Pushing Motion of Humanoid Robot on Dynamic Locomotion

Motomi Ikebe, Toshiaki Tsuji, Yoshiharu Sato, Koichiro Tsuji and Kouhei Ohnishi
Keio University
Department of System Design Engineering
3-14-1, Hiyoshi, Kohoku-ku, Yokohama, Japan
(ikebe,tsuji,yoshiharu,koichiro,ohnishi)@sum.sd.keio.ac.jp

Abstract— **This paper describes the pushing motion of a humanoid robot. It is difficult to achieve a pushing motion at constant velocity of an object since the body velocity fluctuates during the dynamic walking. Furthermore, the analysis of walking stability becomes complicated with the pushing force. We constructed a humanoid control system that is separated to an upper body controller and a lower body controller. The upper body concentrates on the force control with a constant command. The lower body deals with the biped locomotion and is controlled so that the object is within range of the arm movement. This combination of two separated controllers solves the in the pushing motion of humanoid robots. Simulation results show that the pushing motion with constant velocity of the object was achieved.**

Key Words: **Pushing motion, humanoid robot, dynamic stability, walking pattern generator**

1 Introduction

1.1 Background

The labor force is replaced with industrial robots as the results of previous researches on the robot control area. The main stream of the robot control research has shifted to the study to execute more complicated tasks in wider situations including human environments. Biped robot has a mechanical advantage that it can decide the discrete contact areas on the ground and it can accommodate to an unleveled ground. Hence, it has adaptability to complex places such as human environments. Furthermore, a humanoid robot, which is often biped, is similar to human with respect to its structure. This means that a humanoid robot can utilize many kinds of equipments in human environments. It also means that a humanoid robot is well accepted to the human society. In this paper, we focus on the pushing motion of a humanoid robot as a typical example of the tasks in human environments.

Compared to other mobile robots, humanoid robots have the problem that pushing force strongly interferes their walking stability. Harada *et al.* proposed two kinds of ZMPs in order to achieve the pushing motion[1]. The control method for the task which requires large force on hands was developed[2] by Yoshida *et al.* from the view of optimal posture, joint torques and footholds. The whole body cooperation of the humanoid robot to output large force on hands[3] was evaluated by Hwang *et al.*. These methods discussed the stability of the pushing motion when it is just standing. However, it is very important for a humanoid robot to achieve pushing motion during walking motion as well in order to advance the mobility of the humanoid robot.

Considering the walking stability, there is another problem on pushing motion that the object velocity may fluctuate because the body link velocity of the biped robot widely varies during the walking. It is desirable for some kinds of objects to be conveyed with the least acceleration. Variance of the body link velocity also brings a problem that it makes difficult to keep the stable contact to the object.

The goal of this research is to solve these problems and achieve a stable pushing motion of a biped humanoid robot on dynamic locomotion.

1.2 Overview of Pushing Motion

To make the system simple, we make an assumption that the parameters of the pushing object are known. We introduce a method to divide the humanoid robot control system into the upper body controller and the lower body controller. The upper body controller tends only to the constant force control for the pushing motion. The humanoid robot attains constant pushing force while the object is in the range of arm movement. The lower body should be controlled to follow the object with some distance so that the object comes in the range of arm movement. Walking pattern generator[6] is applied for the trajectory planning of the lower body to follow the object.

This control system is easy in the sense that the entire control system is divided to two subsystems and both subsystems are constructed by existing methods. However, this combination of two subsystems solves the problems on the pushing motion of a humanoid robot.

This paper is organized as follows. In section 2,

0-7803-8662-0/04/$20.00 ©2004 IEEE

Table 1: Parameters of the robot

		Size[mm]	Mass[kg]
Upper Body	Body	$173 \times 200 \times 560$	6.0
	Upper Arm	200	0.7×2
	Lower Arm	230	0.3×2
Lower Body	Body	$519 \times 472 \times 132$	19.4
	Thigh	300	1.5×2
	Shin	300	0.5×2
	Foot	$142 \times 270 \times 55$	2.1×2

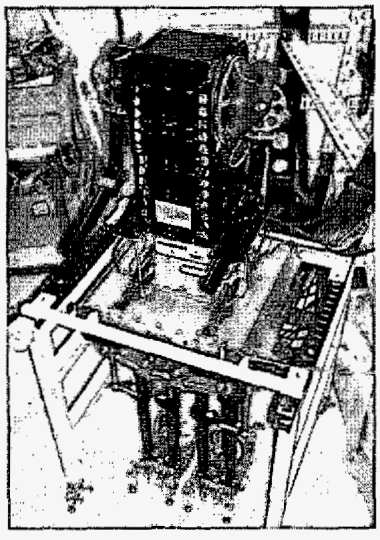

Fig. 1: Overview of the humanoid robot.

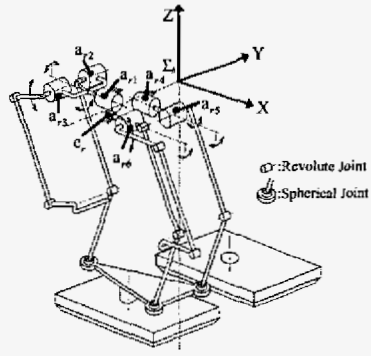

Fig. 2: Structure of the leg.

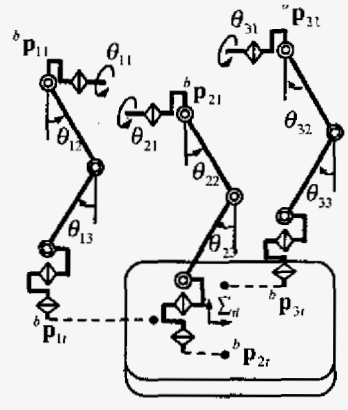

Fig. 3: Left leg model.

we describe the model of the humanoid robot. Linear inverted pendulum mode is introduced for trajectory planning of a humanoid robot in section 3. In section 4, we show how to select the landing point so that the walking velocity corresponds to the object velocity. Whole control system is shown in section 5. Simulation result is shown in section 6. In section 7 we conclude the paper and states our future works.

2 Modeling

Whole body of the humanoid robot is shown in Fig. 1. The upper body of the humanoid robot has 4 DOF (degree of freedom) on each arm. As we fix the robot movement in sagittal plane, two joints are fixed. Accordingly, each arm has 2 DOF in sagittal plane. Each robot's leg with the parallel mechanism has 6 DOF.

A coordinate system of the base link is defined as shown in Fig. 2. The origin of base coordinate system Σ_b is defined at the center of the plane where actuators are located like Fig. 2. The parameters of the robot are represented in Table 1. Hereinafter, we will represent only the kinematics of the left leg and the left arm. These of the right can be also expressed in the same way.

The kinematic relationship from the base to the foot tip is denoted as follows:

$$ {}^b\boldsymbol{x}_f = \begin{bmatrix} {}^b\boldsymbol{p}_f \\ {}^b\boldsymbol{a}_f \end{bmatrix}. \tag{1} $$

where ${}^b\boldsymbol{p}_f$ is a position vector of each foot in Σ_b, and ${}^b\boldsymbol{a}_f$ is a direction vector which expresses the posture of the foot in Σ_b. Inverse kinematics of the legs can be obtained as follows:

$$ \boldsymbol{\theta}_l = \boldsymbol{H}^f({}^b\boldsymbol{x}_f). \tag{2} $$

where $\boldsymbol{\theta}_l = [\theta_{11}, \theta_{12}, \theta_{13}, \theta_{22}, \theta_{23}, \theta_{33}]^T$ is an active joint angle vector as shown in Fig. 3. $\boldsymbol{H}^{\boldsymbol{foot}}(\cdot)$ is the function of kinematics from the foot position to the joint angles. A Jacobian matrix of the parallel mechanism, $\boldsymbol{J}_l$, is defined as follows unlike a serial mechanism:

$$ \dot{\boldsymbol{\theta}}_l = \frac{\partial \boldsymbol{H}^f({}^b\boldsymbol{x}_f)}{\partial\, {}^b\boldsymbol{x}_f}\, {}^b\dot{\boldsymbol{x}}_f \tag{3} $$

$$ = \boldsymbol{J}_l({}^b\boldsymbol{x}_f)\, {}^b\dot{\boldsymbol{x}}_f. \tag{4} $$

Details of kinematics and dynamics of the leg are explained in Ref. [4]. The model of the upper body in sagittal plane is shown in Fig. 4. The kinematic relationship from the shoulder joint to the tip of the arm

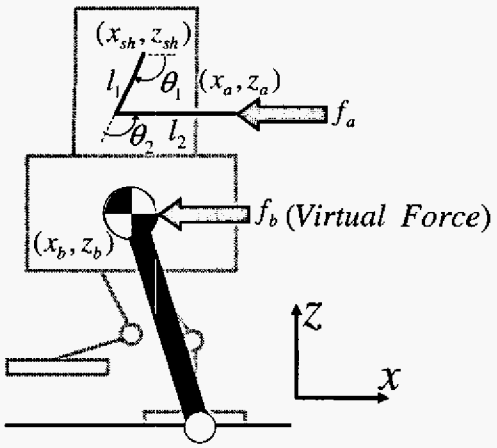

Fig. 4: The model of an inverted pendulum.

is represented as follows:

$$ {}^b\boldsymbol{x}_a = \begin{bmatrix} l_1\cos\theta_1 + l_2\cos(\theta_1+\theta_2) + {}^bx_{sh} \\ -l_1\sin\theta_1 - l_2\sin(\theta_1+\theta_2) + {}^bz_{sh} \end{bmatrix}. \quad (5) $$

$$ \begin{array}{ll} {}^b\boldsymbol{x}_a = [x_a, z_a]^T & : \text{tip position of the arm} \\ {}^b\boldsymbol{x}_{sh} = [x_{sh}, z_{sh}]^T & : \text{shoulder joint position} \\ \boldsymbol{\theta}_a = [\theta_1, \theta_2]^T & : \text{arm joint angle} \\ l_i & : \text{length of arm link } i \end{array} $$

The Jacobian matrix of the arm $\boldsymbol{J}_a$ is defined as follows:

$$ {}^b\dot{\boldsymbol{x}}_a = \boldsymbol{J}_a\dot{\boldsymbol{\theta}}_a. \quad (6) $$

$$ \boldsymbol{J}_a = \begin{bmatrix} -l_1\sin\theta_1 - l_2\sin(\theta_1+\theta_2) & -l_2\sin(\theta_1+\theta_2) \\ -l_1\cos\theta_1 + l_2\cos(\theta_1+\theta_2) & -l_2\cos(\theta_1+\theta_2) \end{bmatrix} $$

Pushing force $\boldsymbol{f}_a$ is represented as follows:

$$ \boldsymbol{\tau} = \boldsymbol{J}_a^T\boldsymbol{f}_a. \quad (7) $$

where $\boldsymbol{f}_a = [f_a, 0]^T$ are the horizontal arm force exerted at the tip of the arm to the obstacle. We assume that the arm force has only the horizontal component.

Since the robot has small mass of the leg and the arm, the COG (center of gravity) position of the robot scarcely varies. This robot is better suited to the model of an inverted pendulum.

3 Linear Inverted Pendulum Mode

There is a need to consider whether a humanoid robot can sustain the locomotion without falling since it is not fixed to the ground. This property to sustain the locomotion is called walking stability. The biped robot should plan its motion so that the stability of the biped locomotion is assured. The robot motion should be calculated on real time in order to adapt the unknown environment. Thus, we apply linear inverted pendulum mode(LIPM)[7] for the trajectory planning of the body since it is able to plan a stable trajectory with a small calculation amount.

However, reaction force of the pushing motion affects the walking stability. A stability analysis of biped locomotion becomes complex in general if we consider the force acting on the upper body. The pushing force is controlled to be constant so as to make the stability analysis simple. The trajectory planning of LIPM with constant pushing force is described briefly in this section. Fig.4 shows a model of an inverted pendulum. (x_b, z_b) is the position of the mass point. The humanoid robot is modeled as an inverted pendulum with the mass point on the base link of the entire robot. The length of the pendulum is variable. Its supporting point is at the center of the support foot surface. The trajectory of the mass point, the trajectory of the base link in fact, is planned so that its height is constant and no torque occurs on the supporting point. The motion of an inverted pendulum with these conditions is figured out from (8).

$$ \ddot{x}_b = \frac{g}{z_b}x_b \quad (8) $$

where z_b denotes the height of the base link and g is the gravity acceleration.

The reaction force acting on the arms will interfere with the motion of an inverted pendulum when humanoid robot is executing some operations with its arms. The effect may become large especially during a pushing motion. f_b, the equivalent force on an inverted pendulum caused by pushing force f_a, is figured out from (9) in order to take account of the pushing force in the dynamics of an inverted pendulum.

$$ \begin{aligned} f_b z_b &= f_a z_a \\ f_b &= \frac{z_a}{z_b}f_a \end{aligned} \quad (9) $$

where z_a represents the height of the tip of arms.

Motion of the inverted pendulum considering the pushing force is given by

$$ \ddot{x}_b(t) = \frac{g}{z_b}x_b(t) - \frac{f_b}{m}. \quad (10) $$

where m is the point mass of an inverted pendulum. With the assumption of the constant pushing force, it becomes able to consider the influence of the pushing force in LIPM. The motion of the inverted pendulum could be treated same as the motion on a slope.

4 Trajectory Planning

We assume that the pushing force is constant because it makes stability analysis of walking simple. Even if the velocity of humanoid robots varies during one walking cycle, stable contact is kept by force control of arms. However, if the object velocity fluctuates, arms may come out of the range. The distance from

the robot to the object should be controlled so that the object comes in the range of arm movement. The walking velocity needs to be kept almost same as the object velocity. Hence, average walking velocity is given as an input for the trajectory planning in order to follow the object.

In this paper, we determine the walking stride for modifying the average walking velocity at the instant of the support leg changeover. Superscripts, *ref*, *cmd*, *res*, denote a reference value, a command value and a response value respectively.

If the walking velocity is controlled to be the same as that of the object, the humanoid robot can keep the moderate distance to the object. Stride command of the next step S^{cmd} which robot can follow the object is given from (12) and (13).

$$S^{cmd} = (\dot{x}_b^{av} + \dot{x}_a^{av})T \tag{11}$$

where $\dot{x}_a^{av}$ denotes average velocity of arms. The average walking velocity $\dot{x}_b^{av}$ is calculated as follows:

$$\dot{x}_b^{av} = \frac{S}{T}. \tag{12}$$

where S, T represent the current walking stride and the constant walking cycle respectively.

The base position of the lower body makes cyclic motion during a walking cycle if the object velocity is constant. Therefore the tip positions of the arms come back to the same standard point x_a^{std} at each changeover. Displacement from cyclic motion of the arms is represented by the differential between cyclic standard position of arms x_a^{std} at changeover and the response position of the arms x_a^{res} at changeover. Hence, average velocity of arms $\dot{x}_a^{av}$ is obtained by (13).

$$\dot{x}_a^{av} = \frac{x_a^{res} - x_a^{std}}{T} \tag{13}$$

Stride command can be calculated from (11). Next, it is necessary to generate the trajectory which robots can walk with the command value of the stride obtained by (11). Motion of the inverted pendulum is determined by reference value of landing point. Now, we define $\omega = \sqrt{\frac{g}{z_b}}$. (10) is rewritten as (14).

$$\ddot{x}_b(t) = \omega^2(x_b(t) - x_{sp}^{ref}(t)) - \frac{f_b}{m}. \tag{14}$$

(14) is digitalized for each step as (15) so that reference value of landing point is input at changeover.

$$\begin{bmatrix} x_b(N+1) \\ \dot{x}_b(N+1) \\ x_{sp}^{ref}(N+1) \end{bmatrix} = \begin{bmatrix} c & \frac{s}{\omega} & 1-c \\ \omega s & c & -\omega s \\ 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} x_c(N) \\ \dot{x}_b(N) \\ x_{sp}^{ref}(N) \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} x_{sp}^{ref}(N+1) + \begin{bmatrix} \frac{1-c}{m\omega^2} \\ -\frac{s}{m\omega} \\ 0 \end{bmatrix} f_b \tag{15}$$

where N is number of steps. $s = \sinh(\omega T)$, $c = \cosh(\omega T)$. Motion of next step is determined by inputting the desired landing point from (15). However,

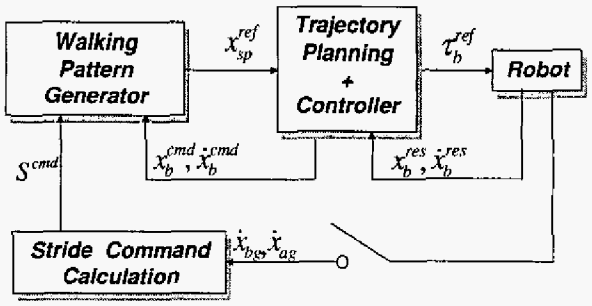

Fig. 5: Block diagram of walking pattern generator.

walking stability of the humanoid robot is not compensated if the command value of the stride of the next step obtained from (11) is input directly. Therefore we apply the walking pattern generator[6] which attains the global stability of the humanoid robot. Applying this method, we can define the next reference value of landing point as follows:

$$x_{sp}^{ref}(N) = f_0(x_b(N) - x_{sp}^{cmd}(N)) + f_1\dot{x}_b(N) + f_2(x_{sp}^{ref}(N) - x_{sp}^{cmd}(N)) + x_{sp}^{cmd}(N). \tag{16}$$

Command value of the landing point x_{sp}^{cmd} is given by using the obtained stride command from (11) and described as (17).

$$x_{sp}^{cmd}(N) = x_{sp}^{cmd}(N-1) + S^{cmd} \tag{17}$$

The gains which stabilize the motion of the inverted pendulum represented by (14) is obtained by substituting (16) to (15). The gains f_0, f_1, f_2 are calculated by desired pole $G(-1 < G < 1)$ as follows:

$$\begin{aligned} f_0 &= \frac{G^3 - 3G^2 + (2c-1)(3G+2c) - 1}{2(c-1)} \\ f_1 &= \frac{-G^3 + 3G^2 + (2c+1)(3G+2c) - 1}{2\omega s} \\ f_2 &= -3G - 2c. \end{aligned} \tag{18}$$

Block diagram of walking pattern generator is shown in Fig. 5.

Desired stride to follow an object is calculated by the response of the arm position which was described as (11). Moreover, note that global stability of humanoid robot is obtained by walking pattern generator.

5 Whole Control System

Whole control system is shown in Fig. 6. This control system is composed of two subsystems, upper body controller and lower body controller.

Legs are controlled by PD controller with full order observer[4].

$${}^b\ddot{x}_f^{ref} = K_{pl}(x_f^{cmd} - x_f^{res}) + K_{vl}(\dot{x}^{cmd} - \dot{x}^{res}) \tag{19}$$

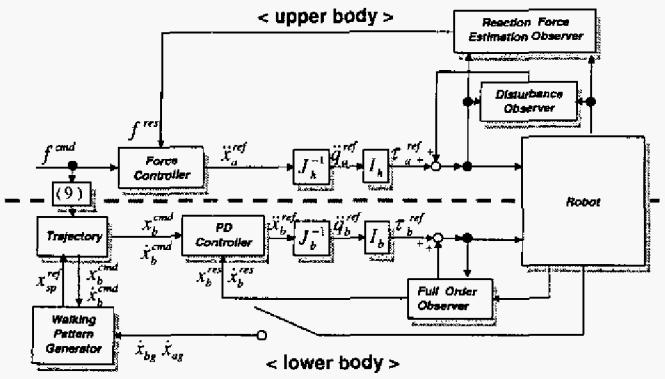

Fig. 6: Whole control system.

5.1 Control of upper body

A force control with constant command is applied for the upper body. $\ddot{x}_b^{ref}$, the reference acceleration input to the robot, is given from the force controller as (20). In fact, compliance control with low position feedback gain is applied in the simulations. The actual input is derived by (21).

$$\ddot{x}_b^{ref} = K_{fa}(f^{cmd} - f^{res}) \tag{20}$$

$$\ddot{x}_b^{ref} = K_{pa}(x_a^{cmd} - x_a^{res}) + K_{va}(\dot{x}_a^{cmd} - \dot{x}_a^{res}) + K_{fa}(f^{cmd} - f^{res}) \tag{21}$$

here, K_{fa} is the force gain of the upper body control and K_{pa}, K_{va} are the position gain and velocity gain respectively. x_a denotes the arm tip position. The traveling direction is only considered in these equations. The arms are controlled with PD controller only on the other directions.

Constant value of the force command f^{cmd} is given so that the force input counters the friction force on pushing object. f^{cmd} is derived related to the average velocity $\dot{x}_{bg}$.

$$f^{cmd} = D_w \dot{x}_{bg} \tag{22}$$

here, D_w is the friction coefficient of wheel on the object.

5.2 Error on force control

Force response on the arm tip varies although the upper body is controlled so as to concentrate to the force control. This problem is on ground of many elements such as inertia torque of the arm links, position error on the body link control and so on. This error on force control may degrade the walking stability of a humanoid robot. The acceptable range of force error is derived as follows.

LIPM is a method to design the body link trajectory so that no torque occurs at the supporting point of the inverted pendulum. However, it is able to compensate the force error on the arm tip applying the torque on the supporting point. The range of torque on the supporting point is figured out from (23) considering the walking stability.

$$l_t mg \leq \tau_{support} \leq l_h mg \tag{23}$$

here, l_h is the distance from supporting point to the heel edge of supporting foot. l_t is the distance from the supporting point to the foot tip edge of supporting foot. $\tau_{support}$ is the torque on supporting point. It is assumed that the supporting point is on the ground.

The range of the force control error is derived from the limit of compensation torque on supporting point as follows.

$$\frac{mgl_h}{z_a} \leq f_{error} \leq \frac{mgl_t}{z_a} \tag{24}$$

here, f_{error} is the error on force control.

Table 2: Parameters of the pushing object(Simulation).

Object mass	$M_o[kg]$	0.5
Damping coefficient of contact	$D_o[Ns/m]$	20.0
Spring constant of contact	$K_o[N/m]$	50.0
Damping coefficient of wheels	$D_w[Ns/m]$	0.5

Table 3: Control parameters(Simulation).

Position gain of arms	K_{pa}	10.0
Velocity gain of arms	K_{va}	20.0
Force gain of arms	K_{fa}	20.0
Position gain of legs	K_{pleg}	2500.0
Velocity gain of legs	K_{vleg}	100.0
Walking stride	$S^{cmd}[m]$	0.06
Walking cycle	$T[sec]$	0.5

6 Simulation

A simulation was executed to verify the validity of the suggested method. Pushing object was put 0.38 m ahead of the robot body while the arm tip comes 0.37 m ahead.

The parameters of pushing object are shown in Table 2. The contact force between the arm tips and the object is figured out applying the collision model of spring-mass-damper. On the other hand, the control parameters are shown in Table 3. Compliance control is applied to the upper body control. The position gain K_{pa} is set low so that its behavior becomes similar to a force control.

Fig. 7 shows the trajectories of the body, the arm tip, the object and the ZMP. It is able to confirm that the pushing task was properly executed from the result that the object went forward closely followed with the arm tips. Steps were modified for the robot to keep the moderate distance with the obstacle. The robot could

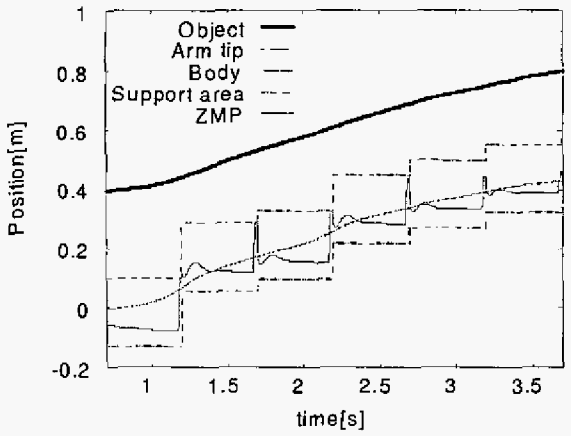

Fig. 7: Trajectories of body, arm tip, object and ZMP during pushing motion.

keep its stable locomotion under the existence of the pushing force.

Fig.8 shows the velocity responses of the arm tip and the base link. The force control with constant force input realized pushing motion with smooth object trajectory. The object velocity variation became small compare to the body velocity.

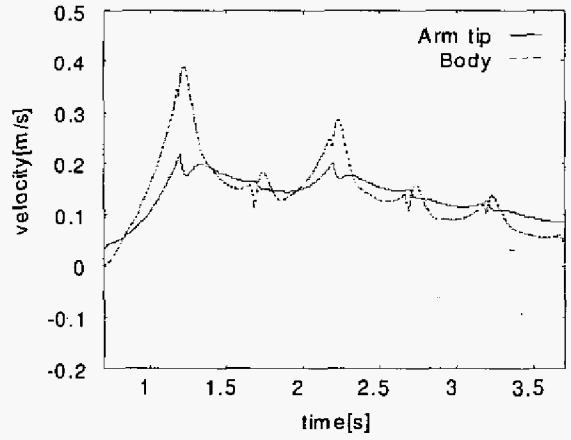

Fig. 8: Velocity of body and arm tip during pushing motion.

7 Conclusion

In this paper, the control method for pushing motion by humanoid robot considering the dynamic walking locomotion has been proposed. Humanoid robot is controlled by two subsystems, upper body controller and lower body controller. Force control was applied to the arms in order to generate the constant pushing force to the obstacle . Walking pattern generator was introduced so as to assure the dynamic walking stability during the pushing motion. With these methods, the robot became able to achieve pushing motion in human environment. This will be the typical example of the task performance of the humanoid robot as a replacement of human. The validity of proposed method is confirmed by simulation.

In this study, stride command is modified at the moment of the changeover. However, if the movement of the object changes significantly, the arms will move out of the range. For further study, we may deal with the trajectory planning so that the robot can modify the stride immediately.

Acknowledgements

This work is supported in part by a Grant in Aid for the 21st century Center Of Excellence from the Ministry of Education, Culture, Sport, Science, and Technology in Japan.

References

[1] K. Harada, S. Kajita, K. Kaneko and H. Hirukawa, "Pushing Manipulation by Humanoid considering Two-Kinds of ZMPs", IEEE Proc. Int. Conf. on R & A, pp. 1627-1632, 2003

[2] H. Yoshida, K. Inoue, T. Arai and Y. Mae, "Mobile Manipulation of Humanoid Robots -Optimal Posture for Generating Large Force Based on Statics-", IEEE Proc. Int. Conf. on R & A, pp. 1856-1863, 2002

[3] Y. Hwang, A. Konno and M. Uchiyama, "Whole Body Cooperative Tasks and Static Stability Evaluations for Humanoid Robot", IEEE Proc. Int. Conf. on Intelligent R & S, pp. 1901-1906, 2003

[4] M. Morisawa, Y. Fujimoto, T. Murakami and K. Ohnishi, "A Walking Pattern Generation for Biped Robot with Parallel Mechanism by Considering Contact Force", IEEE Int. Conf. on IECON'01, Vol. 71, No. 2, pp. 2184-2189, 2001

[5] N. Nishikawa, Y. Fujimoto, T. Murakami and K. Ohnishi, "Variable Compliance Control for 3 Dimensional Biped Robot Considering Environmental Fluctuations", The Institute of Electrical Engineerings of Japan (IEEJ) Trans. on Industry Applications, Vol. 119-D, No. 12, pp. 1507-1514, 1999 (in Japanese)

[6] Fujimoto. Y, "Simulation of an Autonomous Biped Walking Robot Including Environmental Force Interaction", IEEE R & A Magazine, pp. 33-42, June, 1998

[7] Kajita. S, Tani. K, "Study of dynamic biped locomotion on rugged terrain-derivation and application of the linear inverted pendulum mode", IEEE Proc. Int. Conf. on R & A, Vol. 2, pp. 1405-1411, 1991