation system can be represented as $(n-1)$ linear combinations of arbitrary ${}^{\mathrm{o}}\Delta\boldsymbol{f}_{\mathrm{b}i,j}$.

The chosen combinations are organized into an internal force vector ${}^{\mathrm{o}}\boldsymbol{f}_m = \mathbf{C}\,{}^{\mathrm{o}}\boldsymbol{q}_{\mathrm{b}}$, and this internal force set is treated as the set of control variables. Matrix $\mathbf{V}$ can be computed from (5.12).There are an infinite number of representations for the internal force vectors, and these representations can be transformed into each other. The following two internal force vectors are considered, for example.

$$ \begin{aligned} {}^{\mathrm{o}}\boldsymbol{f}^1_{\mathrm{m}} &= \mathbf{C}_1\,{}^{\mathrm{o}}\boldsymbol{q}_{\mathrm{b}} \\ {}^{\mathrm{o}}\boldsymbol{f}^2_{\mathrm{m}} &= \mathbf{C}_2\,{}^{\mathrm{o}}\boldsymbol{q}_{\mathrm{b}} \quad , \end{aligned} \tag{5.16} $$

where ${}^{\mathrm{o}}\boldsymbol{f}^1_{\mathrm{m}}$ and ${}^{\mathrm{o}}\boldsymbol{f}^2_{\mathrm{m}}$ are two different internal force vectors and $\mathbf{C}_1$ and $\mathbf{C}_2$ are two different representations of $\mathbf{C}$ in (5.11).

The internal vectors can be transformed into each other by

$$\begin{aligned} \mathbf{C}_1^{-}\ {}^{o}\boldsymbol{f}^1_{\mathrm{m}} &= {}^{\mathrm{o}}\boldsymbol{q}_{\mathrm{b}} \\ {}^{\mathrm{o}}\boldsymbol{f}^2_{\mathrm{m}} &= \mathbf{C}_2\mathbf{C}_1^{-}\ {}^{o}\boldsymbol{f}^1_{\mathrm{m}} \quad . \end{aligned} \tag{5.17}$$

5.2.3 Determination of Internal Force of m-Humanoid Robots System

As discussed in Section 5.2.2, there are an infinite number of representations for internal force vectors. This section presents a strategy to determine an internal force vector for an m-humanoid robot system.

If it is assumed that a humanoid robot has two arms, there are $2m$ robotic arms in an m-humanoid robot system. When two arms are chosen out of $2m$ arms, the number of combinations is ${}_{2\mathrm{m}}\mathrm{C}_2(= m(2m-1))$. A 6-dimensional internal force/moment vector can be considered between each combination, as described in (5.13). As discussed in Section 5.2.2, $(2m-1)$ vectors out of ${}_{2\mathrm{m}}\mathrm{C}_2$ are independent. The independent $(2m-1)$ vectors are generated by a linear combination of ${}_{2\mathrm{m}}\mathrm{C}_2$ vectors.

First, a force generated by two arms of a humanoid robot is defined as

$${}^{\mathrm{o}}\boldsymbol{f}^i_{\mathrm{a}} \equiv {}^{\mathrm{o}}\boldsymbol{f}^i_{\mathrm{bR}} + {}^{\mathrm{o}}\boldsymbol{f}^i_{\mathrm{bL}}\ , i = 1,\ 2,\ \ldots,\ m \tag{5.18}$$

where ${}^{\mathrm{o}}\boldsymbol{f}^i_{\mathrm{bR}}$ and ${}^{\mathrm{o}}\boldsymbol{f}^i_{\mathrm{bL}}$ denote force vectors generated at the tip of virtual sticks fixed on the right and left hands of the humanoid robot, respectively.

The internal force between the two arms of the humanoid robot is given as

$${}^{\mathrm{o}}\boldsymbol{f}^i_{\mathrm{r}} \equiv \frac{1}{2}({}^{\mathrm{o}}\boldsymbol{f}^i_{\mathrm{bR}} - {}^{\mathrm{o}}\boldsymbol{f}^i_{\mathrm{bL}})\ , i = 1,\ 2,\ \ldots,\ m\ . \tag{5.19}$$

In the strategy discussed in this section, m internal forces given by (5.19) are chosen first. The number of independent vectors is $(2m-1)$. Hence, a further $m-1$ vectors must be chosen. Internal force vectors between two humanoid robots are given by

$${}^{\mathrm{o}}\boldsymbol{f}^{i,j}_{\mathrm{r}} \equiv \frac{1}{2}({}^{\mathrm{o}}\boldsymbol{f}^i_{\mathrm{a}} - {}^{\mathrm{o}}\boldsymbol{f}^j_{\mathrm{a}})\ , i \neq j\ , \tag{5.20}$$

where ${}^{\mathrm{o}}\boldsymbol{f}^i_{\mathrm{a}}$ and ${}^{\mathrm{o}}\boldsymbol{f}^j_{\mathrm{a}}$ are defined by (5.18). Among the m humanoid robots, the number of combinations of two robots is ${}_m\mathrm{C}_2(= \frac{1}{2}m(m-1))$. The remaining $(m-1)$ internal forces can be chosen directly from (5.20), or chosen from a linear combination of ${}^{\mathrm{o}}\boldsymbol{f}^{i,j}_{\mathrm{r}}$, such as $({}^{\mathrm{o}}\boldsymbol{f}^{1,2}_{\mathrm{r}} + {}^{\mathrm{o}}\boldsymbol{f}^{3,4}_{\mathrm{r}})$ or $({}^{\mathrm{o}}\boldsymbol{f}^{1,2}_{\mathrm{r}} + {}^{\mathrm{o}}\boldsymbol{f}^{3,4}_{\mathrm{r}} + {}^{\mathrm{o}}\boldsymbol{f}^{5,6}_{\mathrm{r}})$. Note that the chosen internal forces should be linearly independent with respect to each other. As a case study, the 3-humanoid robot system $(m = 3)$ illustrated in Figure 5.3 is considered. As illustrated in this figure, six arms hold an object. The number of independent vectors for the internal forces is $5\ (= 2 \times 3 - 1)$. In Figure 5.3, ${}^{\mathrm{o}}\boldsymbol{l}^i_{\mathrm{bR}}$ and ${}^{\mathrm{o}}\boldsymbol{l}^i_{\mathrm{bL}}$ $(i = 1, 2, 3)$ are the virtual stick vectors of

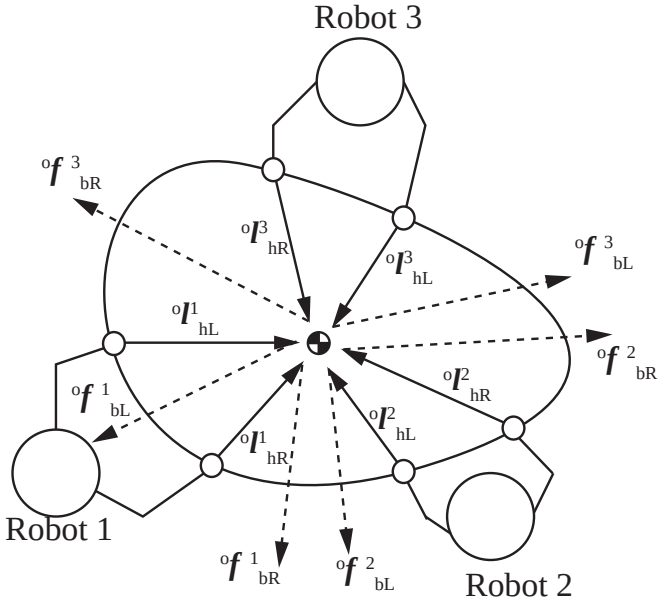


Figure 5.3: Case study: cooperation among three humanoid robots (m=3).

the right and left arms, respectively. The three of the five internal forces are determined from (5.19) as

$$
\begin{aligned}
{}^{\mathrm{o}}\boldsymbol{f}_{\mathrm{r1}} &\equiv {}^{\mathrm{o}}\boldsymbol{f}^{1}_{\mathrm{r}} = \frac{1}{2}({}^{\mathrm{o}}\boldsymbol{f}^{1}_{\mathrm{bR}} - {}^{\mathrm{o}}\boldsymbol{f}^{1}_{\mathrm{bL}}) \ , \\
{}^{\mathrm{o}}\boldsymbol{f}_{\mathrm{r2}} &\equiv {}^{\mathrm{o}}\boldsymbol{f}^{2}_{\mathrm{r}} = \frac{1}{2}({}^{\mathrm{o}}\boldsymbol{f}^{2}_{\mathrm{bR}} - {}^{\mathrm{o}}\boldsymbol{f}^{2}_{\mathrm{bL}}) \ , \\
{}^{\mathrm{o}}\boldsymbol{f}_{\mathrm{r3}} &\equiv {}^{\mathrm{o}}\boldsymbol{f}^{3}_{\mathrm{r}} = \frac{1}{2}({}^{\mathrm{o}}\boldsymbol{f}^{3}_{\mathrm{bR}} - {}^{\mathrm{o}}\boldsymbol{f}^{3}_{\mathrm{bL}}) \ .
\end{aligned}
\tag{5.21}
$$

The remaining two internal forces are chosen from the following three internal forces.

$$
\begin{aligned}
{}^{\mathrm{o}}\boldsymbol{f}^{1,2}_{\mathrm{r}} &= \tfrac{1}{2}({}^{\mathrm{o}}\boldsymbol{f}^{1}_{\mathrm{a}} - {}^{\mathrm{o}}\boldsymbol{f}^{2}_{\mathrm{a}}) \ , \\
{}^{\mathrm{o}}\boldsymbol{f}^{2,3}_{\mathrm{r}} &= \tfrac{1}{2}({}^{\mathrm{o}}\boldsymbol{f}^{2}_{\mathrm{a}} - {}^{\mathrm{o}}\boldsymbol{f}^{3}_{\mathrm{a}}) \ , \\
{}^{\mathrm{o}}\boldsymbol{f}^{3,1}_{\mathrm{r}} &= \tfrac{1}{2}({}^{\mathrm{o}}\boldsymbol{f}^{3}_{\mathrm{a}} - {}^{\mathrm{o}}\boldsymbol{f}^{1}_{\mathrm{a}}) \ \ (= -{}^{\mathrm{o}}\boldsymbol{f}^{2,3}_{\mathrm{r}} - {}^{\mathrm{o}}\boldsymbol{f}^{1,2}_{\mathrm{r}}).
\end{aligned}
\tag{5.22}
$$

If ${}^{\mathrm{o}}\boldsymbol{f}^{1,2}_{\mathrm{r}}$ and ${}^{\mathrm{o}}\boldsymbol{f}^{2,3}_{\mathrm{r}}$ are chosen, these internal forces are organized into ${}^{\mathrm{o}}\boldsymbol{f}_{\mathrm{m}}$ as

$$
{}^{\mathrm{o}}\boldsymbol{f}_{\mathrm{m}} \equiv \begin{bmatrix} {}^{\mathrm{o}}\boldsymbol{f}_{\mathrm{r1}} \\ {}^{\mathrm{o}}\boldsymbol{f}_{\mathrm{r2}} \\ {}^{\mathrm{o}}\boldsymbol{f}_{\mathrm{r3}} \\ {}^{\mathrm{o}}\boldsymbol{f}_{\mathrm{r4}} \\ {}^{\mathrm{o}}\boldsymbol{f}_{\mathrm{r5}} \end{bmatrix} = \begin{bmatrix} \frac{1}{2}\mathbf{I}_6 & -\frac{1}{2}\mathbf{I}_6 & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \frac{1}{2}\mathbf{I}_6 & -\frac{1}{2}\mathbf{I}_6 & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \frac{1}{2}\mathbf{I}_6 & -\frac{1}{2}\mathbf{I}_6 \\ \frac{1}{2}\mathbf{I}_6 & \frac{1}{2}\mathbf{I}_6 & -\frac{1}{2}\mathbf{I}_6 & -\frac{1}{2}\mathbf{I}_6 & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \frac{1}{2}\mathbf{I}_6 & \frac{1}{2}\mathbf{I}_6 & -\frac{1}{2}\mathbf{I}_6 & -\frac{1}{2}\mathbf{I}_6 \end{bmatrix} \begin{bmatrix} {}^{\mathrm{o}}\boldsymbol{f}^{1}_{\mathrm{bR}} \\ {}^{\mathrm{o}}\boldsymbol{f}^{1}_{\mathrm{bL}} \\ {}^{\mathrm{o}}\boldsymbol{f}^{2}_{\mathrm{bR}} \\ {}^{\mathrm{o}}\boldsymbol{f}^{2}_{\mathrm{bL}} \\ {}^{\mathrm{o}}\boldsymbol{f}^{3}_{\mathrm{bR}} \\ {}^{\mathrm{o}}\boldsymbol{f}^{3}_{\mathrm{bL}} \end{bmatrix} = \mathbf{C}\,{}^{\mathrm{o}}\boldsymbol{q}_{\mathrm{b}} \ . \tag{5.23}
$$

5.2.4 Workspace Velocities

The translational and rotational velocities of $\Sigma_{\mathrm{b}i}$ are represented as ${}^{\mathrm{o}}\boldsymbol{v}_{\mathrm{b}i}$ and ${}^{\mathrm{o}}\boldsymbol{\omega}_{\mathrm{b}i}$, respectively. A velocity vector ${}^{\mathrm{o}}\boldsymbol{w}_{\mathrm{b}}$ which includes all velocities of $\Sigma_{\mathrm{b}i}$ is defined as

$$ {}^{\mathrm{o}}\boldsymbol{w}_{\mathrm{b}} \equiv \begin{bmatrix} {}^{\mathrm{o}}\boldsymbol{s}_{\mathrm{b}1} & {}^{\mathrm{o}}\boldsymbol{s}_{\mathrm{b}2} & \dots & {}^{\mathrm{o}}\boldsymbol{s}_{\mathrm{b}i} \end{bmatrix} , \tag{5.24} $$

$$ {}^{\mathrm{o}}\boldsymbol{s}_{\mathrm{b}i} \equiv \begin{bmatrix} {}^{\mathrm{o}}\boldsymbol{v}_{\mathrm{b}i}^{\mathrm{T}} & {}^{\mathrm{o}}\boldsymbol{\omega}_{\mathrm{b}i}^{\mathrm{T}} \end{bmatrix} . \tag{5.25} $$

Denotes that the velocities corresponding to ${}^{\mathrm{o}}\boldsymbol{f}_{\mathrm{a}}$ and ${}^{\mathrm{o}}\boldsymbol{f}_{\mathrm{r}i}$ are ${}^{\mathrm{o}}\boldsymbol{s}_{\mathrm{a}}$ and ${}^{\mathrm{o}}\Delta\boldsymbol{s}_{\mathrm{r}i}$, then generalized velocity vector ${}^{\mathrm{o}}\boldsymbol{u}$ is defined as

$$ {}^{\mathrm{o}}\boldsymbol{u} \equiv \begin{bmatrix} {}^{\mathrm{o}}\boldsymbol{s}_{\mathrm{a}}^{\mathrm{T}} & {}^{\mathrm{o}}\Delta\boldsymbol{s}_{\mathrm{r}1}^{\mathrm{T}} & {}^{\mathrm{o}}\Delta\boldsymbol{s}_{\mathrm{r}2}^{\mathrm{T}} & \dots & {}^{\mathrm{o}}\Delta\boldsymbol{s}_{\mathrm{r,n-1}}^{\mathrm{T}} \end{bmatrix} . \tag{5.26} $$

If it is supposed that ${}^{\mathrm{o}}\boldsymbol{q}_{\mathrm{b}}$ and ${}^{\mathrm{o}}\boldsymbol{h}$ are balanced, then by applying the principle of virtual work to (5.7), the follow relationship is given.

$$ {}^{\mathrm{o}}\boldsymbol{h}^{\mathrm{T}}\,{}^{\mathrm{o}}\boldsymbol{u} = {}^{\mathrm{o}}\boldsymbol{q}_{\mathrm{b}}^{\mathrm{T}}\,{}^{\mathrm{o}}\boldsymbol{w}_{\mathrm{b}} = (\mathbf{U}\,{}^{\mathrm{o}}\boldsymbol{h})^{\mathrm{T}}\,{}^{\mathrm{o}}\boldsymbol{w}_{\mathrm{b}} , \tag{5.27} $$

$$ {}^{\mathrm{o}}\boldsymbol{u} = \mathbf{U}^{\mathrm{T}}\,{}^{\mathrm{o}}\boldsymbol{w}_{\mathrm{b}} . \tag{5.28} $$

${}^{\mathrm{o}}\boldsymbol{w}_{\mathrm{b}}$ is calculated from ${}^{\mathrm{o}}\boldsymbol{w}_{\mathrm{h}}$(see Appendix A) in practical use.

5.2.5 Workspace Coordinates

The position and orientation of the origins of $\Sigma_{\mathrm{b}i}$ and Σ_{a} with respect to Σ_{o} are defined as

$$ {}^{\mathrm{o}}\boldsymbol{p}_{\mathrm{b}i} \equiv \begin{bmatrix} {}^{\mathrm{o}}\boldsymbol{x}_{\mathrm{b}i}^{\mathrm{T}} & {}^{\mathrm{o}}\boldsymbol{\phi}_{\mathrm{b}i}^{\mathrm{T}} \end{bmatrix}^{\mathrm{T}} , \tag{5.29} $$

$$ {}^{\mathrm{o}}\boldsymbol{p}_{\mathrm{a}} \equiv \begin{bmatrix} {}^{\mathrm{o}}\boldsymbol{x}_{\mathrm{a}}^{\mathrm{T}} & {}^{\mathrm{o}}\boldsymbol{\phi}_{\mathrm{a}}^{\mathrm{T}} \end{bmatrix}^{\mathrm{T}} , \tag{5.30} $$

respectively. ${}^{\mathrm{o}}\boldsymbol{x}_{\mathrm{b}i}$ and ${}^{\mathrm{o}}\boldsymbol{x}_{\mathrm{a}}$ represent the positions; ${}^{\mathrm{o}}\boldsymbol{\phi}_{\mathrm{b}i}$ and ${}^{\mathrm{o}}\boldsymbol{\phi}_{\mathrm{a}}$ are rotation angles such as Euler angles.

${}^{\mathrm{o}}\boldsymbol{s}_{\mathrm{b}i}$, ${}^{\mathrm{o}}\boldsymbol{s}_{\mathrm{a}}$ and ${}^{\mathrm{o}}\Delta\boldsymbol{s}_{\mathrm{r}i}$ can be calculated from the derivatives of ${}^{\mathrm{o}}\boldsymbol{p}_{\mathrm{b}i}$, ${}^{\mathrm{o}}\boldsymbol{p}_{\mathrm{a}}$ and ${}^{\mathrm{o}}\Delta\boldsymbol{p}_{\mathrm{r}i}$ as follow.

$$ \begin{aligned} {}^{\mathrm{o}}\boldsymbol{s}_{\mathrm{b}i} &= \mathbf{B}_{\mathrm{s}}({}^{\mathrm{o}}\boldsymbol{\phi}_{\mathrm{b}i})\,{}^{\mathrm{o}}\dot{\boldsymbol{p}}_{\mathrm{b}i} , \\ {}^{\mathrm{o}}\boldsymbol{s}_{\mathrm{a}} &= \mathbf{B}_{\mathrm{s}}({}^{\mathrm{o}}\boldsymbol{\phi}_{\mathrm{a}})\,{}^{\mathrm{o}}\dot{\boldsymbol{p}}_{\mathrm{a}} , \\ {}^{\mathrm{o}}\Delta\boldsymbol{s}_{\mathrm{r}i} &= \mathbf{B}_{\mathrm{s}}({}^{\mathrm{o}}\boldsymbol{\phi}_{\mathrm{a}})\,{}^{\mathrm{o}}\Delta\dot{\boldsymbol{p}}_{\mathrm{r}i} , \end{aligned} \tag{5.31} $$

Where $\mathbf{B}_{\mathrm{s}}(\boldsymbol{\phi})$ is given by

$$ \mathbf{B}_{\mathrm{s}} = \begin{bmatrix} \mathbf{I}_3 & \mathbf{0} \\ \mathbf{0} & \mathbf{B}_{\mathrm{o}}(\boldsymbol{\phi}) \end{bmatrix} . \tag{5.32} $$

$\mathbf{B}_{\mathrm{o}}(\boldsymbol{\phi})$ is a 3×3 matrix calculated from $\boldsymbol{\phi}$ (see Appendix B). Substituting equations (5.31) into (5.28) yields

$$\mathrm{diag}[\ \mathbf{B}_{\mathrm{s}}({}^{\mathrm{o}}\boldsymbol{\phi}_{\mathrm{a}}),\ \mathbf{B}_{\mathrm{s}}({}^{\mathrm{o}}\boldsymbol{\phi}_{\mathrm{a}}),\ \ldots\ ]\ {}^{\mathrm{o}}\dot{\boldsymbol{z}} = \mathbf{U}^{\mathrm{T}}\mathrm{diag}[\ \mathbf{B}_{\mathrm{s}}({}^{\mathrm{o}}\boldsymbol{\phi}_{\mathrm{b1}}),\ \mathbf{B}_{\mathrm{s}}({}^{\mathrm{o}}\boldsymbol{\phi}_{\mathrm{b2}}),\ \ldots\ ]\ {}^{\mathrm{o}}\dot{\boldsymbol{y}}_{\mathrm{b}}\ , \quad (5.33)$$

$$\begin{aligned} {}^{\mathrm{o}}\boldsymbol{y}_{\mathrm{b}} &\equiv \begin{bmatrix} {}^{\mathrm{o}}\boldsymbol{p}_{\mathrm{b1}}^{\mathrm{T}} & {}^{\mathrm{o}}\boldsymbol{p}_{\mathrm{b2}}^{\mathrm{T}} & \ldots & {}^{\mathrm{o}}\boldsymbol{p}_{\mathrm{b}n}^{\mathrm{T}} \end{bmatrix}^{\mathrm{T}}\ , \\ {}^{\mathrm{o}}\boldsymbol{z} &\equiv \begin{bmatrix} {}^{\mathrm{o}}\boldsymbol{p}_{\mathrm{a}}^{\mathrm{T}} & {}^{\mathrm{o}}\Delta\boldsymbol{p}_{\mathrm{r1}}^{\mathrm{T}} & \ldots & {}^{\mathrm{o}}\Delta\boldsymbol{p}_{\mathrm{r},n-1}^{\mathrm{T}} \end{bmatrix}^{\mathrm{T}}\ . \end{aligned}$$

Assumed that tip frames $\Sigma_{\mathrm{b}i}$ are very close to the object frame Σ_{a}, the orientation angles of these frames with respect to Σ_{o} are almost equal. Therefore,

$$\mathbf{B}_{\mathrm{s}}({}^{\mathrm{o}}\boldsymbol{\phi}_{\mathrm{b}i}) \approx \mathbf{B}_{\mathrm{s}}({}^{\mathrm{o}}\boldsymbol{\phi}_{\mathrm{b}j}) \approx \mathbf{B}_{\mathrm{s}}({}^{\mathrm{o}}\boldsymbol{\phi}_{\mathrm{a}})\ . \quad (5.34)$$

It is assume that the initial values of ${}^{\mathrm{o}}\boldsymbol{p}_{\mathrm{a}}$ and ${}^{\mathrm{o}}\boldsymbol{p}_{\mathrm{b}i}$ are all equal, and the initial val-us of ${}^{\mathrm{o}}\Delta\boldsymbol{p}_{\mathrm{r}i}$ are zero. Base on this assumption, the integration of (5.33) is given as

$${}^{\mathrm{o}}\boldsymbol{z} = \mathbf{U}^{\mathrm{T}}\ {}^{\mathrm{o}}\boldsymbol{y}_{\mathrm{b}}\ . \quad (5.35)$$

The generalized force vector ${}^{\mathrm{o}}\boldsymbol{h}$, velocity vector ${}^{\mathrm{o}}\boldsymbol{u}$ and position vector ${}^{\mathrm{o}}\boldsymbol{z}$ are defined with respect to the world coordinate frame Σ_{o}. However for the internal forces $\boldsymbol{f}_{\mathrm{r}i}$, relative velocities $\Delta\boldsymbol{s}_{\mathrm{r}i}$ and relative positions along the internal force $\Delta\boldsymbol{p}_{\mathrm{r}i}$, it is convenient to be represented with respect to the object coordinate frame Σ_{a}. Therefore they are redefined as

$$\boldsymbol{h} \equiv \begin{bmatrix} {}^{\mathrm{o}}\boldsymbol{f}_{\mathrm{a}}^{\mathrm{T}} & {}^{\mathrm{a}}\boldsymbol{f}_{\mathrm{r1}}^{\mathrm{T}} & \ldots & {}^{\mathrm{a}}\boldsymbol{f}_{\mathrm{r},n-1}^{\mathrm{T}} \end{bmatrix}^{\mathrm{T}}\ , \quad (5.36)$$

$$\boldsymbol{z} \equiv \begin{bmatrix} {}^{\mathrm{o}}\boldsymbol{p}_{\mathrm{a}}^{\mathrm{T}} & {}^{\mathrm{a}}\Delta\boldsymbol{p}_{\mathrm{r1}}^{\mathrm{T}} & \ldots & {}^{\mathrm{a}}\Delta\boldsymbol{p}_{\mathrm{r},n-1}^{\mathrm{T}} \end{bmatrix}^{\mathrm{T}}\ . \quad (5.37)$$

In addition, generalized velocity vector $\boldsymbol{u}$ is defines as

$$\boldsymbol{u} \equiv \begin{bmatrix} {}^{\mathrm{o}}\boldsymbol{s}_{\mathrm{a}}^{\mathrm{T}} & {}^{\mathrm{a}}\Delta\boldsymbol{s}_{\mathrm{r1}}^{\mathrm{T}} & \ldots & {}^{\mathrm{a}}\Delta\boldsymbol{s}_{\mathrm{r},n-1}^{\mathrm{T}} \end{bmatrix}^{\mathrm{T}}\ . \quad (5.38)$$

The relationship between $\boldsymbol{u}$ and $\dot{\boldsymbol{z}}$ is given by

$$\boldsymbol{u} = \mathbf{B}_{\mathrm{a}}\dot{\boldsymbol{z}}\ , \quad (5.39)$$

where $\mathbf{B}_{\mathrm{a}}$ translates the derivative of rotation angles into angular velocity (see Appendix B). Let ${}^{\mathrm{a}}\mathbf{A}_{\mathrm{o}}$ be the 3×3 rotational transformation matrix from Σ_{o} to Σ_{a}. ${}^{\mathrm{a}}\mathbf{A}_{\mathrm{o}}$ is calculated from the orientation components ${}^{\mathrm{o}}\boldsymbol{\phi}_{\mathrm{a}}$ of ${}^{\mathrm{o}}\boldsymbol{p}_{\mathrm{a}}$. ${}^{\mathrm{o}}\boldsymbol{h}$ and ${}^{\mathrm{o}}\boldsymbol{u}$ is transferred into $\boldsymbol{h}$ and $\boldsymbol{u}$ as follow:

$$\boldsymbol{h} = \mathbf{H}\ {}^{\mathrm{o}}\boldsymbol{h}\ , \quad (5.40)$$

$$\boldsymbol{u} = \mathbf{H}\ {}^{\mathrm{o}}\boldsymbol{u}\ . \quad (5.41)$$

Where

$$\mathbf{H} \equiv \mathrm{diag}[\mathbf{I}_6,\ {}^{\mathrm{a}}\mathbf{A}_{\mathrm{o}},\ {}^{\mathrm{a}}\mathbf{A}_{\mathrm{o}},\ \ldots\ ] \in \mathbb{R}^{6\times 6n}\ , \quad (5.42)$$

5.2.6 Relationship between Arm Joint Velocities and Generalized Velocity Vector $\boldsymbol{u}$

The velocity vector of the hand ${}^{\mathrm{o}}\boldsymbol{s}_{\mathrm{h}i}$ is calculated from the joint velocity vector $\boldsymbol{v}_i$ as

$$ {}^{\mathrm{o}}\boldsymbol{s}_{\mathrm{h}i} = \mathbf{J}_i \boldsymbol{v}_i \ , \tag{5.43} $$

where $\mathbf{J}_i$ is the Jacobian matrix of the ith robotic arm. Then the velocity vector ${}^{\mathrm{o}}\boldsymbol{w}_{\mathrm{h}}$ is calculated by

$$ {}^{\mathrm{o}}\boldsymbol{w}_{\mathrm{h}} = \mathrm{diag}[\mathbf{J}_1,\ \mathbf{J}_2,\ \ldots\ \mathbf{J}_n][\ \boldsymbol{v}_1^{\mathrm{T}}, \boldsymbol{v}_2^{\mathrm{T}},\ \ldots\ \boldsymbol{v}_n^{\mathrm{T}}]^{\mathrm{T}}\ \equiv \mathbf{J}_{\mathrm{h}}\dot{\boldsymbol{\Theta}} \ , \tag{5.44} $$

where $\dot{\boldsymbol{\Theta}}$ is the joint angular velocity. Substituting (5.44) and (A.8) into (5.41), the generalized velocity is also given by

$$ \boldsymbol{u} = \mathbf{H}\mathbf{U}^{\mathrm{T}}({}^{\mathrm{o}}\mathbf{D}_{\mathrm{h}}^{\mathrm{T}})^{-1}\mathbf{J}_{\mathrm{h}}\dot{\boldsymbol{\Theta}} \equiv \mathbf{J}\dot{\boldsymbol{\Theta}} \ . \tag{5.45} $$

where $\mathbf{J}$ is a Jacobian matrix involves all joint angles.

From the principle of virtual work, the relationship between all joint torque $\boldsymbol{\Lambda}$ and the generalized force $\boldsymbol{h}$ is given by

$$ \begin{aligned} \dot{\boldsymbol{z}}^{\mathrm{T}}\boldsymbol{h} &= \boldsymbol{\sigma}^{\mathrm{T}}\boldsymbol{\Lambda} \\ (\mathbf{J}\boldsymbol{\sigma})^{\mathrm{T}}\boldsymbol{h} &= \boldsymbol{\sigma}^{\mathrm{T}}\boldsymbol{\Lambda} \\ \boldsymbol{\sigma}^{\mathrm{T}}\mathbf{J}^{\mathrm{T}}\boldsymbol{h} &= \boldsymbol{\sigma}^{\mathrm{T}}\boldsymbol{\Lambda} \\ \boldsymbol{\Lambda} &= \mathbf{J}^{\mathrm{T}}\boldsymbol{h} \ , \end{aligned} \tag{5.46} $$

5.3 Position/Force Hybrid Control

In this section, a hybrid position/force controller is presented.

A block diagram of the hybrid position/force controller is presented in Figure 5.7.

Control space is divided into external space and internal space. In the external space, position and orientation of the transported object or external force and moment to apply to the object are controlled, while in the internal space, the internal force and moment are controlled by force controller or position controller. A selection matrix $\mathbf{S}$ specifies force controller or position controller for every axes in the control space.

The current state of the external and internal force vector $\boldsymbol{h}_{\mathrm{c}}$ is calculated from equations (5.11) and (5.36) by using force signals measured by wrist force sensors of the robots. The current state of the position of the object and relative position along the internal forces $\boldsymbol{z}_c$ is calculated from equations (5.35) and (5.37) with solving forward kinematics. A robot operator gives the references $\boldsymbol{h}_{\mathrm{r}}$ and/or $\boldsymbol{z}_{\mathrm{r}}^{\mathrm{input}}$ as illustrated in Figure 5.7.

Most of recent humanoid robot is controlled by joint position controller with high reduction gear transmission. Hence force control law proposed in this paper is converted into an equivalent position control law. Dynamic forces generated by moving the

transported object and robot body affect on actuator's torque. However the dynamic effect decreases in proportion to the square of the reduction ratio, hence general joint PD servo generally achieves good performance in position control. Therefore, dynamics is not considered in the controller design presented in this section. Consideration of dynamics is future work. It is assumed that the mass and moment of inertia of the transported object are assumed to be small and negligible.

The following subsections present the details of the hybrid position/force controller.

5.3.1 Force Control

Since the humanoid robot HRP-2 [7] used in this research is a position controlled robot, the force control law (5.46) cannot be directly applied. The force control command must be converted into a corresponding position control command. It is assumed that a compliance between deviation of generalized force $\delta\boldsymbol{h}$ and deviation of generalized position $\delta\boldsymbol{z}$ as

$$k_{\mathrm{h}}\delta\boldsymbol{h} = \mathbf{B}_{\mathrm{a}}\delta\boldsymbol{z} \tag{5.47}$$

where k_{h} behaves as a gain of force controller. From (5.47), the reference of the generalized position for force controller is given by

$$\delta\boldsymbol{z}_{\mathrm{rh}} = \mathbf{B}_{\mathrm{a}}^{-1}k_{\mathrm{h}}\mathbf{S}(\boldsymbol{h}_{\mathrm{r}} - \boldsymbol{h}_{\mathrm{c}}) \ , \tag{5.48}$$

where $\boldsymbol{h}_{\mathrm{r}}$ and $\boldsymbol{h}_{\mathrm{c}}$ are the reference and current vectors of generalized force, respectively. $\mathbf{S}$ ($\in \mathbb{R}^{6n\times 6n}$) is a diagonal selection matrix for switching force/position control ('1' for a diagonal element corresponds to force control while '0' corresponds to position control).

5.3.2 Position Control

The reference of the generalized position for position controller is given by

$$\boldsymbol{z}_{\mathrm{rp}} = (\mathbf{I}_{6n} - \mathbf{S})\boldsymbol{z}_{\mathrm{r}}^{\mathrm{input}} \ , \tag{5.49}$$

where $\mathbf{I}_{6n}$ is a $6n\times 6n$ identity matrix, and $\boldsymbol{z}_{\mathrm{r}}^{\mathrm{input}}$ is the user specified generalized position command.

5.3.3 Hybrid Control

In order to realize a hybrid controller, the reference of generalized position $\boldsymbol{z}_{\mathrm{r}}$ is given as

$$\boldsymbol{z}_{\mathrm{r}} = \delta\boldsymbol{z}_{\mathrm{rh}} + \boldsymbol{z}_{\mathrm{rp}} = \mathbf{B}_{\mathrm{a}}^{-1}k_{\mathrm{h}}\mathbf{S}(\boldsymbol{h}_{\mathrm{r}} - \boldsymbol{h}_{\mathrm{c}}) + (\mathbf{I}_{6n} - \mathbf{S})\boldsymbol{z}_{\mathrm{r}}^{\mathrm{input}} \ . \tag{5.50}$$

The joint angle reference $\boldsymbol{\Theta}_{\mathrm{r}}$ is given by

$$\boldsymbol{\Theta}_{\mathrm{r}} = \boldsymbol{\Theta}_{\mathrm{c}} + \delta\boldsymbol{\Theta}_{\mathrm{r}} \ , \tag{5.51}$$

$$\delta\boldsymbol{\Theta}_{\mathrm{r}} = \mathbf{J}^{\dagger}k_{\mathrm{z}}\mathbf{B}_{\mathrm{a}}(\boldsymbol{z}_{\mathrm{r}} - \boldsymbol{z}_{\mathrm{c}}) \ , \tag{5.52}$$

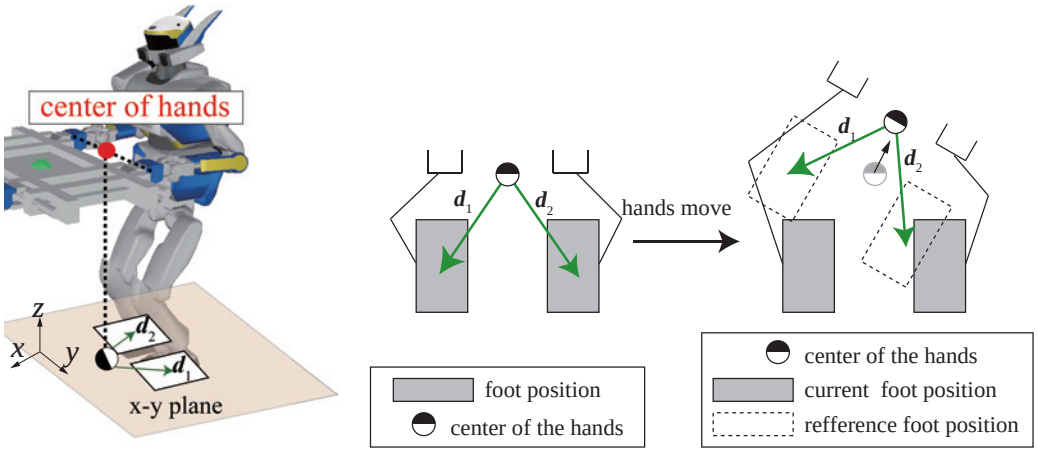


Figure 5.4: Walking planning.

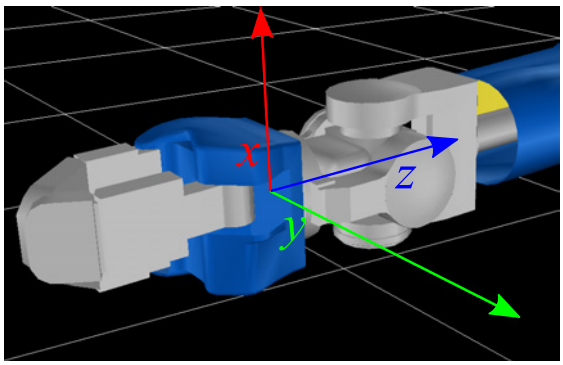


Figure 5.5: Positioning of the force sensors.

where $\mathbf{\Theta}_c$ is the current joint angle vector, and $\mathbf{J}^\dagger$ is the pseudo inverse matrix of $\mathbf{J}$.

In this way, both outputs of force control and position control are integrated into a position control command.

5.4 Application to Multiple Humanoid Robots

A cooperative object transportation task of n humanoid robots is considered. A centralized control system is implemented, and its concept is illustrated in Figure 5.8. The centralized control system explicitly controls the positions of the transported object or the force applied to it. The robot motions to achieve the object reference position or force are calculated and sent simultaneously to all robots. Therefore, the time lag among the robots is expected to be small. The centralized controller accepts object reference position or force command from an operator and force sensor information.

In order to maintain walking stability, only the arms are used for force/position

hybrid control in the cooperation task. All the reference hand positions and attitudes of every robot are computed from the hybrid controller proposed in Section 5.3.

The concept of a walking plan for the cooperation task is shown in Figure 5.4. The walking plan is individually adapted for each humanoid robot. After all the robots grasp the object, each robot records the relative displacement from the center of their two hands to their foot position in the x-y plane ($\boldsymbol{d}_1$, $\boldsymbol{d}_2$). When the robots start to move the object, the reference foot position is continuously calculated using $\boldsymbol{d}_1$ and $\boldsymbol{d}_2$. When the error between the current and reference foot position exceeds a specified threshold, the robot starts to walk. Once the reference foot position is determined, the reference zero moment point (ZMP) trajectory is calculated by interpolating the discrete foot positions from the current to the reference one. The reference center of mass (CoM) trajectory is computed using preview control theory [40]. The robots stop walking when their foot position errors become lower than a threshold. This error includes position error e_{p} and attitude error in yaw direction e_{A}, which are defined by

$$\begin{aligned} e_{\mathrm{p}} &= \| \boldsymbol{d}_{\mathrm{r}} - \boldsymbol{d}_{\mathrm{c}} \| \ , \\ e_{\mathrm{A}} &= |\psi_{\mathrm{r}} - \psi_{\mathrm{c}}| \ , \end{aligned} \tag{5.53}$$

where $\boldsymbol{d}_{\mathrm{r}}$ and $\boldsymbol{d}_{\mathrm{c}}$ are the reference and current foot positions, and ψ_{r} and ψ_{c} are the reference and current yaw angles of the foot, respectively. Errors e_p and e_A are both checked for both left and right feet after every step to decide if a step should be stopped or the next step should be started. The flow chart of walking planning is shown in Figure 5.6.

Because the internal force of the system is controlled by stretching and shrinking the arms, and the reference foot position of a robot is calculated from the center of its two hands, the relative position error between robots is compensated. In order to integrate the walking control, each humanoid robot is controlled based on the following kinematic equations

$$\delta\boldsymbol{\Theta} = \mathbf{J}^{\dagger}_{\mathrm{whole}} \delta\boldsymbol{p}_{\mathrm{whole}} \tag{5.54}$$

where

$$\delta\boldsymbol{\Theta} = \boldsymbol{\Theta}^{\mathrm{ref}} - \boldsymbol{\Theta}^{\mathrm{cur}}, \ \mathbf{J}_{\mathrm{whole}} = \begin{bmatrix} \mathbf{J}_{\mathrm{CoM}} \\ \mathbf{J}_{\mathrm{waist}} \\ \mathbf{J}_{\mathrm{swingFoot}} \\ \mathbf{J}_{\mathrm{Hands}} \end{bmatrix}, \ \delta\boldsymbol{p}_{\mathrm{whole}} = \begin{bmatrix} \boldsymbol{r}^{\mathrm{ref}}_{\mathrm{CoM}} - \boldsymbol{r}^{\mathrm{cur}}_{\mathrm{CoM}} \\ \boldsymbol{t}^{\mathrm{ref}}_{\mathrm{waist}} - \boldsymbol{t}^{\mathrm{cur}}_{\mathrm{waist}} \\ \boldsymbol{p}^{\mathrm{ref}}_{\mathrm{swingFoot}} - \boldsymbol{p}^{\mathrm{cur}}_{\mathrm{swingFoot}} \\ \boldsymbol{p}^{\mathrm{ref}}_{\mathrm{Hands}} - \boldsymbol{p}^{\mathrm{cur}}_{\mathrm{Hands}} \end{bmatrix},$$

and $\mathbf{J}^{\dagger}_{\mathrm{whole}}$ is the pseudo inverse matrix of $\mathbf{J}_{\mathrm{whole}}$.

The control variables are:

$\boldsymbol{r}_{\mathrm{CoM}} \in \mathbb{R}^3$: position of center of mass

$\boldsymbol{t}_{\mathrm{waist}} \in \mathbb{R}^3$: attitude of the waist

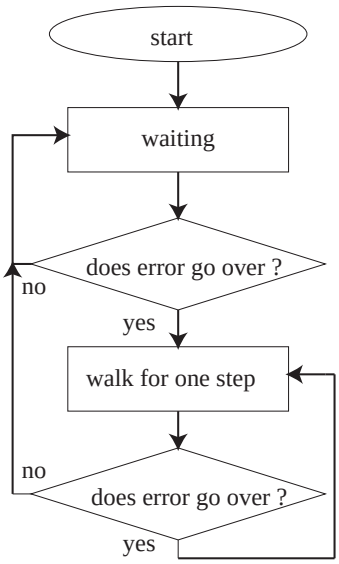


Figure 5.6: Flow chart of walking planning.

$\boldsymbol{p}_{\text{swingFoot}} \in \mathbb{R}^6$: position and attitude of the swing foot

$\boldsymbol{p}_{\text{Hands}} \in \mathbb{R}^{12}$: position and attitude of both hands

$\dot{\boldsymbol{\Theta}}$: all-joint angular velocity

$\mathbf{J}_{\text{CoM}}$: Jacobian matrix mapping $\dot{\boldsymbol{\Theta}}$ to $\dot{\boldsymbol{r}}_{\text{CoM}}$

$\mathbf{J}_{\text{waist}}$: Jacobian matrix mapping $\dot{\boldsymbol{\Theta}}$ to $\dot{\boldsymbol{t}}_{\text{waist}}$

$\mathbf{J}_{\text{swingFoot}}$: Jacobian matrix mapping $\dot{\boldsymbol{\Theta}}$ to $\dot{\boldsymbol{p}}_{\text{swingFoot}}$

$\mathbf{J}_{\text{Hands}}$: Jacobian matrix mapping $\dot{\boldsymbol{\Theta}}$ to $\dot{\boldsymbol{p}}_{\text{Hands}}$

The upper right suffix "ref" indicates the reference value, while "cur" indicates the current value. The reference position of hand $\boldsymbol{p}^{\text{ref}}_{\text{Hands}}$ is given by the hybrid position/force controller, and $\boldsymbol{r}^{\text{ref}}_{\text{CoM}}$, $\boldsymbol{t}^{\text{ref}}_{\text{waist}}$, and $\boldsymbol{p}^{\text{ref}}_{\text{swingFoot}}$, are given by a walking pattern generator. Matrix $\mathbf{J}_{\text{swingFoot}}$ differs for the left and right legs. Hence, the $\mathbf{J}_{\text{swingFoot}}$ matrices corresponding to the left and right legs are alternately used depending upon the swing phase. When the robot is in the double support phase, the inverse kinematics are solved by setting $\boldsymbol{p}^{\text{ref}}_{\text{swingFoot}} = \boldsymbol{p}^{\text{cur}}_{\text{swingFoot}}$. The overall control law of the system is illustrated in Figure 5.7. FK and InvK in Figure 5.7 indicate the forward kinematics calculation and inverse kinematics solutions, respectively, and $\boldsymbol{\Theta}^{\text{ref}}_i$ is the servo motor command of robot i. Equation (5.54) is used to calculate $\boldsymbol{\Theta}^{\text{ref}}_i$. Meanwhile, the current generalized force $\boldsymbol{h}_c$ is computed by the force data measured from the force sensors mounted on each robot's hands. Virtual force sensors are mounted between the hand and wrist of each arm, as illustrated in Figure 5.5, in the same place as on the real humanoid robot HRP-2. The force sensor signal is simulated in the dynamic simulator OpenHRP-3.

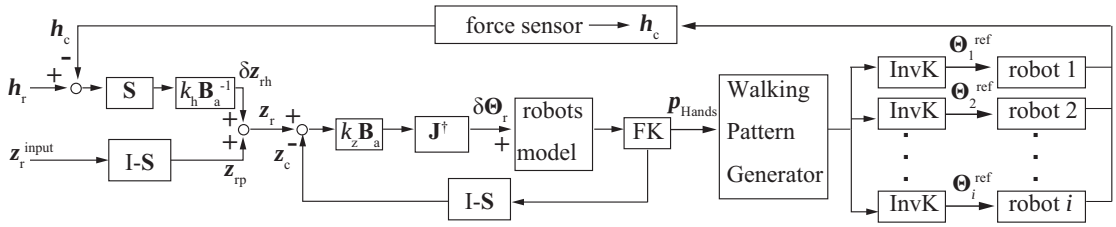


Figure 5.7: Control law of the system.

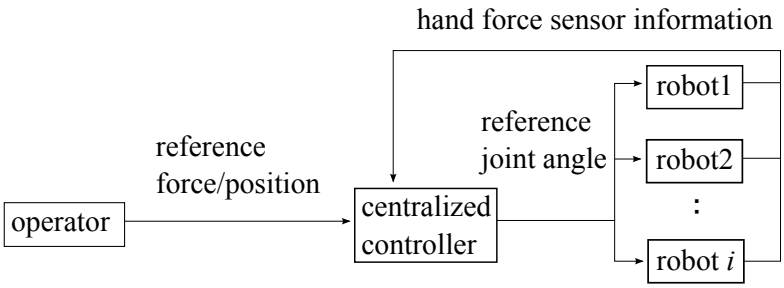


Figure 5.8: Concept of a centralized control system.

5.5 Dynamic Simulation

In order to validate the proposed symmetry cooperation framework, dynamic simulations were performed. Although dynamic effects were not considered in the design of the robot controller presented in Section 5.3, all dynamic effects were computed in the simulations presented in this section.

5.5.1 Comparison with Leader-Follower Type Cooperation

The proposed symmetry type cooperation was compared with leader-follower type cooperation by performing dynamic simulations using OpenHRP [34]. The leader-follower type cooperation in the simulation was basically the same as the method proposed in [46], but impedance control was not applied to the arms in order to perform it under the same conditions as the symmetry type cooperation. The results of the simulations are shown in Figures 5.9 and 5.10. The same constant reference velocity was assigned to the holding object in the x direction.

In the leader-follower type cooperation, the leader robot lost its balance because of the time-lag between the leader and follower robots, as shown in Figure 5.9(a). This figure shows the ZMP trajectory of the leader robot during the simulation. Because the leader robot moved too fast and the follower robot could not respond quickly enough, at approximately $t = 7.2$ s, the ZMP in the x direction went to the front fringe of the support polygon.

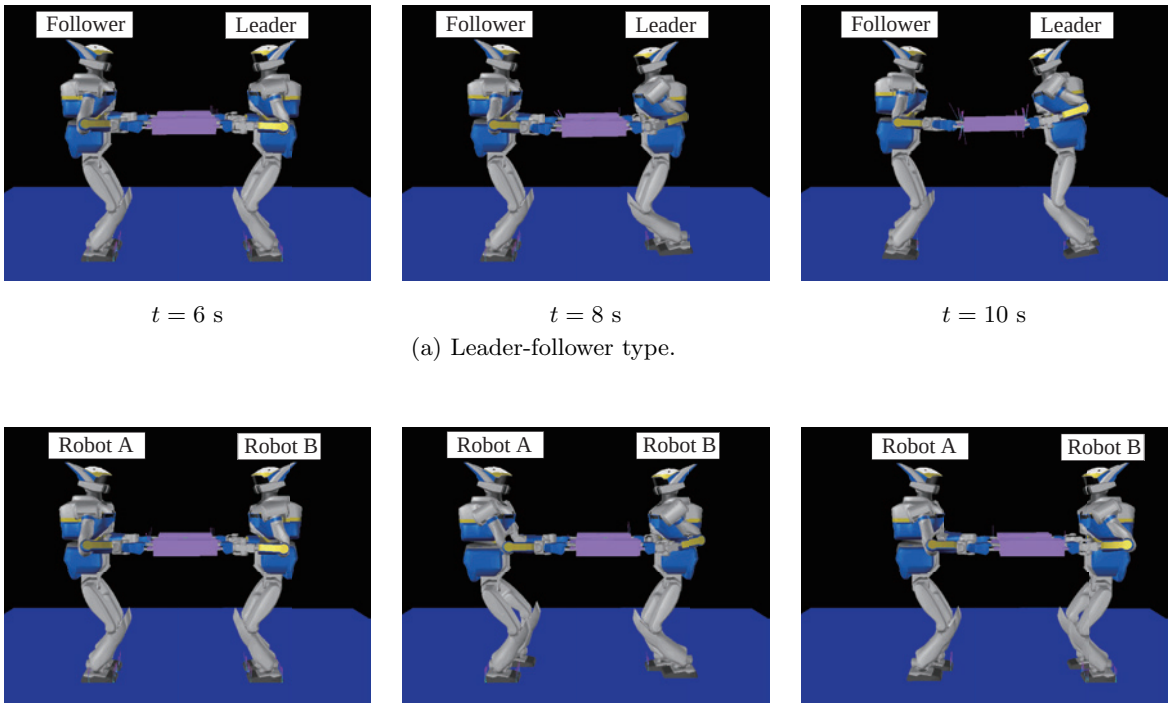
In the symmetry type cooperation, the two robots stably transported the object, as shown in Figure 5.9(b), which shows the ZMP trajectory of robot B during simulation. The ZMP always stayed near the center of the support polygon. These results clearly show the advantage of the proposed symmetry type cooperation over the leader-follower type cooperation.

However, a limitation of moving velocity exists in both cooperation types. Figure 5.11(a) shows a simple 2D model of the follower robot in leader-follower type cooperation, where x is the object displacement moved by the leader robot, k is the stiffness at the hand of the follower robot, F is the force applied to the follower robot caused by the movement of the object, Z_a is the height of the object, M is the mass of the robot, and g is gravitational acceleration. Further $P_{\max}$ is the maximum displacement of ZMP calculated from the region of the foot, and P_{IZMP} is the imaginary ZMP [43]. In order to prevent unexpected tilting, P_{IZMP} must be less than $P_{\max}$. Hence, force F applied to the follower robot must satisfy $FZ_{\mathrm{a}} < MgP_{\max}$. If the applied force F can be approximated by $F = kx$, the object displacement x during a step must satisfy the following condition,

$$x < \frac{MgP_{\max}}{kZ_{\mathrm{a}}} \ . \tag{5.55}$$

Therefore, if the object displacement x during a step does not satisfy (5.55), the follower robot will fall down. The conditions used in the dynamic simulations presented in Figures 5.9 and 5.10 were as follows: $M = 54$ kg, $P_{\max} = 0.13$ m, $Z_{\mathrm{a}} = 0.96$ m, and $k = 1000$ N/m, and the velocity along x axis of the leader robot was $\dot{x} = 0.1$ m/s. Substituting the above simulation conditions into (5.55), the stability condition is obtained as: $0.1\ t < 0.072$, where t is the elapsed time after the leader robot starts moving. Therefore, if the follower robot does not take its next step before $t = 0.72$ s, the ZMP of the follower robot (or leader robot) will reach the anterior boundary. In the simulation of leader-follower type cooperation presented in Figure 5.9(a), the follower robot started walking approximately 1.3 s later than the leader robot started walking. Therefore, the leader robot could not keep its balance, as presented in Figure 5.9(a). This is just a simple 2D analysis, however, from this analysis, it is concluded that the stability of the leader-follower type cooperation strongly depends upon the velocity of the leader robot.

In the symmetry type cooperation proposed in this paper, object displacement x is theoretically equal to zero because the robots simultaneously move. However, if the reference object velocity v_{a} is faster than robot moving velocity v_{R}, as illustrated in Figure 5.11(b), the error will accumulate and the robot may fall down, although this error can be compensated for by adjusting the hand position to a certain extent. Therefore, the reference object velocity should be slower than the maximum moving velocity of the robots.



$t = 6$ s $\quad$ $t = 8$ s $\quad$ $t = 10$ s

(a) Leader-follower type.

$t = 18$ s $\quad$ $t = 20$ s $\quad$ $t = 22$ s

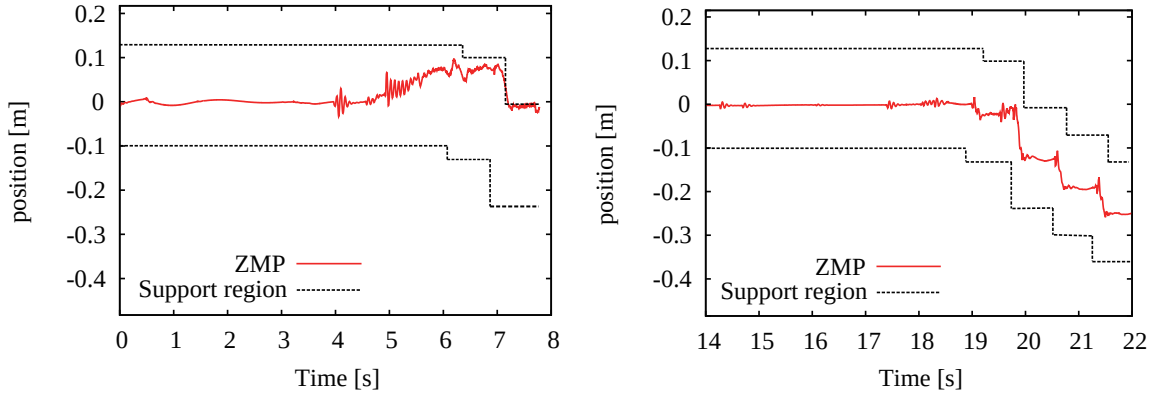
(b) Symmetry type.

Figure 5.9: Comparison between leader-follower type and symmetry type.

5.5.2 Symmetry Cooperation Among Four Humanoid Robots

The proposed method was implemented for four humanoid robots HRP-2 (eight arms in total) and verified with dynamic simulator OpenHRP. In this simulation, the following internal force set are chosen.

$$
\begin{aligned}
{}^{\mathrm{o}}\boldsymbol{f}_{\mathrm{r1}} \equiv {}^{\mathrm{o}}\boldsymbol{f}_{\mathrm{r}}^{\mathrm{A,C}} + {}^{\mathrm{o}}\boldsymbol{f}_{\mathrm{r}}^{\mathrm{B,D}} &= \tfrac{1}{2}{}^{\mathrm{o}}\boldsymbol{f}_{\mathrm{b1}} + \tfrac{1}{2}{}^{\mathrm{o}}\boldsymbol{f}_{\mathrm{b2}} + \tfrac{1}{2}{}^{\mathrm{o}}\boldsymbol{f}_{\mathrm{b3}} + \tfrac{1}{2}{}^{\mathrm{o}}\boldsymbol{f}_{\mathrm{b4}} - \tfrac{1}{2}{}^{\mathrm{o}}\boldsymbol{f}_{\mathrm{b5}} - \tfrac{1}{2}{}^{\mathrm{o}}\boldsymbol{f}_{\mathrm{b6}} - \tfrac{1}{2}{}^{\mathrm{o}}\boldsymbol{f}_{\mathrm{b7}} - \tfrac{1}{2}{}^{\mathrm{o}}\boldsymbol{f}_{\mathrm{b8}} \\
{}^{\mathrm{o}}\boldsymbol{f}_{\mathrm{r2}} \equiv {}^{\mathrm{o}}\boldsymbol{f}_{\mathrm{r}}^{\mathrm{A,B}} &= \tfrac{1}{2}{}^{\mathrm{o}}\boldsymbol{f}_{\mathrm{b1}} + \tfrac{1}{2}{}^{\mathrm{o}}\boldsymbol{f}_{\mathrm{b2}} - \tfrac{1}{2}{}^{\mathrm{o}}\boldsymbol{f}_{\mathrm{b3}} - \tfrac{1}{2}{}^{\mathrm{o}}\boldsymbol{f}_{\mathrm{b4}} \\
{}^{\mathrm{o}}\boldsymbol{f}_{\mathrm{r3}} \equiv {}^{\mathrm{o}}\boldsymbol{f}_{\mathrm{r}}^{\mathrm{C,D}} &= \tfrac{1}{2}{}^{\mathrm{o}}\boldsymbol{f}_{\mathrm{b5}} + \tfrac{1}{2}{}^{\mathrm{o}}\boldsymbol{f}_{\mathrm{b6}} - \tfrac{1}{2}{}^{\mathrm{o}}\boldsymbol{f}_{\mathrm{b7}} - \tfrac{1}{2}{}^{\mathrm{o}}\boldsymbol{f}_{\mathrm{b8}} \\
{}^{\mathrm{o}}\boldsymbol{f}_{\mathrm{r4}} \equiv {}^{\mathrm{o}}\boldsymbol{f}_{\mathrm{r}}^{\mathrm{A}} &= \tfrac{1}{2}{}^{\mathrm{o}}\boldsymbol{f}_{\mathrm{b1}} - \tfrac{1}{2}{}^{\mathrm{o}}\boldsymbol{f}_{\mathrm{b2}} \\
{}^{\mathrm{o}}\boldsymbol{f}_{\mathrm{r5}} \equiv {}^{\mathrm{o}}\boldsymbol{f}_{\mathrm{r}}^{\mathrm{B}} &= \tfrac{1}{2}{}^{\mathrm{o}}\boldsymbol{f}_{\mathrm{b3}} - \tfrac{1}{2}{}^{\mathrm{o}}\boldsymbol{f}_{\mathrm{b4}} \\
{}^{\mathrm{o}}\boldsymbol{f}_{\mathrm{r6}} \equiv {}^{\mathrm{o}}\boldsymbol{f}_{\mathrm{r}}^{\mathrm{C}} &= \tfrac{1}{2}{}^{\mathrm{o}}\boldsymbol{f}_{\mathrm{b5}} - \tfrac{1}{2}{}^{\mathrm{o}}\boldsymbol{f}_{\mathrm{b6}} \\
{}^{\mathrm{o}}\boldsymbol{f}_{\mathrm{r7}} \equiv {}^{\mathrm{o}}\boldsymbol{f}_{\mathrm{r}}^{\mathrm{D}} &= \tfrac{1}{2}{}^{\mathrm{o}}\boldsymbol{f}_{\mathrm{b7}} - \tfrac{1}{2}{}^{\mathrm{o}}\boldsymbol{f}_{\mathrm{b8}}
\end{aligned}
\tag{5.56}
$$



(a) Leader-follower type.

(b) Symmetry type.

Figure 5.10: ZMP trajectory in x direction.

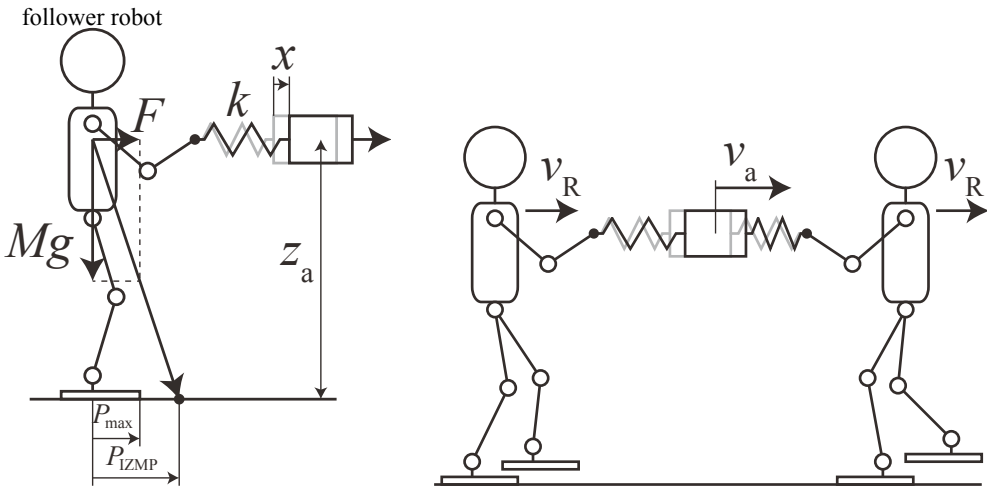
Coefficient matrix $\mathbf{C}$ of this internal force set is

$$\mathbf{C} = \begin{bmatrix} \frac{1}{2}\mathbf{I}_6 & \frac{1}{2}\mathbf{I}_6 & \frac{1}{2}\mathbf{I}_6 & \frac{1}{2}\mathbf{I}_6 & -\frac{1}{2}\mathbf{I}_6 & -\frac{1}{2}\mathbf{I}_6 & -\frac{1}{2}\mathbf{I}_6 & -\frac{1}{2}\mathbf{I}_6 \\ \frac{1}{2}\mathbf{I}_6 & \frac{1}{2}\mathbf{I}_6 & -\frac{1}{2}\mathbf{I}_6 & -\frac{1}{2}\mathbf{I}_6 & \mathbf{0}_6 & \mathbf{0}_6 & \mathbf{0}_6 & \mathbf{0}_6 \\ \mathbf{0}_6 & \mathbf{0}_6 & \mathbf{0}_6 & \mathbf{0}_6 & \frac{1}{2}\mathbf{I}_6 & \frac{1}{2}\mathbf{I}_6 & -\frac{1}{2}\mathbf{I}_6 & -\frac{1}{2}\mathbf{I}_6 \\ \frac{1}{2}\mathbf{I}_6 & -\frac{1}{2}\mathbf{I}_6 & \mathbf{0}_6 & \mathbf{0}_6 & \mathbf{0}_6 & \mathbf{0}_6 & \mathbf{0}_6 & \mathbf{0}_6 \\ \mathbf{0}_6 & \mathbf{0}_6 & \frac{1}{2}\mathbf{I}_6 & -\frac{1}{2}\mathbf{I}_6 & \mathbf{0}_6 & \mathbf{0}_6 & \mathbf{0}_6 & \mathbf{0}_6 \\ \mathbf{0}_6 & \mathbf{0}_6 & \mathbf{0}_6 & \mathbf{0}_6 & \frac{1}{2}\mathbf{I}_6 & -\frac{1}{2}\mathbf{I}_6 & \mathbf{0}_6 & \mathbf{0}_6 \\ \mathbf{0}_6 & \mathbf{0}_6 & \mathbf{0}_6 & \mathbf{0}_6 & \mathbf{0}_6 & \mathbf{0}_6 & \frac{1}{2}\mathbf{I}_6 & -\frac{1}{2}\mathbf{I}_6 \end{bmatrix}. \tag{5.57}$$

Matrix $\mathbf{V}$ is then determined from (5.12) as

$$\mathbf{V} = \begin{bmatrix} \frac{1}{4}\mathbf{I}_6 & \frac{1}{2}\mathbf{I}_6 & \mathbf{0}_6 & \mathbf{I}_6 & \mathbf{0}_6 & \mathbf{0}_6 & \mathbf{0}_6 \\ \frac{1}{4}\mathbf{I}_6 & \frac{1}{2}\mathbf{I}_6 & \mathbf{0}_6 & -\mathbf{I}_6 & \mathbf{0}_6 & \mathbf{0}_6 & \mathbf{0}_6 \\ \frac{1}{4}\mathbf{I}_6 & -\frac{1}{2}\mathbf{I}_6 & \mathbf{0}_6 & \mathbf{0}_6 & \mathbf{I}_6 & \mathbf{0}_6 & \mathbf{0}_6 \\ \frac{1}{4}\mathbf{I}_6 & -\frac{1}{2}\mathbf{I}_6 & \mathbf{0}_6 & \mathbf{0}_6 & -\mathbf{I}_6 & \mathbf{0}_6 & \mathbf{0}_6 \\ -\frac{1}{4}\mathbf{I}_6 & \mathbf{0}_6 & \frac{1}{2}\mathbf{I}_6 & \mathbf{0}_6 & \mathbf{0}_6 & \mathbf{I}_6 & \mathbf{0}_6 \\ -\frac{1}{4}\mathbf{I}_6 & \mathbf{0}_6 & \frac{1}{2}\mathbf{I}_6 & \mathbf{0}_6 & \mathbf{0}_6 & -\mathbf{I}_6 & \mathbf{0}_6 \\ -\frac{1}{4}\mathbf{I}_6 & \mathbf{0}_6 & -\frac{1}{2}\mathbf{I}_6 & \mathbf{0}_6 & \mathbf{0}_6 & \mathbf{0}_6 & \mathbf{I}_6 \\ -\frac{1}{4}\mathbf{I}_6 & \mathbf{0}_6 & -\frac{1}{2}\mathbf{I}_6 & \mathbf{0}_6 & \mathbf{0}_6 & \mathbf{0}_6 & -\mathbf{I}_6 \end{bmatrix}. \tag{5.58}$$

Each internal force is shown in Figure 5.12, and their physical meanings are presented



(a) Limitation of leader-follower type. (b) Limitation of symmetry type.

Figure 5.11: Limitation of moving velocity.

force	physical meanings
${}^{\mathrm{o}}\boldsymbol{f}_{\mathrm{r1}}$	The internal force between robot AB and CD
${}^{\mathrm{o}}\boldsymbol{f}_{\mathrm{r2}}$	The internal force between robot A and B
${}^{\mathrm{o}}\boldsymbol{f}_{\mathrm{r3}}$	The internal force between robot C and D
${}^{\mathrm{o}}\boldsymbol{f}_{\mathrm{r4}}$	The internal force between two hands of robot A
${}^{\mathrm{o}}\boldsymbol{f}_{\mathrm{r5}}$	The internal force between two hands of robot B
${}^{\mathrm{o}}\boldsymbol{f}_{\mathrm{r6}}$	The internal force between two hands of robot C
${}^{\mathrm{o}}\boldsymbol{f}_{\mathrm{r7}}$	The internal force between two hands of robot D

Table 5.1: Internal forces and their physical meanings

in Table 5.1. The workspace position vectors ${}^{\mathrm{o}}\boldsymbol{z}$ can be derived from (5.35) as follows.

$$
\begin{aligned}
{}^{\mathrm{o}}\boldsymbol{p}_{\mathrm{a}} &= \tfrac{1}{8}{}^{\mathrm{o}}\boldsymbol{p}_{\mathrm{b1}} + \tfrac{1}{8}{}^{\mathrm{o}}\boldsymbol{p}_{\mathrm{b2}} + \tfrac{1}{8}{}^{\mathrm{o}}\boldsymbol{p}_{\mathrm{b3}} + \tfrac{1}{8}{}^{\mathrm{o}}\boldsymbol{p}_{\mathrm{b4}} + \tfrac{1}{8}{}^{\mathrm{o}}\boldsymbol{p}_{\mathrm{b5}} + \tfrac{1}{8}{}^{\mathrm{o}}\boldsymbol{p}_{\mathrm{b6}} + \tfrac{1}{8}{}^{\mathrm{o}}\boldsymbol{p}_{\mathrm{b7}} + \tfrac{1}{8}{}^{\mathrm{o}}\boldsymbol{p}_{\mathrm{b8}} \ , \\
{}^{\mathrm{o}}\Delta\boldsymbol{p}_{\mathrm{r1}} &= \tfrac{1}{4}{}^{\mathrm{o}}\boldsymbol{p}_{\mathrm{b1}} + \tfrac{1}{4}{}^{\mathrm{o}}\boldsymbol{p}_{\mathrm{b2}} + \tfrac{1}{4}{}^{\mathrm{o}}\boldsymbol{p}_{\mathrm{b3}} + \tfrac{1}{4}{}^{\mathrm{o}}\boldsymbol{p}_{\mathrm{b4}} - \tfrac{1}{4}{}^{\mathrm{o}}\boldsymbol{p}_{\mathrm{b5}} - \tfrac{1}{4}{}^{\mathrm{o}}\boldsymbol{p}_{\mathrm{b6}} - \tfrac{1}{4}{}^{\mathrm{o}}\boldsymbol{p}_{\mathrm{b7}} - \tfrac{1}{4}{}^{\mathrm{o}}\boldsymbol{p}_{\mathrm{b8}} \ , \\
{}^{\mathrm{o}}\Delta\boldsymbol{p}_{\mathrm{r2}} &= \tfrac{1}{2}{}^{\mathrm{o}}\boldsymbol{p}_{\mathrm{b1}} + \tfrac{1}{2}{}^{\mathrm{o}}\boldsymbol{p}_{\mathrm{b2}} - \tfrac{1}{2}{}^{\mathrm{o}}\boldsymbol{p}_{\mathrm{b3}} - \tfrac{1}{2}{}^{\mathrm{o}}\boldsymbol{p}_{\mathrm{b4}} \ , \\
{}^{\mathrm{o}}\Delta\boldsymbol{p}_{\mathrm{r3}} &= \tfrac{1}{2}{}^{\mathrm{o}}\boldsymbol{p}_{\mathrm{b5}} + \tfrac{1}{2}{}^{\mathrm{o}}\boldsymbol{p}_{\mathrm{b6}} - \tfrac{1}{2}{}^{\mathrm{o}}\boldsymbol{p}_{\mathrm{b7}} - \tfrac{1}{2}{}^{\mathrm{o}}\boldsymbol{p}_{\mathrm{b7}} \ , \\
{}^{\mathrm{o}}\Delta\boldsymbol{p}_{\mathrm{r4}} &= {}^{\mathrm{o}}\boldsymbol{p}_{\mathrm{b1}} - {}^{\mathrm{o}}\boldsymbol{p}_{\mathrm{b2}} \ , \\
{}^{\mathrm{o}}\Delta\boldsymbol{p}_{\mathrm{r5}} &= {}^{\mathrm{o}}\boldsymbol{p}_{\mathrm{b3}} - {}^{\mathrm{o}}\boldsymbol{p}_{\mathrm{b4}} \ , \\
{}^{\mathrm{o}}\Delta\boldsymbol{p}_{\mathrm{r6}} &= {}^{\mathrm{o}}\boldsymbol{p}_{\mathrm{b5}} - {}^{\mathrm{o}}\boldsymbol{p}_{\mathrm{b6}} \ , \\
{}^{\mathrm{o}}\Delta\boldsymbol{p}_{\mathrm{r7}} &= {}^{\mathrm{o}}\boldsymbol{p}_{\mathrm{b7}} - {}^{\mathrm{o}}\boldsymbol{p}_{\mathrm{b8}} \ .
\end{aligned}
\tag{5.59}
$$

The selection matrix $\mathbf{S}$ is set as $\mathrm{diag}(\mathbf{0}_6, \mathbf{I}_6, \mathbf{I}_6, \mathbf{I}_6, \mathbf{I}_6, \mathbf{I}_6, \mathbf{I}_6, \mathbf{I}_6)$, so that the system controlled the object position and internal forces in the simulation.

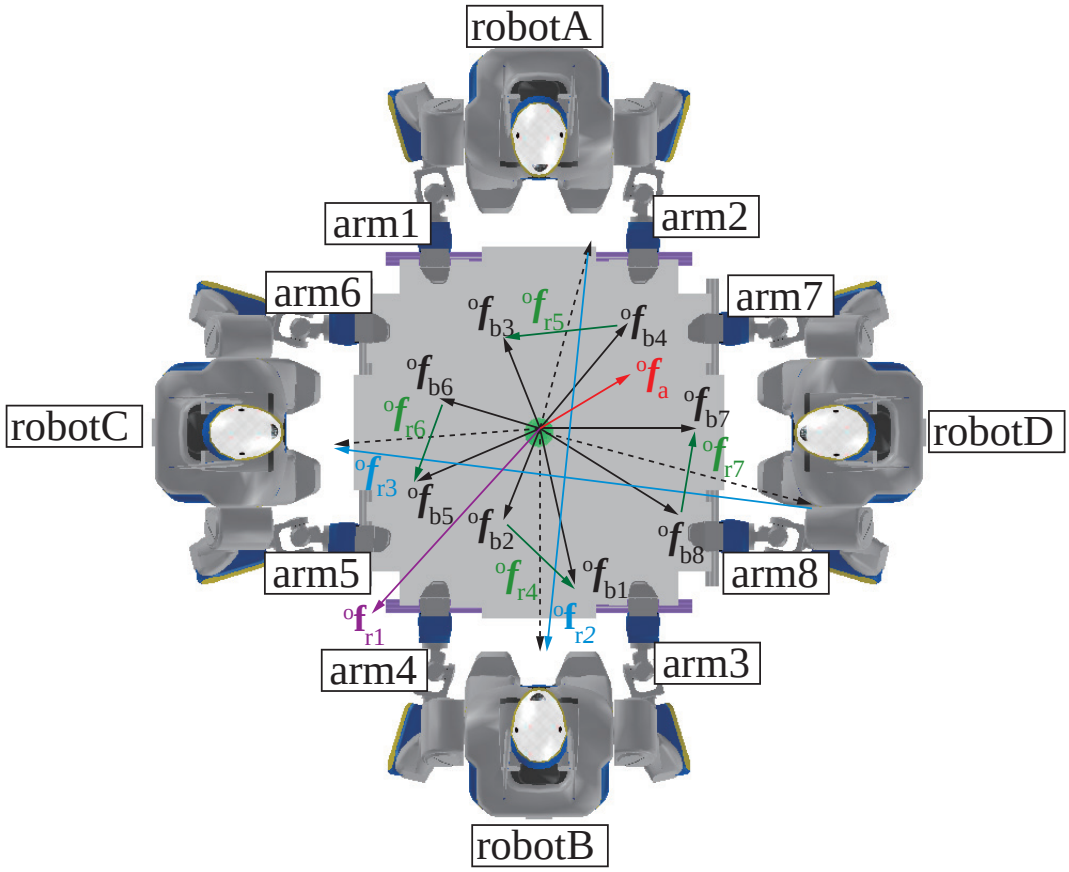


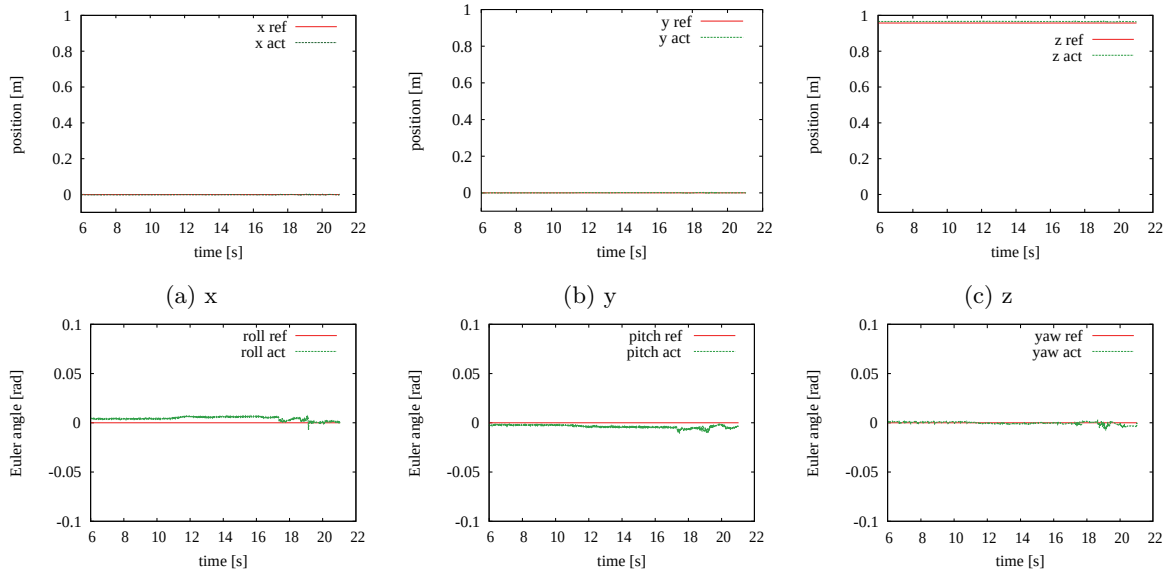
Figure 5.12: All force vectors.

5.5.3 Validation of Internal Force Control without Stepping

In the first simulation, the internal force control without stepping is validated. All four robots stood facing an object and held it as illustrated in Figure 5.12. Because the robots were not fixed to the floor, the excessive internal force reference may not have been achieved because of the slippage between the feet and floor. In this simulation, all the references of the internal forces were set to zero. The object position was controlled to stay at its initial position.

The procedure of this simulation was as follows.

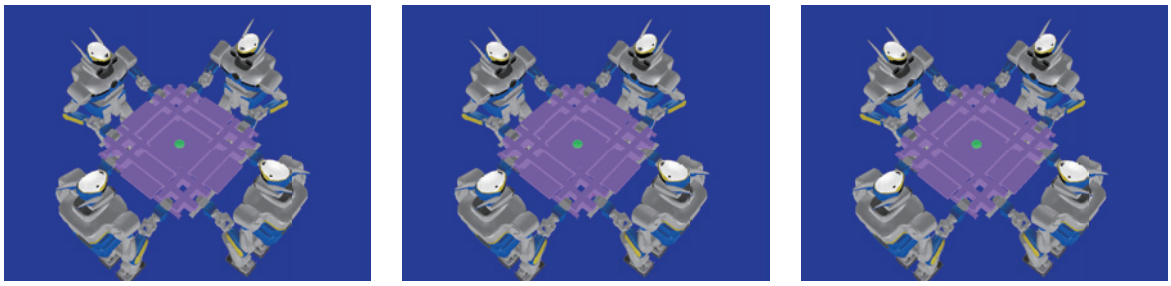
1. Four robots grasped the object.
2. Virtual sticks, as shown in Figure 5.2, were created.
3. The reference position of the object was set to the initial position.
4. The object position and internal force hybrid control were started at a specific time.



(a) x (b) y (c) z

(d) roll (e) pitch (e) yaw

Figure 5.13: External position of the object.



(a) $t = 6$ s (b) $t = 17$ s (c) $t = 21$ s

Figure 5.14: Snapshots in the hybrid control simulation.

The external position history and its reference are shown in Figure 5.13, and snapshots of the simulation are shown in Figure 5.14. At first, the robots held the object, and the motor command was kept constant. At this point, the internal force of each direction might not have equaled zero. The hybrid control for object position and internal force was started at time $t = 10.5$ s. After the hybrid control was started, all internal forces ${}^{\mathrm{a}}\boldsymbol{f}_{\mathrm{r}1}, {}^{\mathrm{a}}\boldsymbol{f}_{\mathrm{r}2}, \ldots {}^{\mathrm{a}}\boldsymbol{f}_{\mathrm{r}7}$ converged to zero (Figure 5.15).

In these graphs, the suffix "ref" indicates the reference value, and "act" is the actual value obtained from the dynamic simulator. The reference positions in the x and y directions and the reference Euler angles were set to zero in this simulation. Although there was some oscillation in the attitude, its amplitude was small enough to ignore. Both position and attitude closely follow the reference. The proposed hybrid control is hence validated.

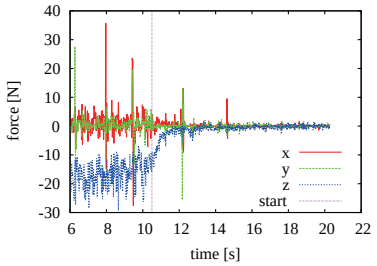
5.5.4 Cooperative object transportation Simulation

The second simulation involved a walking motion and 6-dimensional transportation of the object. The procedure of this simulation was as follows.

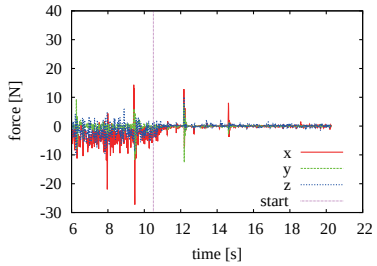
1. Four robots grasped the object.
2. The virtual sticks were created, as shown in Figure 5.2.
3. The reference position of the object was set at the initial position.
4. The external position and internal force hybrid control were started.
5. A reference velocity was assigned to the center of the object, and the velocity was randomly changed during the simulation.
6. After a few steps, the reference velocity of the object was set to zero in all directions.

The first four steps are the same as the previous simulation. In this simulation, the reference position was computed by integrating the reference velocity. As plotted in Figure 5.16, the actual position and attitude of the object closely tracked the references. Figure 5.17 shows a top view of the robot motion in the simulation. Figure 5.18 shows the robot motion that moved the object in the six axes.

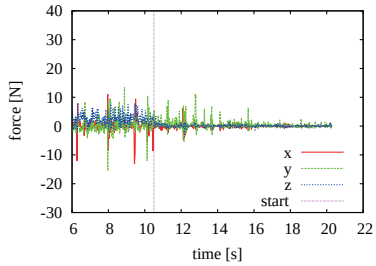
Figure 5.19 shows the internal force. Although some spikes can be observed at the moments when a foot landed, the internal forces converged to zero.



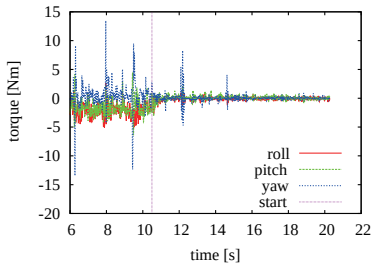
(a) Force of ${}^{\mathrm{a}}\boldsymbol{f}_{\mathrm{r}1}$.



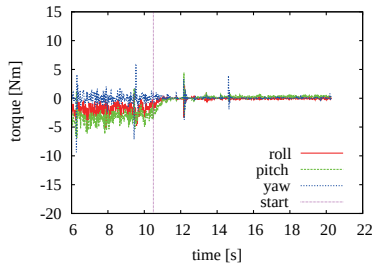
(b) Force of ${}^{\mathrm{a}}\boldsymbol{f}_{\mathrm{r}2}$.



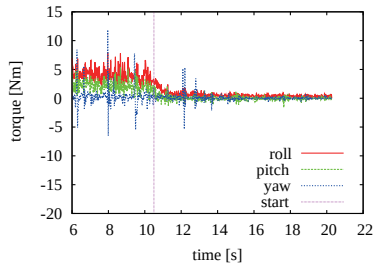
(c) Force of ${}^{\mathrm{a}}\boldsymbol{f}_{\mathrm{r}3}$.



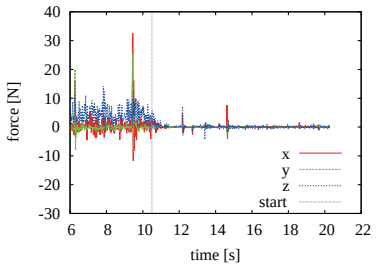
(d) Moment of ${}^{\mathrm{a}}\boldsymbol{f}_{\mathrm{r}1}$.



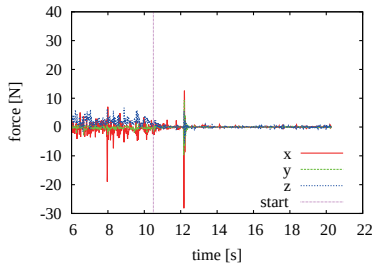
(e) Moment of ${}^{\mathrm{a}}\boldsymbol{f}_{\mathrm{r}2}$.



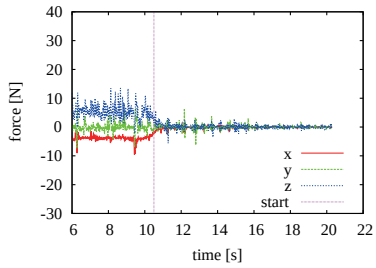
(f) Moment of ${}^{\mathrm{a}}\boldsymbol{f}_{\mathrm{r}3}$.



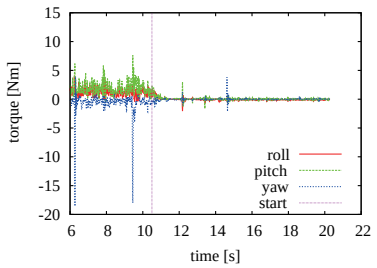
(g) Force of ${}^{\mathrm{a}}\boldsymbol{f}_{\mathrm{r}4}$.



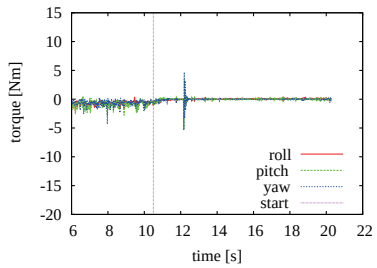
(h) Force of ${}^{\mathrm{a}}\boldsymbol{f}_{\mathrm{r}5}$.



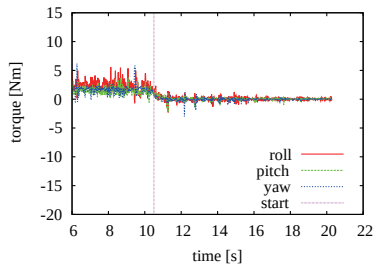
(i) Force of ${}^{\mathrm{a}}\boldsymbol{f}_{\mathrm{r}6}$.



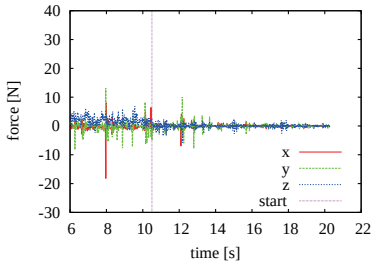
(j) Moment of ${}^{\mathrm{a}}\boldsymbol{f}_{\mathrm{r}4}$.



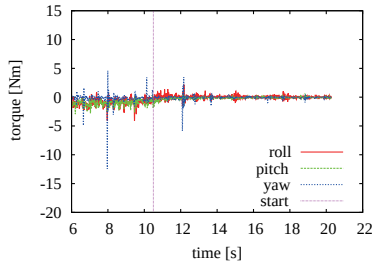
(k) Moment of ${}^{\mathrm{a}}\boldsymbol{f}_{\mathrm{r}5}$.



(l) Moment of ${}^{\mathrm{a}}\boldsymbol{f}_{\mathrm{r}6}$.

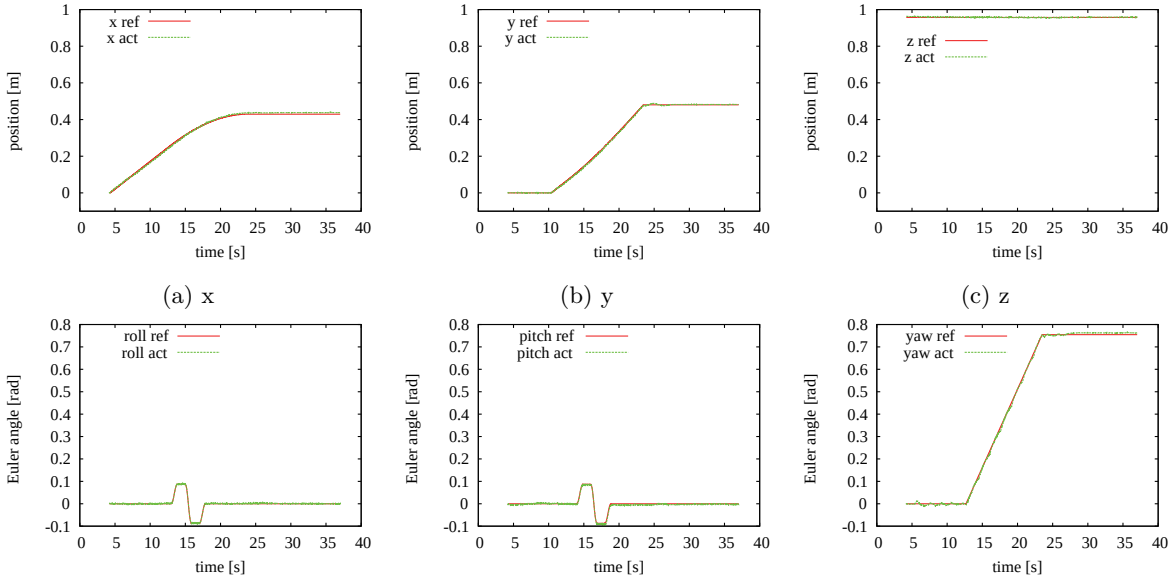


(m) Force of ${}^{\mathrm{a}}\boldsymbol{f}_{\mathrm{r}7}$.



(n) Moment of ${}^{\mathrm{a}}\boldsymbol{f}_{\mathrm{r}7}$.

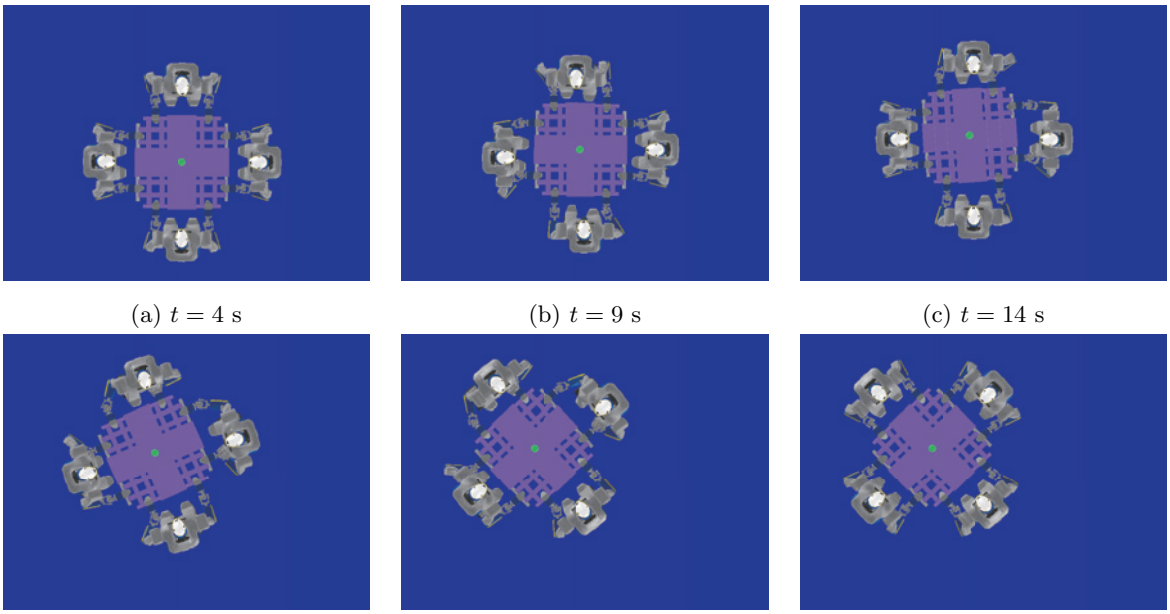
Figure 5.15: The history of internal forces.



(a) x (b) y (c) z

(d) roll (d) pitch (f) yaw

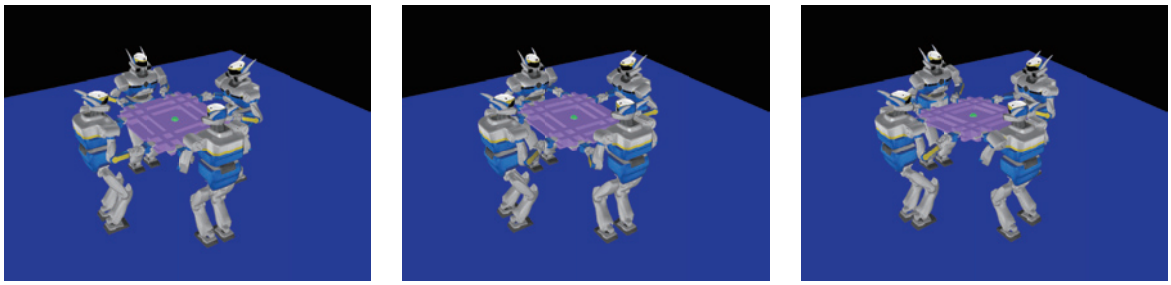
Figure 5.16: External position of the object.



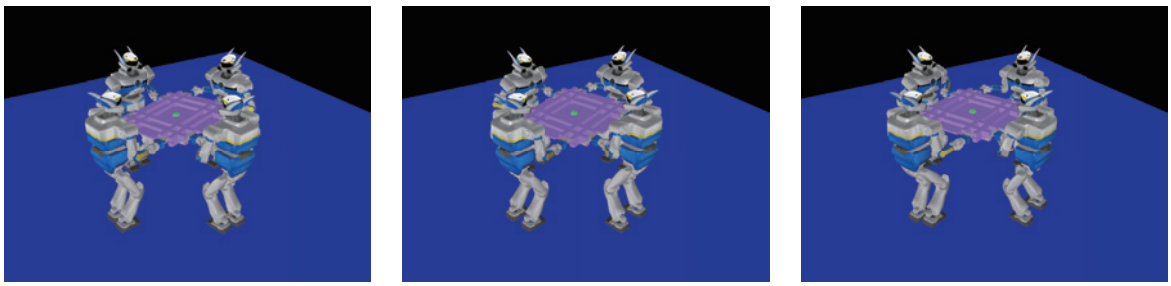
(a) $t = 4$ s (b) $t = 9$ s (c) $t = 14$ s

(a) $t = 19$ s (b) $t = 24$ s (c) $t = 29$ s

Figure 5.17: Sequential photo graphs of simulation (top view).

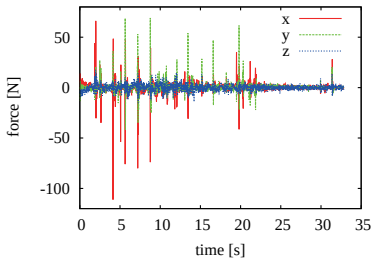


(a) $t = 14$ s (b) $t = 15$ s (c) $t = 16$ s

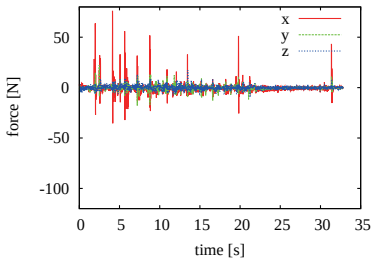


(a) $t = 17$ s (b) $t = 18$ s (c) $t = 19$ s

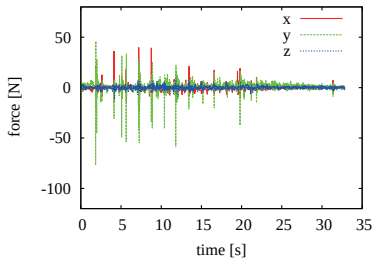
Figure 5.18: Sequential photo graphs of simulation.



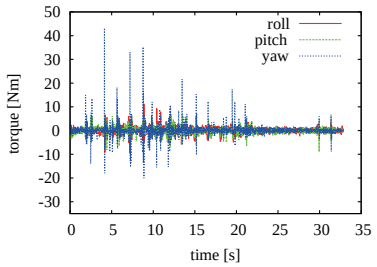
(a) Force of ${}^{\mathrm{a}}\boldsymbol{f}_{\mathrm{r1}}$.



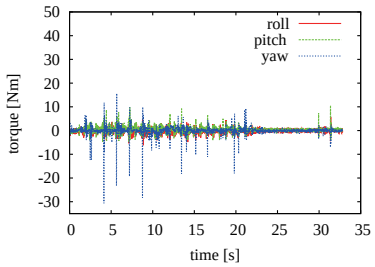
(b) Force of ${}^{\mathrm{a}}\boldsymbol{f}_{\mathrm{r2}}$.



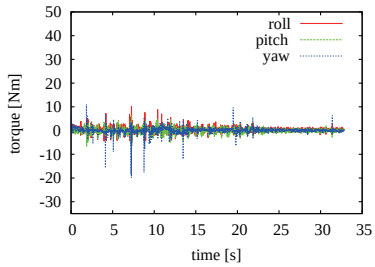
(c) Force of ${}^{\mathrm{a}}\boldsymbol{f}_{\mathrm{r3}}$.



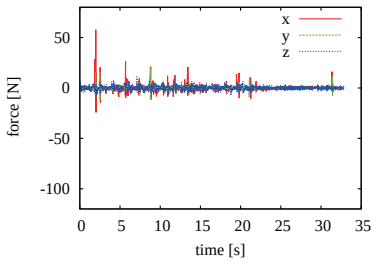
(d) Moment of ${}^{\mathrm{a}}\boldsymbol{f}_{\mathrm{r1}}$.



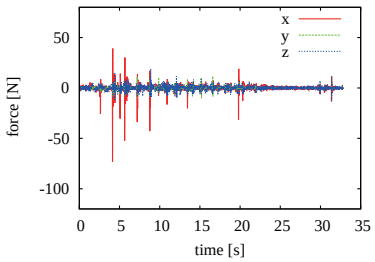
(e) Moment of ${}^{\mathrm{a}}\boldsymbol{f}_{\mathrm{r2}}$.



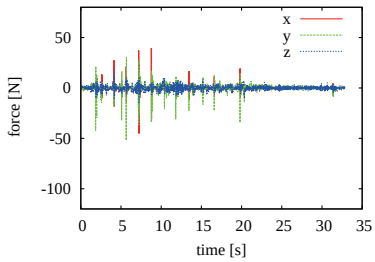
(f) Moment of ${}^{\mathrm{a}}\boldsymbol{f}_{\mathrm{r3}}$.



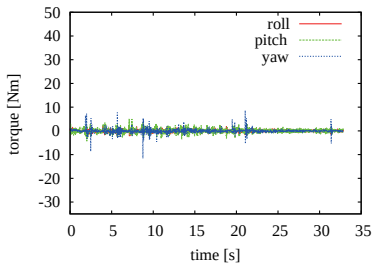
(g) Force of ${}^{\mathrm{a}}\boldsymbol{f}_{\mathrm{r4}}$.



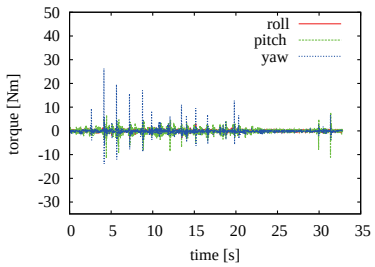
(h) Force of ${}^{\mathrm{a}}\boldsymbol{f}_{\mathrm{r5}}$.



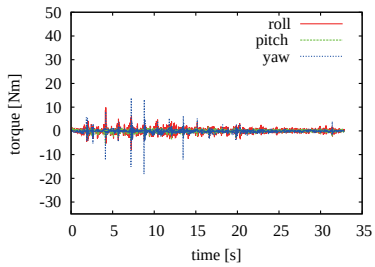
(i) Force of ${}^{\mathrm{a}}\boldsymbol{f}_{\mathrm{r6}}$.



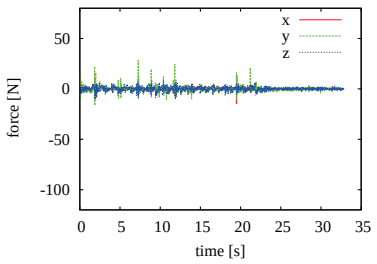
(j) Moment of ${}^{\mathrm{a}}\boldsymbol{f}_{\mathrm{r4}}$.



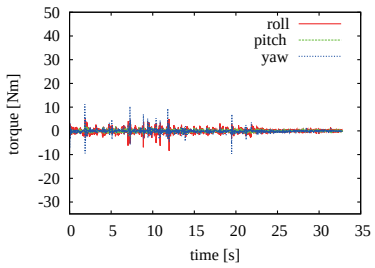
(k) Moment of ${}^{\mathrm{a}}\boldsymbol{f}_{\mathrm{r5}}$.



(l) Moment of ${}^{\mathrm{a}}\boldsymbol{f}_{\mathrm{r6}}$.



(m) Force of ${}^{\mathrm{a}}\boldsymbol{f}_{\mathrm{r7}}$.



(n) Moment of ${}^{\mathrm{a}}\boldsymbol{f}_{\mathrm{r7}}$.

Figure 5.19: The history of internal forces.

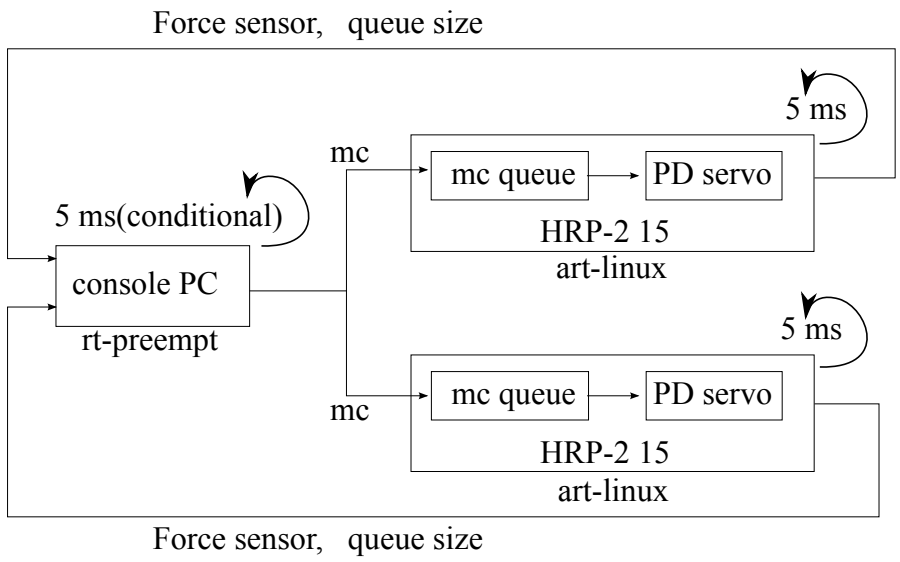


Figure 5.20: Overview of experiment system.

5.6 Experiment of Symmetry Type Cooperative Object Transportation

The proposed method is verified with dynamic simulation in last section. The method is verified with two humanoid robots HRP-2 (four robotic arms in total) in this section.

5.6.1 Framework of Experiment System

The framework of experiment system is built as illustrated in Figure 5.20. A external computer (console PC) is used to calculated and transfer the reference joint angles of the two robot simultaneously. This system is consisted of three computers:

1. Console PC (rt-preempt)
2. HRP-2 20 PC (art-linux)
3. HRP-2 15 PC (art-linux).

The two HRP-2 robots are running in about 5 [ms] real time process, but there is a tiny difference between this two robots. This difference is accumulated by times, and becomes a large time lag between the two robots. Since there is no synchronization between this two robots, and since the console PC is faster than the two robots PC, it is possible to use the console PC to synchronize them by adjusting its computing period.

A mc (motor command) queue (a reference joint angles sequence) is built in the robot, motor command is stocked here before it is executed. PD servo is a PD control algorithm of all joint angles. The queue size is set as 5, i. e. it is a 25 [ms] trajectory. Therefore

the motor commend unreaching time is allowed within 25 [ms]. The current queue size is sent to console PC every time after one motor command is executed. When the queue size of both robots go over 5, the console PC sleeps 5 [ms] after one motor command is computed and transferred to the two robots. And when the queue size is differ from another one by 3, the motor command will not be sent to the slower one. Although all the data is transferred by socket through a LAN cable, a approximate synchronization system can be achieved.

5.6.2 Internal Force Control Experiment without Stepping

HRP-2 NO.20 is treated as robot A, and HRP-2 NO.15 is treated as robot B as shown in Figure 5.21. the following internal force set is chosen in this experiment.

$$
\begin{aligned}
{}^{\mathrm{o}}\boldsymbol{f}_{\mathrm{r1}} \equiv {}^{\mathrm{o}}\boldsymbol{f}_{\mathrm{r}}^{\mathrm{A,B}} &= \tfrac{1}{2}{}^{\mathrm{o}}\boldsymbol{f}_{\mathrm{b1}} + \tfrac{1}{2}{}^{\mathrm{o}}\boldsymbol{f}_{\mathrm{b2}} - \tfrac{1}{2}{}^{\mathrm{o}}\boldsymbol{f}_{\mathrm{b3}} - \tfrac{1}{2}{}^{\mathrm{o}}\boldsymbol{f}_{\mathrm{b4}} && (5.60)\\
{}^{\mathrm{o}}\boldsymbol{f}_{\mathrm{r2}} \equiv {}^{\mathrm{o}}\boldsymbol{f}_{\mathrm{r}}^{\mathrm{A}} &= \tfrac{1}{2}{}^{\mathrm{o}}\boldsymbol{f}_{\mathrm{b1}} - \tfrac{1}{2}{}^{\mathrm{o}}\boldsymbol{f}_{\mathrm{b2}} && (5.61)\\
{}^{\mathrm{o}}\boldsymbol{f}_{\mathrm{r3}} \equiv {}^{\mathrm{o}}\boldsymbol{f}_{\mathrm{r}}^{\mathrm{B}} &= \tfrac{1}{2}{}^{\mathrm{o}}\boldsymbol{f}_{\mathrm{b3}} - \tfrac{1}{2}{}^{\mathrm{o}}\boldsymbol{f}_{\mathrm{b4}} && (5.62)\\
& && (5.63)
\end{aligned}
$$

Coefficients matrix $\mathbf{C}$ of this internal force set is

$$
\mathbf{C} = \begin{bmatrix} \frac{1}{2}\mathbf{I}_6 & \frac{1}{2}\mathbf{I}_6 & -\frac{1}{2}\mathbf{I}_6 & -\frac{1}{2}\mathbf{I}_6 \\ \frac{1}{2}\mathbf{I}_6 & -\frac{1}{2}\mathbf{I}_6 & \mathbf{0}_6 & \mathbf{0}_6 \\ \mathbf{0}_6 & \mathbf{0}_6 & \frac{1}{2}\mathbf{I}_6 & -\frac{1}{2}\mathbf{I}_6 \end{bmatrix} . \tag{5.64}
$$

Then $\mathbf{V}$ is determined from (5.12) as

$$
\mathbf{V} = \begin{bmatrix} \frac{1}{2}\mathbf{I}_6 & \mathbf{I}_6 & \mathbf{0}_6 \\ \frac{1}{2}\mathbf{I}_6 & -\mathbf{I}_6 & \mathbf{0}_6 \\ -\frac{1}{2}\mathbf{I}_6 & \mathbf{0}_6 & \mathbf{I}_6 \\ -\frac{1}{2}\mathbf{I}_6 & \mathbf{0}_6 & -\mathbf{I}_6 \end{bmatrix} . \tag{5.65}
$$

The workspace position vectors ${}^{\mathrm{o}}\boldsymbol{z}$ can be derived from (5.35) as follows.

$$
\begin{aligned}
{}^{\mathrm{o}}\boldsymbol{p}_{\mathrm{a}} &= \tfrac{1}{4}{}^{\mathrm{o}}\boldsymbol{p}_{\mathrm{b1}} + \tfrac{1}{4}{}^{\mathrm{o}}\boldsymbol{p}_{\mathrm{b2}} + \tfrac{1}{4}{}^{\mathrm{o}}\boldsymbol{p}_{\mathrm{b3}} + \tfrac{1}{4}{}^{\mathrm{o}}\boldsymbol{p}_{\mathrm{b4}} \ ,\\
{}^{\mathrm{o}}\Delta\boldsymbol{p}_{\mathrm{r1}} &= \tfrac{1}{2}{}^{\mathrm{o}}\boldsymbol{p}_{\mathrm{b1}} + \tfrac{1}{2}{}^{\mathrm{o}}\boldsymbol{p}_{\mathrm{b2}} - \tfrac{1}{2}{}^{\mathrm{o}}\boldsymbol{p}_{\mathrm{b3}} - \tfrac{1}{2}{}^{\mathrm{o}}\boldsymbol{p}_{\mathrm{b4}} \ ,\\
{}^{\mathrm{o}}\Delta\boldsymbol{p}_{\mathrm{r2}} &= {}^{\mathrm{o}}\boldsymbol{p}_{\mathrm{b1}} - {}^{\mathrm{o}}\boldsymbol{p}_{\mathrm{b2}} \ ,\\
{}^{\mathrm{o}}\Delta\boldsymbol{p}_{\mathrm{r3}} &= {}^{\mathrm{o}}\boldsymbol{p}_{\mathrm{b3}} - {}^{\mathrm{o}}\boldsymbol{p}_{\mathrm{b4}} \ , && (5.66)
\end{aligned}
$$

The selection matrix $\mathbf{S}$ is set as $\mathrm{diag}(\mathbf{0}_6, \mathbf{I}_6, \mathbf{I}_6, \mathbf{I}_6)$, so that the system controls the object position and internal forces in the experiment.

In the first experiment, the internal force control is validated without stepping. In this experiment, a specific reference value of ${}^{\mathrm{o}}\boldsymbol{f}_{\mathrm{r1}}$ is given in x and roll direction in order. The references value of other internal forces are set to 0. The object position is controlled to stay at the initial position.

The procedure of this experiment is as follows.

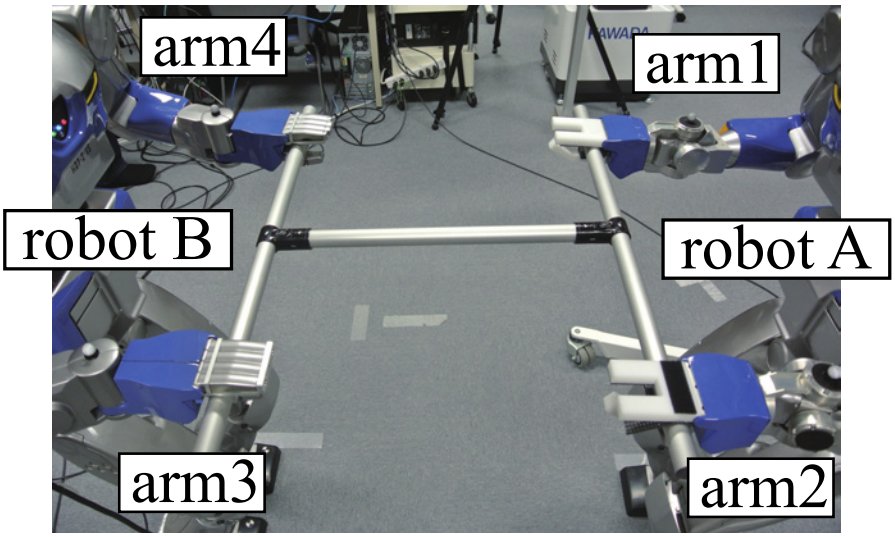


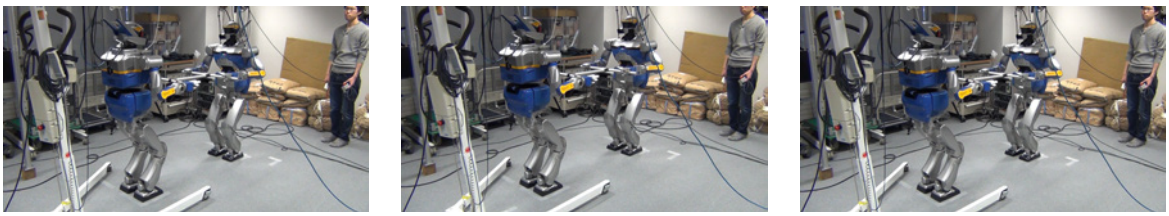
Figure 5.21: Definition of robot A, robot B, robotic arms assignment.

1. Two robots grasp the object.
2. Capture the hand wrist positions of two robots.
3. Create the virtual sticks as shown in Figure 5.21.
4. Offset the current internal forces to 0.
5. Set the reference position of the object at the initial position.
6. Start the object position and internal force hybrid control.
7. Set reference value of ${}^{\mathrm{o}}\boldsymbol{f}_{\mathrm{r1}}$ in x direction, then set in roll direction in order.
8. Set reference value of all internal forces as 0.

Optical tracking system [44] is used to capture the initial hand wrist positions in world coordinate. The snapshots of the experiment are shown in Figure 5.23. At first, reference value of all internal forces is set as 0. At about $t = 10$ s, only the reference value of ${}^{\mathrm{o}}\boldsymbol{f}_{\mathrm{r1}}$ in x direction is set to 3 [N]. k_{h} is adjusted online in order to well converge to reference value. After the value of ${}^{\mathrm{o}}\boldsymbol{f}_{\mathrm{r1}}$ in x direction converges to reference value, The reference value of all internal forces are set as 0. The internal forces history (t= 10 [s] to 210 [s]).) is shown in Figure 5.24. From this Figure, it is observed that ${}^{\mathrm{o}}\boldsymbol{f}_{\mathrm{r1}}$ in x direction converges to reference value 3 [N] at t= 140 [s] to 170 [s]. The other internal forces is controlled to 0.

After the tracking of ${}^{\mathrm{o}}\boldsymbol{f}_{\mathrm{r1}}$ in x direction is validated, next ${}^{\mathrm{o}}\boldsymbol{f}_{\mathrm{r1}}$ in *roll* direction is controlled. The internal forces history (t= 430 [s] to 630 [s]).) is shown in Figure 5.25. In this span, only the reference value of ${}^{\mathrm{o}}\boldsymbol{f}_{\mathrm{r1}}$ in *roll* direction is set to 2.5 [Nm], other

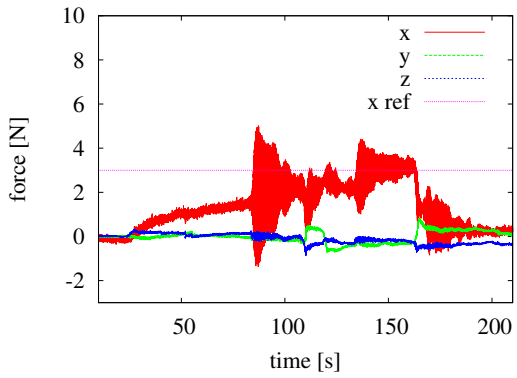
Figure 5.22: Optical tracking system.



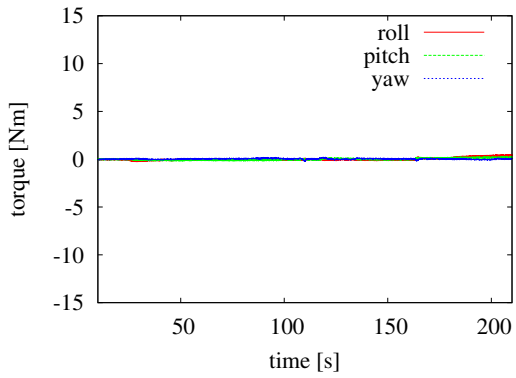
(a) t=10 [s] (b t=40 [s] (b) t=70 [s]

Figure 5.23: The snapshot of internal forces experiment.

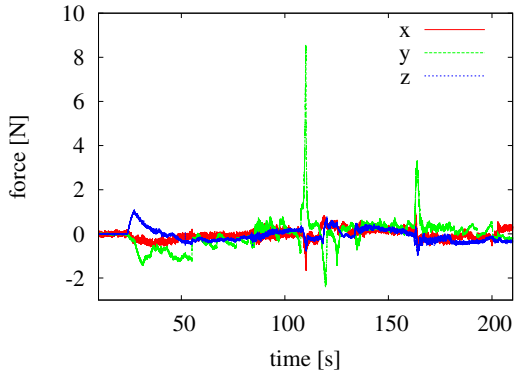
internal forces reference is set to 0. There is a huge oscillation at about $t = 470$ [s] because k_{h} gain is set improperly. After a proper k_{h} gain is set, the oscillation stops and ${}^{\mathrm{o}}\boldsymbol{f}_{\mathrm{r1}}$ in *roll* converges to reference value. Finally, reference value of all internal forces is set to 0, and it is observed that they did converge to 0 from Figure 5.25.



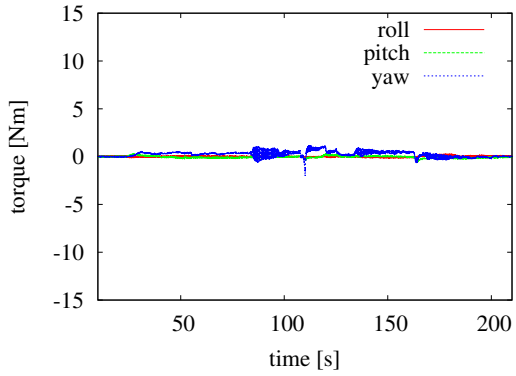
(a) Force of ${}^{a}\boldsymbol{f}_{r1}$.



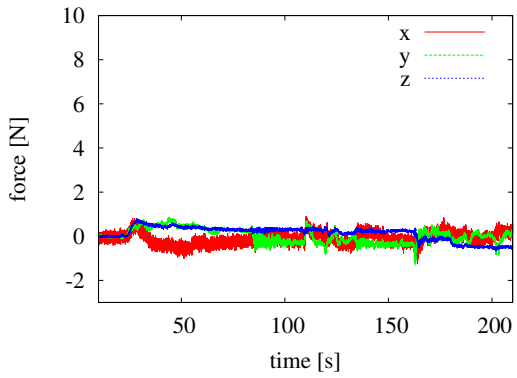
(b) Moment of ${}^{a}\boldsymbol{f}_{r1}$.



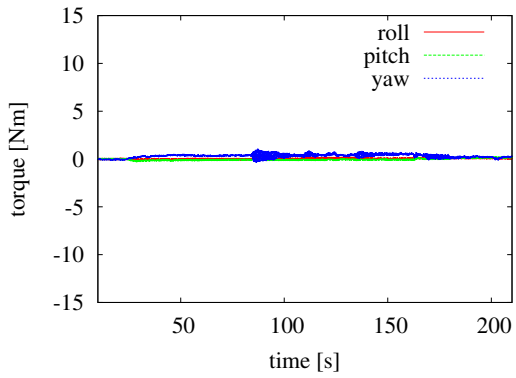
(a) Force of ${}^{a}\boldsymbol{f}_{r2}$.



(b) Moment of ${}^{a}\boldsymbol{f}_{r2}$.

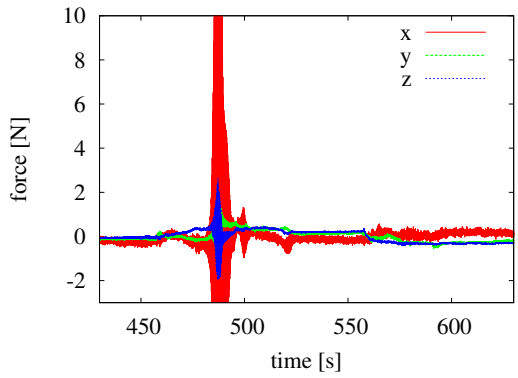


(a) Force of ${}^{a}\boldsymbol{f}_{r3}$.

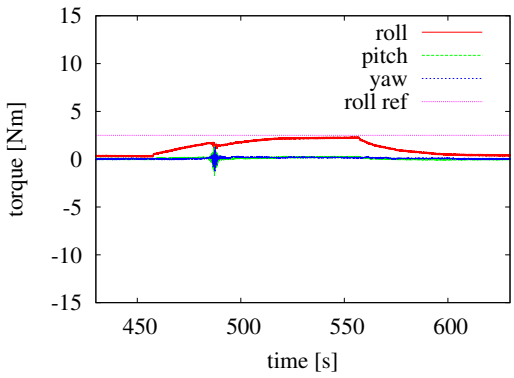


(b) Moment of ${}^{a}\boldsymbol{f}_{r3}$.

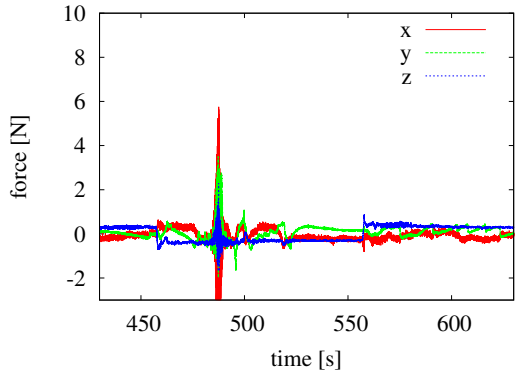
Figure 5.24: The history of internal forces(t= 10 [s] to 210 [s]).



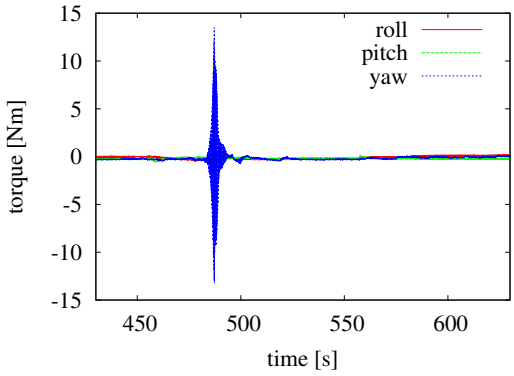
(a) Force of ${}^{a}\boldsymbol{f}_{r1}$.



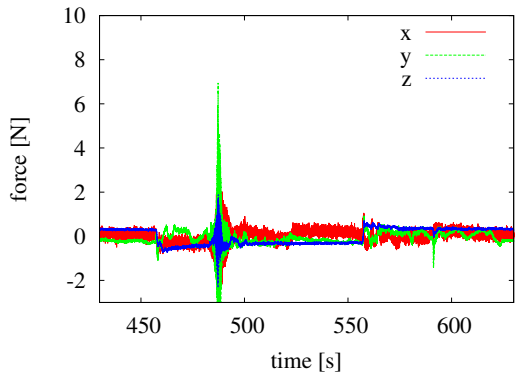
(b) Moment of ${}^{a}\boldsymbol{f}_{r1}$.



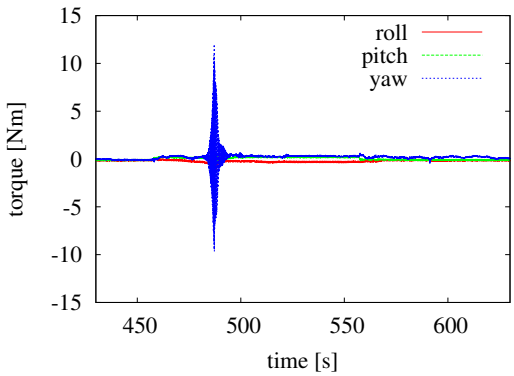
(a) Force of ${}^{a}\boldsymbol{f}_{r2}$.



(b) Moment of ${}^{a}\boldsymbol{f}_{r2}$.



(a) Force of ${}^{a}\boldsymbol{f}_{r3}$.



(b) Moment of ${}^{a}\boldsymbol{f}_{r3}$.

Figure 5.25: The history of internal forces(t= 430 [s] to 630 [s]).

5.6.3 Cooperative object transportation Experiment

The second experiment involves walking motion and transportation of the object. The procedure of this experiment is as follows.

1. Two robots grasp the object.

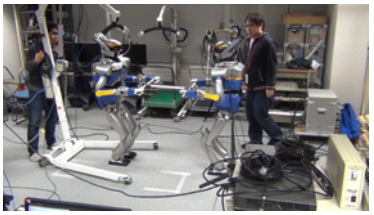
2. Capture the hand wrist positions of two robots.
3. Create the virtual sticks as shown in Figure 5.21.
4. Offset the current internal forces to 0.
5. Set the reference position of the object at the initial position.
6. Set reference value of all internal forces as 0. Start the external position and internal force hybrid control.
7. Start the stabilization control.
8. A reference constant velocity is given to the center of the object.
9. After a few walk, the reference velocity of the object is set to 0 in all direction.

The first five steps are the same as the previous experiment. In this experiment, the reference position was computed by integrating the reference velocity. During $t = 202$ $[s] \sim t = 208$ $[s]$, the reference object velocity is set to 0.03 [m/s] in x direction. Figure 5.26 shows the robot motion that moved the object. The internal forces history is shown in Figure 5.27.

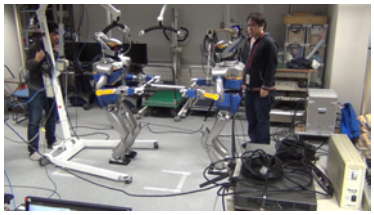
The transportation experiment with rotational movement is also conducted. During $t = 220$ $[s] \sim t = 230$ $[s]$, the reference object velocity is set to 0.03 [m/s] in x direction, and 5 [degree/s] in yaw direction. Figure 5.28 shows the robot motion, and the internal forces history is showing in Figure 5.29. The robot moves the object with walking properly. Unfortunately, the internal forces do not converge to the reference values 0. This may due to the effect of the stabilization control of HRP-2 [38]. When there is a disturbance or a external force acts at HRP-2, the stabilization control system adjust its whole body posture to recover ZMP error. In this experiment, the internal force, i. e. the external force acts at one HRP-2 is not always be zero. As a result, HRP-2 adjust its whole body joint angle, and this may influence the internal force control. And since the out put joint angles of stabilization control is vibrational, it is improper to update the calculation model with this joint angles. How to feedback this information to calculation model is one of future works.

5.7 Summary

In this chapter, A framework of symmetry cooperation for arbitrary number of robots is proposed. Dynamics simulations were performed to validate the proposed framework. The results of the dynamics simulation clearly showed the advantage of the proposed symmetry cooperation over the leader-follower cooperation. Furthermore, cooperation among four humanoid robots was taken as example, and simulations were performed. In the simulation, position and attitude of the object, and internal forces were controlled.

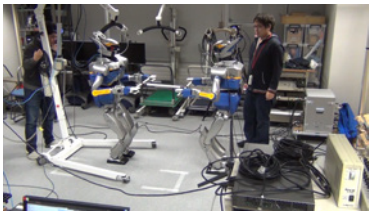


(a) t=202.5 [s]



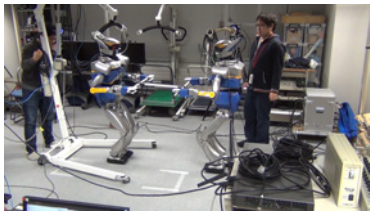
(b) t=203 [s]

(c) t=203.5 [s]

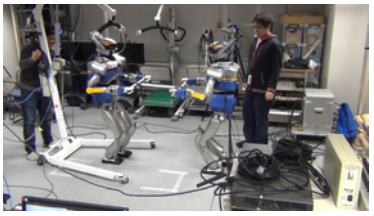


(d) t=204 [s]

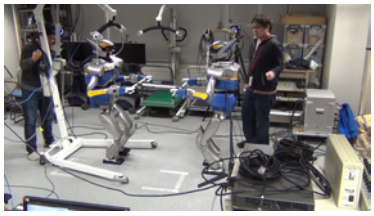
(e) t=204.5 [s]



(f) t=205 [s]

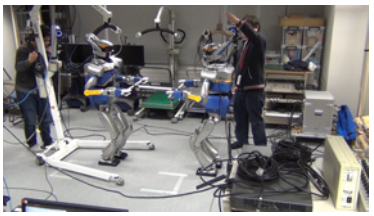


(g) t=205.5 [s]

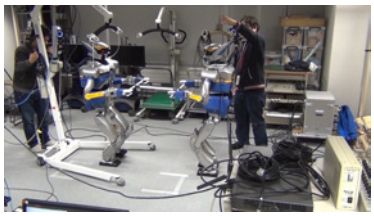


(h) t=206 [s]

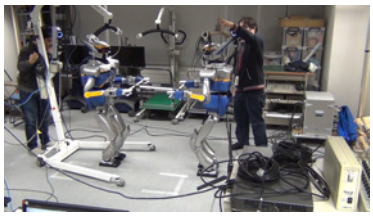
(i) t=206.5 [s]



(j) t=207 [s]



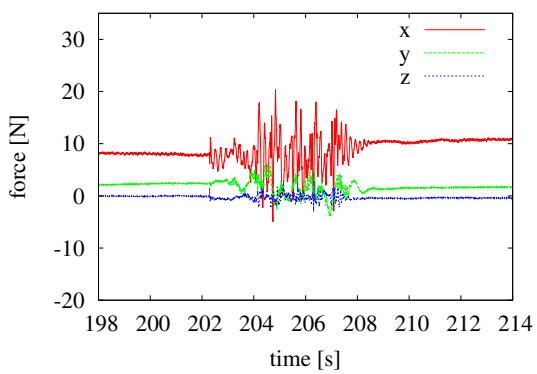
(k) t=207.5 [s]



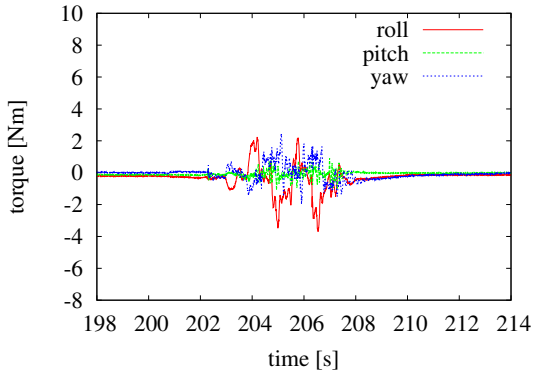
(l) t=208 [s]

Figure 5.26: Sequential photo graphs of experiment (x direction movement).

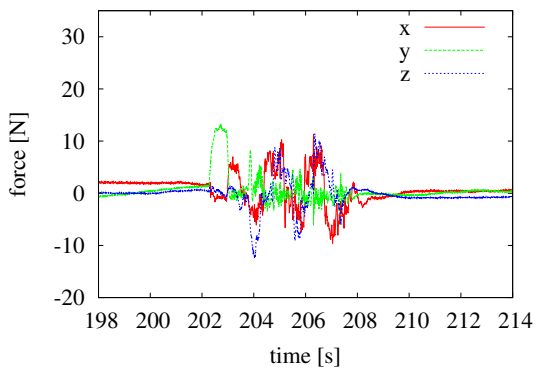
The results validated the effectiveness of the proposed hybrid controller. The proposed hybrid control is also validated by two HRP-2 robot.



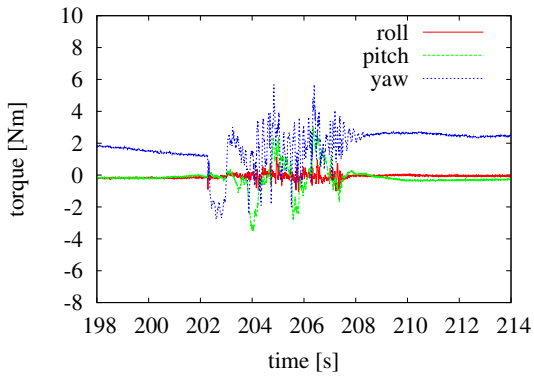
(a) Force of ${}^{a}\boldsymbol{f}_{r1}$.



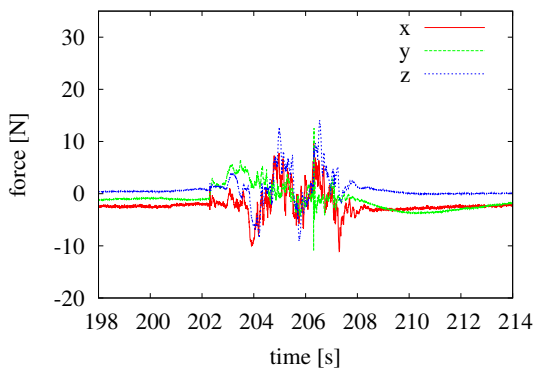
(b) Moment of ${}^{a}\boldsymbol{f}_{r1}$.



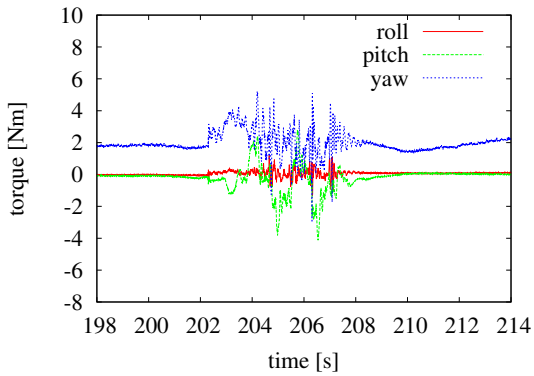
(a) Force of ${}^{a}\boldsymbol{f}_{r2}$.



(b) Moment of ${}^{a}\boldsymbol{f}_{r2}$.

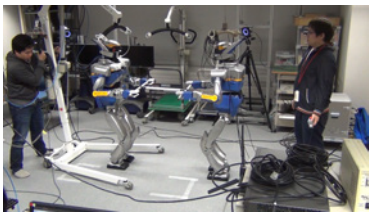


(a) Force of ${}^{a}\boldsymbol{f}_{r3}$.

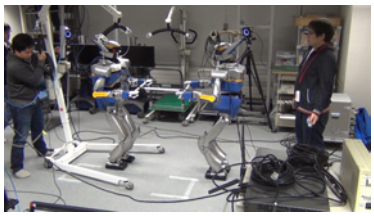


(b) Moment of ${}^{a}\boldsymbol{f}_{r3}$.

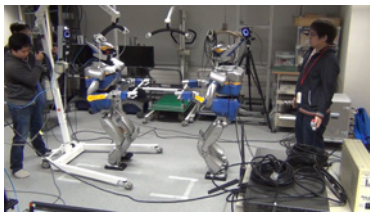
Figure 5.27: The history of internal forces(t= 198 [s] to 214 [s]).



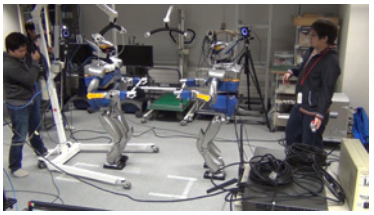
(a) t=222.5 [s]



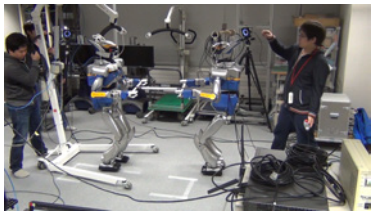
(b) t=223 [s]



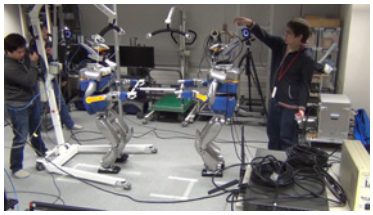
(c) t=223.5 [s]



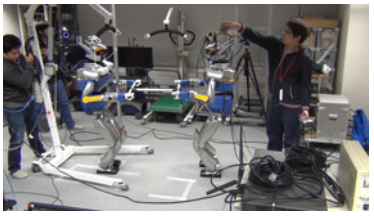
(d) t=224 [s]



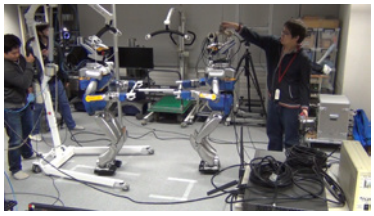
(e) t=224.5 [s]



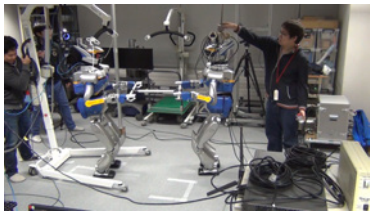
(f) t=225 [s]



(g) t=225.5 [s]



(h) t=226 [s]



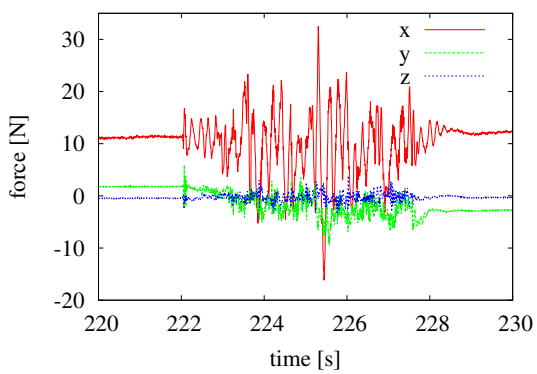
(i) t=226.5 [s]

(j) t=227 [s]

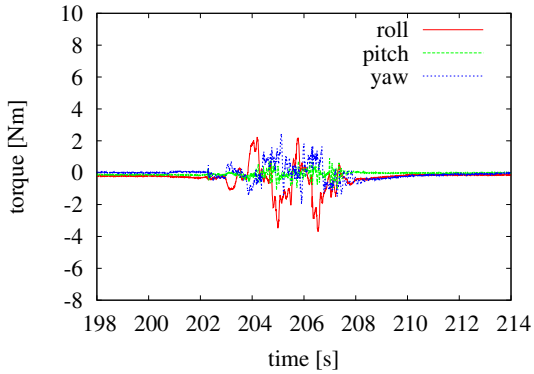
(k) t=227.5 [s]

(l) t=228 [s]

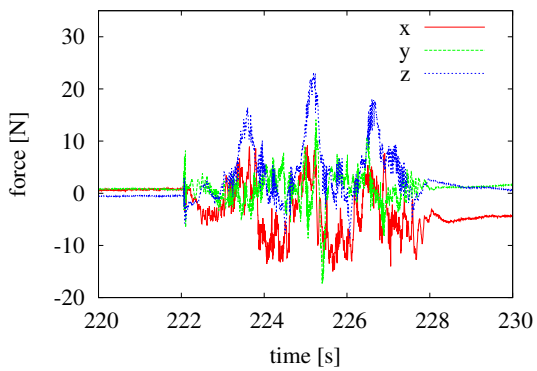
Figure 5.28: Sequential photo graphs of experiment (x + yaw direction move-met).



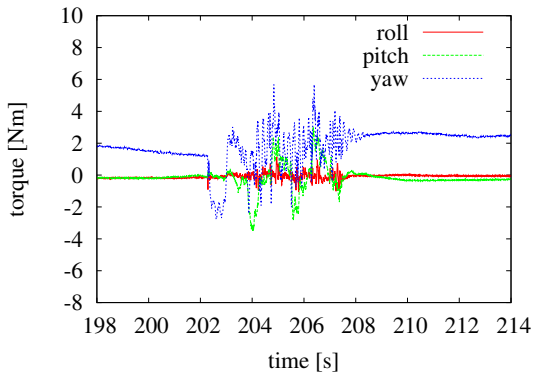
(a) Force of ${}^{a}\boldsymbol{f}_{r1}$.



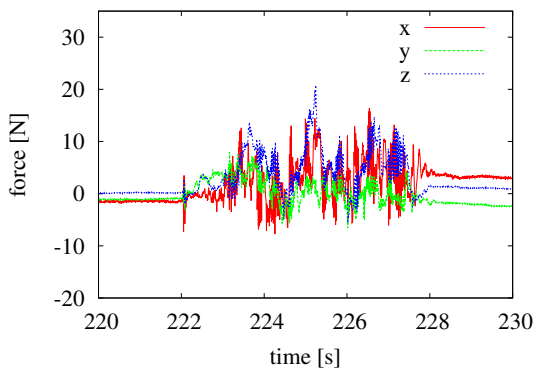
(b) Moment of ${}^{a}\boldsymbol{f}_{r1}$.



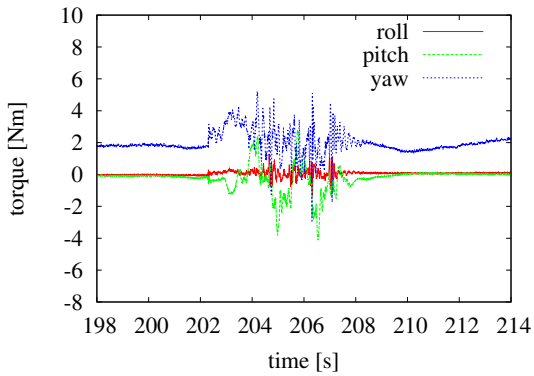
(a) Force of ${}^{a}\boldsymbol{f}_{r2}$.



(b) Moment of ${}^{a}\boldsymbol{f}_{r2}$.



(a) Force of ${}^{a}\boldsymbol{f}_{r3}$.



(b) Moment of ${}^{a}\boldsymbol{f}_{r3}$.

Figure 5.29: The history of internal forces(t= 220 [s] to 230 [s]).

Chapter 6. Conclusion

6.1 Conclusion of this thesis

In the thesis, a framework of cooperative object transportation by multiple humanoid robots is proposed. The two cooperation types, leader-follow type and symmetry type, are validated with humanoid robots, respectively. Unlike wheeled robot, the support polygon of a walking humanoid is narrow. Therefore, the balance of one walking humanoid robot may be affected by external force easily. The external force of one humanoid robot, i. e. the internal force between humanoid robots should be eased in some way. In leader-follower type, a impedance based control is used, and in symmetry type, a hybrid position/force control is proposed. This two type is compared to each other in this thesis. The briefings of each chapter are as follow.

chapter 1 Introduction

The weakness of joint actuator output is a bottleneck so that a robot can not perform as powerful as human being. Cooperative working may be a effective way to exploit the working capacity of a robot. The target of this research is to develop a cooperative object transportation framework for humanoid robots in order to move heavy object.

chapter 2 System

Dynamic simulator "OpenHRP" is used and humanoid robot "HRP-2" for validating our method. The controller of humanoid robots are made in "OpenRTM" format. In this format, user is able to operate the robots both in simulation world and in real world with the same program. The joint of HRP-2 is control in torque-control mode. Each joint torque is calculated with a simple PD control.

chapter 3 Online Walking Generator

The preview control theory is used as walking pattern generator. This method can calculate a reference center of mass (CoM) trajectory that meets the reference zero moment point (ZMP) trajectory. After the foot placement is determined, the reference ZMP is computed by fifth-interpolating. Then by applying the preview control theory, the reference CoM trajectory is computed. The DOF of two legs are 12 and the constrain condition is also in 12 dimension, and the reference joint angles of two legs are calculated numerically. The walking motion is generated with online walking command.

chapter 4 Leader-follower Type Cooperation Control

Leader-follower type cooperation control with two humanoid robots is validated. The foot placement of leader-robot is given by operator online. And impedance control is applied to two robots in order to ease internal force between leader and follower robot. Robot moves their hand along the direction of external force estimated form the force sensor mounted at its wrist. Then the foot placement of the follower robot is determined by current hand position. Leader-follower type control is validated in x direction only.

chapter 5 Symmetry Type Cooperation Control

The external and internal force among arbitrary robots are defined, respectively. Since the internal force can not be determined uniquely, a method to determine the internal forces which will become controlled variables is proposed. A hybrid position/force control law is derived, and is able to be applied to arbitrary humanoid robots. Also, a walking pattern generator (the same as used in leader-follower type) is integrated into the control law. The proposed method is validated with four HRP-2 robot in simulator. Our proposed symmetry type control is validated both in steady state (without walking) and transportation motion (with walking), respectively. In steady state, the reference object position is set as initial position, and a reference internal forces are given online. The robots can track the reference internal forces while keeping in initial position. And in transportation motion experiment, the humanoid robots are able to generate its whole body motion to transport an object autonomously, and easing the internal force in the same time.

6.2 Future works

There are some tasks are not clear yet. Future works includes:

Improve the moving direction prediction of follower robot

In leader-follower type control, moving direction of a follower robot is based on the force sensor information. In this thesis, only x direction is validated. However, as the DOF of moving direction increases, the complexity also increases. Besides, this thesis only validate leader-follower type with 2 humanoid robots. Complexity also increases if there are more humanoid robots. Internal forces are not from leader robot only, but also form other follower robots. The moving direction prediction of follower robot should be improved in order to follower the leader robot properly.

A smart way to adjust impedance gain of leader-follower type

Currently, the impedance gain is adjusted by trail and error. The impedance gain is very critical in leader-follower type, and currently there is not a smart way to determine it. An auto adjust method is expected.

Feedback the information of stabilizer

In this thesis, the correction information of stabilizer is not feedback into trajectory planning part. Since the stabilizer may adjust the hand position, the hybrid position/force control may be nonfunctional if the amount of correction is neglected. The out put of the stabilizer is fluctuating, and if the information of stabilizer is feed backed with out any prepossessing, the output of trajectory planning part also becomes fluctuating. This problem may be improved by a proper filter

Estimate the relative position of humanoid robots without environment camera

In symmetry type control, environment camera is used to capture the initial relative posit on of the robots. However, environment camera may not always exist. It is ideal if the humanoid robots can estimate their relative position by their camera mounted in their heads.

Take weight of holding object into consider

In this thesis, a method to ease the internal forces between humanoid robots is mainly developed. The weight of a holding object is neglected. However, robots may falling down if the holding object is heavy. The weight of a holding object should take into consider when generating the CoM trajectory.

Acknowledgements

First of all, I would like to thank my adviser, Professor Konno for his continued support. He taught me many things not only in my research but also about how engineering and a researcher should be. Acknowledgment are also given to the members of my advisory committee, Professors Ogasawara and Yamashita for their useful advice, recommendations and patient reading.

I and greatly thankful to all the members of Uchiyama and Konno Lab. especially to Professor Uchiyama, Dr. Koyu Abe, Dr. Xin Jiang, Dr. Satoko Abiko, Dr. Teppei Tsujita and the previous and current members in the humanoid group. Professor Uchiyama masaru receive me to the Lab. , and taught me what research is. Dr. Koyu Abe was always ready to give me help both in research and my life in Sendai. The discussions with the members of humanoid group provided me with insightful ideas. Mr. Shuhei Ogawa helped me to develop the software for controlling robots. Mr. Kazuya Sase, Mr. Yoshihito Onuki, Mr. Kouki Kasai, Mr. Taisho Sugimoto, Mr. Nobuhiro Nagayasu, Mr. Keisuke Nishizaki, Mr. Sota Hayakawa helped me to proceed my experiment. Dr. Shunsuke Komizunai helped my instrument purchasing.

The Lab. members always supported me with readiness and made my student life fun.

Finally, I sincerely thank my family for their support during the doctoral program. Also, my friends in Taiwan. Without their encouragement and support, this work would have been impossible.

References

[1] I. Kato, Develop of biped robot WABOT–1 (in Japanese), bio mechanism2, pp. 173–214, 1973.

[2] J. Y amaguchi, S. Inoue, D. Nishino and A. Takanishi, Development of a bipedal humanoid robot having antagonistic driven joints and three DOF trunk, in *Proc. of Int. Conf. on Intelligent Robots and Systems*, pp. 96-—101, 1998.

[3] K. Hirai, Current and future perspective of Honda humanoid robot, in *Proc. of IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, pp. 500–508, 1997.

[4] M. Hirose, Y. Haikawa, T. Takenaka and K. Hirai: Development of Humanoid Robot ASIMO, in *Proc. of IEEE/RSJ Int. Workshop on Intelligent Robots and Systems*, pp. 3–10, 2001.

[5] http://autoc-one.jp/news/893190/photo/0007.html

[6] H. Inoue, H. Hirukawa, HRP: Humanoid Robotics Project of MITI, in *Journal of the Robotics Society of Japan*, vol. 19, no. 1, pp. 2–7, 2001.

[7] K. Kaneko, F. Kanehiro, S. Kajita, H. Hirukawa, T. Kawasaki, M. Hirata, K. Akachi and T. Isozumi, Humanoid robot HRP-2, in *IEEE Int. Conf. on Robotics and Automation*, pp. 1083–1090, 2004.

[8] http://global.kawada.jp/mechatronics/hrp2.html

[9] http://www.kawada.co.jp/ams/hrp-3p/index.html

[10] K. Kaneko, F. Kanehiro, M. Morisawa, K. Miura, S. Nakaoka, S. Kajita, Cybernetic human HRP-4C, *Proceedings of the 20 IEEE-RAS International Conference on Humanoid Robots (Humanoids 2009)*, pp.7–14.

[11] S. Kagami, K. Nishiwaki, T. Sugihara, J. J. Kuffner Jr., M. Inaba, H. Inoue: Design and Implementation of Software Research Platform for Humanoid Robotics : H6, in *Proc. of IEEE Int. Conf. on Robotics and Automation*, pp. 2431–2436, 2001.

[12] Y. Nakanishi, Y. Asano, T. Kozuki, H. Mizoguchi, Y. Motegi, M. Osada, T. Shirai, J. Urata, K. Okada, M. Inaba, Design Concept of Detail Musculoskeletal Humanoid "Kenshiro" -Toward a real human body musculoskeletal simulator, *Proceedings of the 2012 IEEE-RAS International Conference on Humanoid Robots (Humanoids 2012)*, pp.1–6, 2012.

[13] A. Konno, N. Kato, S. Shirata, T. Furuta, and M. Uchiyama: "Development of a Light-Weight Biped Humanoid Robot, " In *Proc. of IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, pp. 1565–1570, 2000.

[14] S. Shirata, A. Konno, and M. Uchiyama: "Design and Development of a Light-Weight Biped Humanoid Robot Saika-4," In *Proc. of IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, pp. 148–153, 2004.

[15] Y. Kuroki, M. Fujita, T. Ishida, K. Nagasaka, J. Yamaguchi, A small biped entertainment robot exploring attractive applications, in *IEEE Int. Conf. on Robotics and Automation*, pp. 471–476, 2003.

[16] E. Nakano, S. Ozaki, T. Ishida, I. Kato, Cooperational Control of the Anthropomorphous Manipulator "MELARM", Proc. 4th International Symposium on Industrial Robots, pp.251–260, 1974.

[17] K. Munawar, M. Uchiyama, Experimental verification of distributed event-based control of multiple unifunctional manipulators, in *IEEE Int. Conf. on Robotics and Automation*, pp. 1213–1218, 1999.

[18] A. Babazadeh, N. Sadati, Optimal control of multiple-arm robotic systems using gradient method, *Robotics, Automation and Mechatronics, 2004 IEEE Conference on*, vol.1 Volume: 1, pp. 312 –317, 2004.

[19] D. Williams, Oussama Khatib, The virtual linkage: a model for internal forces in multi-grasp manipulation, in *IEEE Int. Conf. on Robotics and Automation*, pp. 1025–1030, 1993.

[20] S. Hayati, Hybrid position/force control of multi-arm cooperating robotics, in *IEEE Int. Conf. on Robotics and Automation*, pp. 82–89, 1986.

[21] M. Uchiyama, and P. Dauchez, Symmetric kinematic formulation and non-master/slave coordinated control of two-arm robots, in *Advanced Robotics*, vol .7, No. 4, pp. 361–383, 1993.

[22] P. Pagilla, M. Tomizuka, Hybrid force/motion control of two arms carrying an object, *American Control Conference*, 1994, pp. 195–199,1994

[23] T. Yoshikawa, X.Z Zheng, Coordinated dynamic control for multiple robot manipulators handling an object, *Advanced Robotics, 1991. 'Robots in Unstructured Environments', 91 ICAR., Fifth International Conference on*, vol.1, pp. 579–584, 1991.

[24] S. Schneider and R. Cannon, Object Impedance Control for Cooperative Manipulation: Theory and Experimental Results, *IEEE Trans. Robot. Automat.*, vol. 8, no. 3, pp. 383–394, 1992.

[25] J. Szewczyk, F. Plumet, and P. Bidaud, Planning and controlling cooperating robots through distributed impedance, *Journal of Robotic Systems*, vol. 19, no. 6, pp. 283–297, 2002.

[26] R. Rastegari, S.A.A. Moosavian, Multiple Impedance control of cooperative manipulators using virtual object grasp, Computer Aided Control System Design, *IEEE International Conference on Control Applications, 2006 IEEE International Symposium on Intelligent Control*, pp. 2872–2877, 2006.

[27] Y. Kume, Y. Hirata, Z.D. Wang, K. Kosuge, Decentralized Control of Multiple Mobile Manipulators Handling a Single Object in Coordination, in *Int. Conf. on Intelligent Robots and Systems*, pp. 2758–2763, 2002.

[28] Y. Hirata, T. Sawada, Z. Wang and K. Kosuge, Leader-Follower type Motion Control Algorithm of Multiple Mobile Robots with Dual Manipulators for Handling a Single Object in Coordination, in *Int. Conf. on Intelligent Robots and Systems*, pp. 362–367, 2004.

[29] S. Erhart, D. Sieber, and S. Hirche, An impedance-based control architecture for multi-robot dual-arm mobile manipulation,in *Int. Conf. on Intelligent Robots and Systems*, pp. 315–322, 2013.

[30] O. Khatib, K. Yokoi, K. Chang, D. Ruspini, R. Holmberg, A. Casal, A. Baader, Force strategies for cooperative tasks in multiple mobile manipulation systems, in *International Symposium on Robotics Research*, pp. 333–342, 1996.

[31] K. Yokoyama, H. Handa, T. Isozumi, Y. Fukase, K. Kaneko, F. Kanehiro, Y. Kawai, F. Tomita, and H. Hirukawa, Cooperative works by a human and a humanoid Robot, in *IEEE Int. Conf. on Robotics and Automation*, pp. 2985–2991, 2003.

[32] D. J. Agravante, A. Cherubini, A. Bussy, P. Gergondet, A. Kheddar, Collaborative human-humanoid carrying using vision and haptic sensing, in *IEEE Int. Conf. on Robotics and Automation*, pp. 607–612, 2014.

[33] G. Pratt and J. Manzo, The DARPA Robotics Challenge, *IEEE Robotics and Automation Magazine*, Vol. 20, No. 2, pp. 10–12, 2013.

[34] F. Kanehiro, H.Hirukawa, S.Kajita, OpenHRP: Open Architecture Humanoid Robotics Platform, *International Journal of Robotics Research*, vol. 23, no. 2, pp. 155–165, 2004.

[35] National Institute of Advanced Industrial Science and Technology (AIST), "RT-Middleware: OpenRTM-aist," 2009, http://www.openrtm.org/.

[36] OMG's Corba Official website： http://www.corba.org/

[37] The Jython Project website： http://www.jython.org/

[38] K. Kaneko F. Kanehiro, S. Kajita, K. Yokoi, K. Fujiwaera, K. Harada, H. Hirukawa, Stabilizer for biped walking (in Japanese), in *Proceedings of the 21st Annual Conference of The Robotics Society of Japan*, Tokyo, Japan, September 2003.

[39] M. Vukobratovic, J. Stepanenko, On the Stability of Anthropomorphic Systems, Mathematical Biosciences, vol. 15, pp. 1-37, 1972.

[40] S. Kajita, F. Kanehiro, K. Kaneko, K. Fujiwara, K. Harada, K. Yokoi and H. Hirukawa, Biped Walking pattern Generation by Using Preview Control of Zero-moment Point, in *IEEE Int. Conf. on Robotics and Automation*, pp. 1620–1626, 2003.

[41] S. Kajita, H. Hirukawa, K. Harada, K. Yokoi, Introduction to Humanoid Robotics, Springer, 2014.

[42] K. Nishiwaki, S. Kagami, Online Walking Control System for Humanoids with Short Cycle Pattern Generation, *The International Journal of Robotics Research*, vol. 28, no. 6, 729–742.

[43] M. Vukobratović, B. Borovac, and D. Ŝurdilović, Zero-moment point–proper interpretation and new applications, in *Int. Conf. on Humanoid Robots*, pages 237–244, Tokyo, Japan, 2001.

[44] http://www.optitrack.com/

[45] M. Wu, A.Konno, S. Ogawa, S. Komizunai, Symmetry Cooperative Object Transportation by Multiple Humanoid Robots, in *IEEE Int. Conf. on Robotics and Automation*, pp. 3446–3451, 2014.

[46] M. Wu, A. Konno, M. Uchiyama, Cooperative Object Transportation by Multiple Humanoid Robots,in *Proceedings of the 2011 IEEE/SICE Int. Symposium on System Integration*, pp. 779–pp. 784 ,2011.

Appendix A Calculation of ${}^{\mathrm{o}}\boldsymbol{w}_{\mathrm{b}}$ from ${}^{\mathrm{o}}\boldsymbol{w}_{\mathrm{h}}$

Let ${}^{\mathrm{o}}\boldsymbol{F}_{\mathrm{h}i}$ and ${}^{\mathrm{o}}\boldsymbol{N}_{\mathrm{h}i}$ be the force and moment vectors applied to the object by hand i as illustrated in Figure A.1. The relationship between $\begin{bmatrix} {}^{\mathrm{o}}\boldsymbol{F}_{\mathrm{b}i}^{\mathrm{T}} & {}^{\mathrm{o}}\boldsymbol{N}_{\mathrm{b}i}^{\mathrm{T}} \end{bmatrix}^{\mathrm{T}}$ and $\begin{bmatrix} {}^{\mathrm{o}}\boldsymbol{F}_{\mathrm{h}i}^{\mathrm{T}} & {}^{\mathrm{o}}\boldsymbol{N}_{\mathrm{h}i}^{\mathrm{T}} \end{bmatrix}^{\mathrm{T}}$ is given as

$$\begin{bmatrix} {}^{\mathrm{o}}\boldsymbol{F}_{\mathrm{b}i}^{\mathrm{T}} \\ {}^{\mathrm{o}}\boldsymbol{N}_{\mathrm{b}i}^{\mathrm{T}} \end{bmatrix} = \begin{bmatrix} \mathbf{I}_3 & \mathbf{0}_3 \\ -{}^{\mathrm{o}}\hat{\mathbf{l}}_{\mathrm{h}i} & \mathbf{I}_3 \end{bmatrix} \begin{bmatrix} {}^{\mathrm{o}}\boldsymbol{F}_{\mathrm{h}i}^{\mathrm{T}} \\ {}^{\mathrm{o}}\boldsymbol{N}_{\mathrm{h}i}^{\mathrm{T}} \end{bmatrix} \equiv {}^{\mathrm{o}}\mathbf{D}_{\mathrm{h}i}\,{}^{\mathrm{o}}\boldsymbol{f}_{\mathrm{h}i} \ , \tag{A.1}$$

where ${}^{\mathrm{o}}\hat{\mathbf{l}}_{\mathrm{h}i}$ is a skew symmetric 3×3 matrix that satisfies

$${}^{\mathrm{o}}\hat{\mathbf{l}}_{\mathrm{h}i}\,{}^{\mathrm{o}}\boldsymbol{F}_{\mathrm{h}i} = {}^{\mathrm{o}}\mathbf{l}_{\mathrm{h}i} \times {}^{\mathrm{o}}\boldsymbol{F}_{\mathrm{h}i} \ . \tag{A.2}$$

In a n robotic arms system, we obtain

$${}^{\mathrm{o}}\boldsymbol{q}_{\mathrm{b}} = \mathrm{diag}[{}^{\mathrm{o}}\mathbf{D}_{\mathrm{h}1},\ {}^{\mathrm{o}}\mathbf{D}_{\mathrm{h}2},\ \ldots\ {}^{\mathrm{o}}\mathbf{D}_{\mathrm{h}n}]\,{}^{\mathrm{o}}\boldsymbol{q}_{\mathrm{h}} \equiv {}^{\mathrm{o}}\mathbf{D}_{\mathrm{h}}\,{}^{\mathrm{o}}\boldsymbol{q}_{\mathrm{h}} \ . \tag{A.3}$$

Applying the principle of virtual work to (A.3), we obtain

$${}^{\mathrm{o}}\boldsymbol{q}_{\mathrm{h}}^{\mathrm{T}}\ {}^{\mathrm{o}}\boldsymbol{w}_{\mathrm{h}} = ({}^{\mathrm{o}}\mathbf{D}_{\mathrm{h}}\,{}^{\mathrm{o}}\boldsymbol{q}_{\mathrm{h}})^{\mathrm{T}}\ {}^{\mathrm{o}}\boldsymbol{w}_{\mathrm{b}}, \tag{A.4}$$

$${}^{\mathrm{o}}\boldsymbol{w}_{\mathrm{h}} = {}^{\mathrm{o}}\mathbf{D}_{\mathrm{h}}^{\mathrm{T}}\ {}^{\mathrm{o}}\boldsymbol{w}_{\mathrm{b}} \ , \tag{A.5}$$

Where ${}^{\mathrm{o}}\boldsymbol{w}_{\mathrm{h}}$ is velocity vector of $\Sigma_{\mathrm{b}i}$ which is defined as

$${}^{\mathrm{o}}\boldsymbol{w}_{\mathrm{h}} \equiv \begin{bmatrix} {}^{\mathrm{o}}\boldsymbol{s}_{\mathrm{h}1} & {}^{\mathrm{o}}\boldsymbol{s}_{\mathrm{h}2} & \ldots & {}^{\mathrm{o}}\boldsymbol{s}_{\mathrm{h}i} \end{bmatrix} \ , \tag{A.6}$$

$${}^{\mathrm{o}}\boldsymbol{s}_{\mathrm{h}i} \equiv \begin{bmatrix} {}^{\mathrm{o}}\boldsymbol{v}_{\mathrm{h}i}^{\mathrm{T}} & {}^{\mathrm{o}}\boldsymbol{\omega}_{\mathrm{h}i}^{\mathrm{T}} \end{bmatrix} \ . \tag{A.7}$$

${}^{\mathrm{o}}\boldsymbol{v}_{\mathrm{h}i}$ and ${}^{\mathrm{o}}\boldsymbol{\omega}_{\mathrm{h}i}$ are translational and rotational velocities of $\Sigma_{\mathrm{h}i}$, respectively.

Finally, ${}^{\mathrm{o}}\boldsymbol{w}_{\mathrm{b}}$ is calculated from ${}^{\mathrm{o}}\boldsymbol{w}_{\mathrm{h}}$ by

$${}^{\mathrm{o}}\boldsymbol{w}_{\mathrm{b}} = ({}^{\mathrm{o}}\mathbf{D}_{\mathrm{h}}^{\mathrm{T}})^{-1}\ {}^{\mathrm{o}}\mathbf{w}_{\mathrm{h}} \ . \tag{A.8}$$

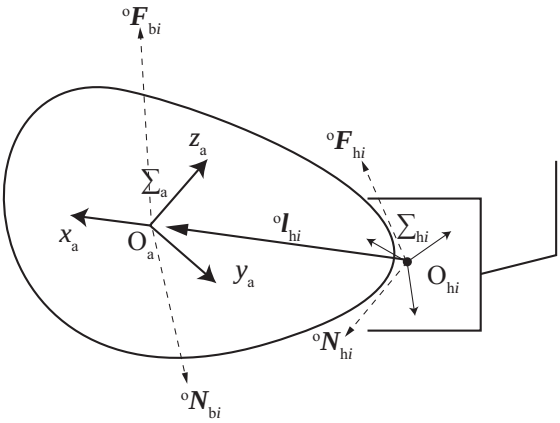


Figure A.1: Relationship between .

Appendix B Translation between Derivative of Rotation Angles and Angular Velocity

When an attitude is represented in Z-Y-X Euler angle, the derivative of rotation angles (α, β, γ) is translated into angular velocity (w_x, w_y, w_z) as follows

$$\begin{aligned}\begin{bmatrix} w_x \\ w_y \\ w_z \end{bmatrix} &= \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \dot{\alpha} + \begin{bmatrix} C_\alpha & -S_\alpha & 0 \\ S_\alpha & C_\alpha & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix} \dot{\beta} + \begin{bmatrix} C_\alpha & -S_\alpha & 0 \\ S_\alpha & C_\alpha & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} C_\beta & 0 & S_\beta \\ 0 & 1 & 0 \\ -S_\beta & 0 & C_\beta \end{bmatrix} \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix} \dot{\gamma} \\ &= \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \dot{\alpha} + \begin{bmatrix} -S_\alpha \\ C_\alpha \\ 0 \end{bmatrix} \dot{\beta} + \begin{bmatrix} C_\alpha C_\beta \\ S_\alpha C_\beta \\ -S_\beta \end{bmatrix} \dot{\gamma} = \begin{bmatrix} 0 & -S_\alpha & C_\alpha C_\beta \\ 0 & C_\alpha & S_\alpha C_\beta \\ 1 & 0 & -S_\beta \end{bmatrix} \begin{bmatrix} \dot{\alpha} \\ \dot{\beta} \\ \dot{\gamma} \end{bmatrix} = \mathbf{B}_0 \begin{bmatrix} \dot{\alpha} \\ \dot{\beta} \\ \dot{\gamma} \end{bmatrix} , \end{aligned} \tag{B.1}$$

where S and C are *sin* and *cos* calculation, respectively.

Assuming that ${}^{\mathrm{o}}\Delta \boldsymbol{p}_{\mathrm{r},n}$ is very small, ${}^{\mathrm{o}}\Delta \boldsymbol{s}_{\mathrm{r},n}$ is approximated as ${}^{\mathrm{o}}\Delta \dot{\boldsymbol{p}}_{\mathrm{r},n}$. The relationship between $\boldsymbol{u}$ and $\dot{\boldsymbol{z}}$ is given by

$$\boldsymbol{u} = \mathbf{B}_{\mathrm{a}} \dot{\boldsymbol{z}} , \tag{B.2}$$

where $\mathbf{B}_{\mathrm{a}}$ is given by

$$\mathbf{B}_{\mathrm{a}} \equiv \begin{array}{c} \\ 6n-6 \end{array} \overbrace{\left\{ \begin{bmatrix} \mathbf{I}_3 & & & & & & 0 \\ & \mathbf{B}_0 & & & & & \\ & & \mathbf{I}_3 & & & & \\ & & & {}^{\mathrm{a}}\mathbf{B}_0 & & & \\ & & & & \mathbf{I}_3 & & \\ 0 & & & & & {}^{\mathrm{a}}\mathbf{B}_0 & \\ & & & & & & \ddots \end{bmatrix} \right.}^{6n} , \tag{B.3}$$

$${}^{\mathrm{a}}\mathbf{B}_0 \equiv {}^{\mathrm{a}}\mathbf{R}_{\mathrm{o}} \, \mathbf{B}_0 \, {}^{\mathrm{a}}\mathbf{R}_{\mathrm{o}}^{\mathrm{T}} . \tag{B.4}$$

${}^{\mathrm{a}}\mathbf{R}_{\mathrm{o}}$ is a rotational transformation matrix from Σ_{o} to Σ_{a}.