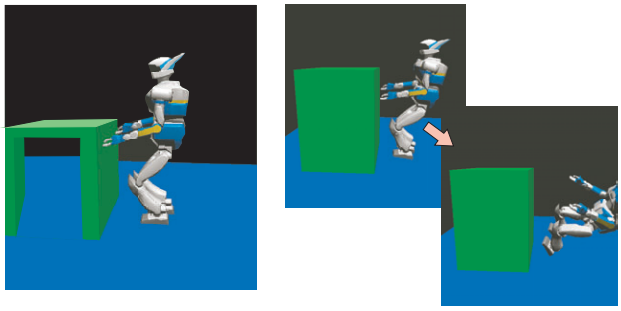
Proceedings of the 2004 IEEE
International Conference on Robotics & Automation
New Orleans, LA • April 2004

Real-Time Planning of Humanoid Robot's Gait for Force Controlled Manipulation

Kensuke Harada, Shuuji Kajita, Fumio Kanehiro, Kiyoshi Fujiwara,
Kenji Kaneko, Kazuhito Yokoi, and Hirohisa Hirukawa
Humanoid Research Group, Intelligent Systems Institute
National Institute of Advanced Industrial Science and Technology(AIST)
1-1-1 Umezono, Tsukuba, Ibaraki 305-8568, JAPAN

***Abstract*— This paper proposes a new style of manipulation by a humanoid robot. Focusing on the task of pushing an object, the foot placement of it is planed in real-time according to the result of manipulation of an object. By controlling the arms using the impedance control, a humanoid robot can push an object stably regardless of the mass of an object. If an object is heavy, a humanoid robot pushes an object with walking slowly, and vice versa. Also, for planning the gait in real-time, we newly propose an analytical method where the newly calculated trajectory of the robot motion is smoothly connected to the current one. The effectiveness of the proposed method is confirmed by simulation and experiment.**



(a) Manipulation of a light object (b) Manipulation of a heavy object

Fig. 1. Pushing Manipulation by a Humanoid Robot

I. INTRODUCTION

A human makes arms and legs cooperate and realizes various tasks. Since the kinematical structure of a humanoid robot is similar to that of a human, a humanoid robot is expected to work instead of a human in our daily life. Although the manipulation is necessary for a humanoid robot to work in the real environment, there are not many researches on the manipulation by a humanoid robot. Based on this consideration, we focus on the manipulation of an object by a humanoid robot in this research.

As an example of the manipulation of an object by a humanoid robot, let us consider pushing a large object placed on the floor. We can assume two styles of pushing manipulation by a humanoid robot, i.e., (1) The position control based pushing manipulation, and (2) The force control based pushing manipulation. As for (1), the gait pattern is determined in advance of the actual motion, and it is not changed depending on the pushing force information measured by the force/torque sensors attached at the tip of the arms. Although the position control based method can be easily implemented, the robot may not keep the dynamical balance if the weight of the object or the friction coefficient between the object and the floor is changed. Fig.1 shows an image of the position control based method. If the object is light, the robot can easily move the object with keeping the dynamical balance. However, if the object becomes very heavy, the walking velocity of the robot becomes higher than the velocity of the object, and the robot will finally fall down. On the other hand, as for (2), the gait pattern is adaptively changed depending on the force sensor information at the tip of the arms. By using the force control based method, we can expect that the robot can keep the dynamical balance even if the weight of the object or the friction coefficient between the object and the floor is changed.

While we proposed the position control based method in our previous paper[1], we newly consider the force control based method in this paper. Although we focus on the pushing manipulation, our proposed force control based method can be applied to several arm/leg cooperated tasks of a humanoid robot such as opening/closing a door by holding the knob, or leaning on a wall. The force control based method is realized by separating the pushing phase from the stepping phase. The humanoid robot pushes an object in the pushing phase by controlling the force applied at the tip of the arms without stepping. And, the humanoid robot steps without pushing an object where the length of each step is determined by the amount of pushing an object in the pushing phase.

For realizing such style of pushing manipulation, the gait pattern should be planned in real-time. However, the real-time planning of the gait pattern has been considered to be difficult due to the following reasons: (1)In most of the existing method, for a given ZMP trajectory, the trajectory of COG(Center of Gravity) has been calculated by using numerical iterations., and (2)The trajectory of COG is obtained by solving the two-point boundary value problem of a differential equation. Therefore, it is difficult to obtain the smooth trajectory of COG if the target ZMP is changed in real-time. As for (1), we consider analytically obtaining the COG trajectory for the given ZMP. And, as for (2), we newly develop a method for smoothly connecting the newly calculated COG trajectory to an existing one.

0-7803-8232-3/04/$17.00 ©2004 IEEE

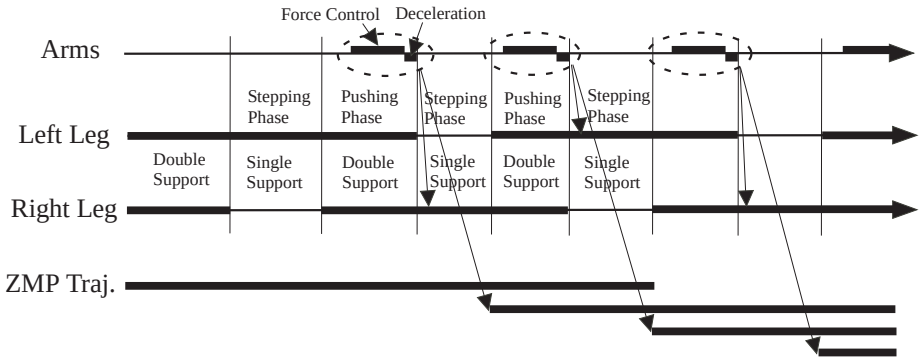


Fig. 2. Time Chart of the Proposed Algorithm

After discussing on the previous works in Section 2, we show the overview of the proposed method in Section 3. In Section 4, we discuss a method for controlling the arms of a humanoid robot. In Section 4, we show a method for planning the gait of a humanoid robot in real-time. Lastly, in Section 5 and 6, we show simulation and experimental results.

II. Related Works

Manipulation by Humanoid Robot:
As for the manipulation of an object by a humanoid robot, the authors[1] proposed the position control based pushing manipulation. The authors[2] also studied the ZMP during the manipulation of an object. As an application study of HRP[13], Yokoyama et al.[3] realized by the humanoid robot HRP-2 carrying a panel with a human. Inoue et al.[4] and Nishihara et al.[5] determined the foot position maximizing the manipulability of the arms. Hwang et al.[6] also studied the static relationship between the hand reaction force and the ZMP position. Yokokohji et al.[7] studied the posture of a humanoid robot whose hands contact with an environment. However, there has been no research on a humanoid robot adaptively changing the gait pattern according to the hand reaction force.

Real-time Generation of Gait Pattern:
In most of the previous researches[8], [9], [12], the gait pattern for a biped robot has been generated offline in advance of the actual motion. On the other hand, recently, some researchers research the on-line generation or on-line modification of the gait pattern for a given ZMP trajectory. Lim et al.[10] considered combining the unit gait patterns generated offline. Nishiwaki et al.[11] proposed a method for modifying the gait pattern on-line. On the other hand, the feature of the proposed on-line modification method for gait pattern is that the gait pattern can be calculated very fast since the COG trajectory is analytically obtained, and that newly calculated COG trajectory is smoothly connected to the current one.

III. Separation of Stepping/Pushing Phases

Let us assume that both of the hands of a humanoid robot contact with an object. Let the position of the ZMP, the COG, and each hand be $p_{zmp} = [x_{zmp}\ y_{zmp}\ z_{zmp}]^T$, $p_G = [x_G\ y_G\ z_G]^T$, $p_{Hj} = [x_{Hj}\ y_{Hj}\ z_{Hj}]^T$ $(j = 1, 2)$, respectively. Let the hand reaction force and the total mass of the robot be $\boldsymbol{f}_j = [f_{xj}\ f_{yj}\ f_{zj}]^T$ $(j = 1, 2)$ and M, respectively. The relationship between the ZMP position and the hand reaction force is expressed by[1]:

$$x_{zmp} = \frac{-\dot{L}_{Gy} + M x_G(\ddot{z}_G + g) - M(z_G - z_{zmp})\ddot{x}_G}{M(\ddot{z}_G + g)} - \sum_{j=1}^{2} \frac{(z_{Hj} - z_{zmp}) f_{xj}}{M(\ddot{z}_G + g)}, \quad (1)$$

$$y_{zmp} = \frac{\dot{L}_{Gx} + M y_G(\ddot{z}_G + g) - M(z_G - z_{zmp})\ddot{y}_G}{M(\ddot{z}_G + g)} - \sum_{j=1}^{2} \frac{(z_{Hj} - z_{zmp}) f_{yj}}{M(\ddot{z}_G + g)}, \quad (2)$$

where $\boldsymbol{L}_G = [L_{Gx}\ L_{Gy}\ L_{Gz}]^T$ denotes the angular momentum of the robot about the COG. Eqs.(1) and (2) show that the ZMP position changes depending on the reaction force applied at the hands. The robot can keep the dynamical balance if the ZMP is included in the foot supporting area. However, since it is difficult to predict the amount of hand reaction force in advance of the actual motion, the ZMP may lie on the edge of the foot supporting area if unexpected reaction force is applied at the hands. Therefore, we consider pushing the object during the double support phase where both of the feet contact the floor and where the convex hull of the foot supporting area becomes wide. And, the robot steps without pushing an object.

The timing chart of the proposed algorithm is shown in Fig.2. In the pushing phase, the robot pushes the object where the reaction force at the tip of the arm is controlled. In the stepping phase, the amount of step within the sagittal plane is set same as the amount of pushing the object in the pushing phase. Also, by using the amount of pushing, the desired ZMP trajectory is recalculated, and is connected to the previous ZMP trajectory. For realizing the desired ZMP trajectory, the COG trajectory is calculated by using the algorithm shown in Section 5. By using this algorithm for pushing manipulation by a humanoid robot, the robot can push the object with keeping the dynamical balance with the knowledge of neither

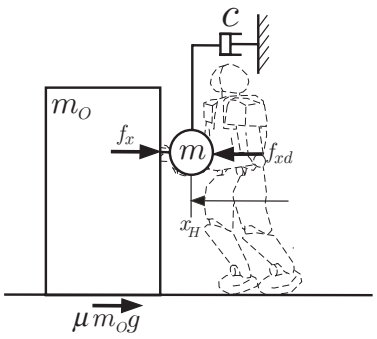


Fig. 3. Arm Impedance Control

the weight of the object nor the friction coefficient between the object and the floor. The detail of the proposed algorithm is shown from the next section.

IV. Arm Impedance Control

In the pushing phase, the reaction force at the tip of the arm is controlled by using the impedance control law. Let m and c be the desired mass and damping coefficient, respectively. And, let f_{xd} be the desired value of f_x. The target impedance is defined by

$$m\ddot{x}_H + c\dot{x}_H = f_{xd} - f_x \tag{3}$$

where we set $x_H = x_{H1} = x_{H2}$ and $f_x = f_{x1} = f_{x2}$. x axis is defined to be the walking direction of the robot. The overview of the impedance control used in the pushing manipulation is shown in Fig. 3. When $f_{xd} \leq \mu m_O g$, the object will not move since f_x balances with f_{xd}. On the other hand, when $f_{xd} > \mu m_O g$, the object moves where the acceleration of the object depends on the difference between f_{xd} and f_x. Under the condition where the target impedance is realized, the object with light weight moves with high acceleration. The heavier the weight of the object becomes, the lower the acceleration of the object will become. And, finally the object will not move if the weight of the object becomes very heavy. Therefore, by using this algorithm, we can expect that the robot can keep the dynamical balance without taking the mass of the object into consideration.

For a position controlled robot, the target impedance is approximately realized by controlling the tip position of the arm by

$$x_H(t+\Delta t) = \frac{\Delta t^2}{m}\left(f_{xd} - f_x(t) - c\frac{x_H(t) - x_H(t-\Delta t)}{\Delta t}\right) + 2x_H(t) - x_H(t-\Delta t), \tag{4}$$

where Δt denotes the sampling period for controlling the robot.

V. Connection of COG Trajecotries

In this section, we will discuss a method for planning the gait of a humanoid robot in real-time according to the hand reaction force information. Let us focus on the motion of a humanoid robot within the sagittal plane. As shown in Fig.4,

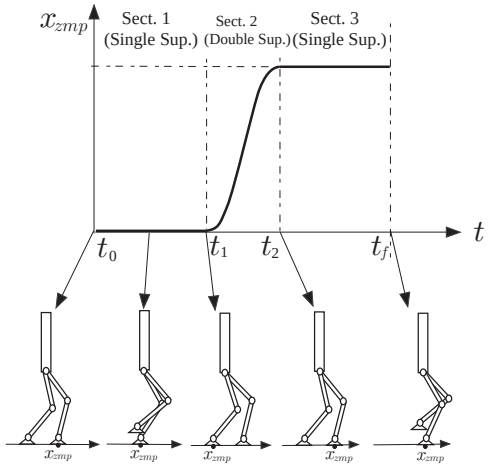


Fig. 4. ZMP Trajectory

let us assume that the desired ZMP trajectory is given by a spline function:

$$x_{zmp}^{(j)} = \sum_{i=0}^{n} a_i^{(j)}(t - t_{j-1})^i, \quad t_{j-1} \leq t \leq t_j, \quad j = 1, \cdots, m \tag{5}$$

where $x_{zmp}^{(j)}$ denotes the ZMP trajectory belonging to the j-th section. Since the robot does not push the object in the stepping phase, we do not consider the effect of hand reaction force. By approximating the robot by a inverted pendulum whose height is constant, the relationship between the ZMP and the COG positions of the robot is given by

$$x_{zmp}^{(j)} = x_G^{(j)} - \frac{z_G - z_{zmp}}{g}\ddot{x}_G^{(j)}, \quad j = 1, \cdots, m. \tag{6}$$

Substituting eq.(5) into eq.(6) and solving with respect to $x_G^{(j)}$, we obtain the COG trajectory as

$$x_G^{(j)} = V^{(j)}\cosh(T_c(t - t_{j-1})) + W^{(j)}\sinh(T_c(t - t_{j-1})) + \sum_{i=0}^{n} A_i^{(j)}(t - t_{j-1})^i, \quad j = 1, \cdots, m \tag{7}$$

$$a_i^{(j)} = A_i^{(j)} - \frac{1}{T_c^2}(i+1)(i+2)A_{i+2}, \quad i = 0, \cdots, n-2 \tag{8}$$

$$a_i^{(j)} = A_i^{(j)}, \quad i = n-1, n \tag{9}$$

where $T_c = \sqrt{g/(z_G - z_{zmp})}$, and $V^{(j)}$ and $W^{(j)}$ denote the constants defined in the following paragraph to obtain the smooth trajectory of the COG.

For the purpose of changing the gait pattern in real-time, we consider connecting the newly calculated COG trajectory to the current one. As shown in Fig.5, at the beginning of the double support phase, the new ZMP trajectory whose distance of each step is same as the amount of pushing an object in the pushing phase is calculated. By using the new ZMP trajectory, the COG trajectory is calculated, and is smoothly connected to the current one.

Here, eq.(7) includes $\sinh$ and $\cosh$ in its homogenous part. To prevent the solution of eq.(7) to diverge as time goes by, $V^{(j)}$ and $W^{(j)}$ in eq.(7) should be determined by considering the two-point boundary value problems where the initial and the terminal conditions for the position of COG are given. Here, we note that there are $2m$ unknowns in eq.(7), i.e., $V^{(j)}$ and $W^{(j)}$ $(j = 1, \cdots, m)$. To determine these unknowns, we set the following m boundary conditions:

Initial Condition (Position of COG)

$$x_G^{(1)}(t_0) = V^{(1)} + A_0^{(1)} \tag{10}$$

Connection of two Sections (Position/Velocity of COG)

$$j = 1, \cdots, m-1$$

$$V^{(j)} \cosh(T_c(t_j - t_{j-1})) + W^{(j)} \sinh(T_c(t_j - t_{j-1})) + \sum_{i=0}^{n} A_i^{(j)} (t_j - t_{j-1})^i = V^{(j+1)} + A_0^{(j+1)} \tag{11}$$

$$V^{(j)} T_c \sinh(T_c(t_j - t_{j-1})) + W^{(j)} T_c \cosh(T_c(t_j - t_{j-1})) + \sum_{i=1}^{n} i A_i^{(j)} (t_j - t_{j-1})^{i-1} = W^{(j+1)} T_c + A_1^{(j+1)} \tag{12}$$

Terminal Condition (Position of COG)

$$\bar{x}_G^{(m)}(t_f) = \tilde{V}^{(m)} \cosh(T_c(t_f - t_{m-1})) + \tilde{W}^{(m)} \sinh(T_c(t_f - t_{m-1})) + \sum_{i=0}^{n} \tilde{A}_i^{(m)} (t_f - t_{m-1})^i \tag{13}$$

By considering the two-point boundary value problem, the initial velocity of COG cannot be given as a boundary condition. Therefore, to consider connecting the newly calculated COG trajectory to the current one, discontinuity of the velocity will occur. To ensure the continuity of the velocity of COG, the ZMP trajectory of the first section is recalculated. By setting $A_i^{(1)}$ $(a_i^{(1)})$ $(i = 0, \cdots, n)$ as unknowns constants, the following boundary conditions are added

Initial Condition (Position of ZMP)

$$x_{zmp}^{(1)}(t_0) = a_0^{(1)} \tag{14}$$

Terminal Condition for the 1st Section (Position of ZMP)

$$x_{zmp}^{(2)}(t_1) = \sum_{i=0}^{n} a_i^{(1)} (t_1 - t_0)^i \tag{15}$$

Initial Condition (Velocity of COG)

$$\dot{x}_G^{(1)}(t_0) = W^{(1)} T_c + A_1^{(1)} \tag{16}$$

When $n = 2$ for the first section, the $2m + n + 1$ unknowns $A_0^{(1)}, \cdots, A_n^{(1)}$, $V^{(1)}, W^{(1)}, \cdots, V^{(m)}, W^{(m)}$ can be determined by using eqs.(10), $\cdots$, (16) as follows:

$$\boldsymbol{y} = \boldsymbol{Z}^{-1} \boldsymbol{w}, \tag{17}$$

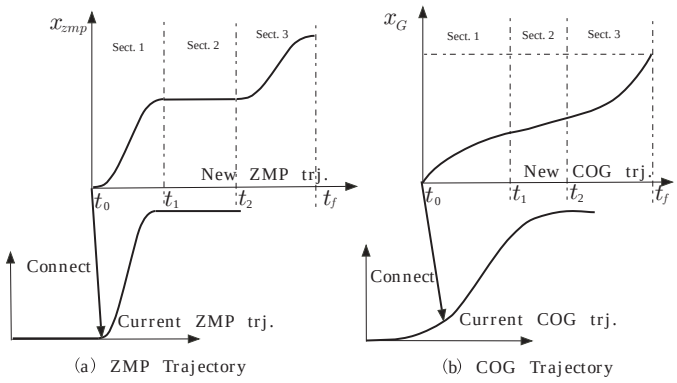


Fig. 5. Change of ZMP Trajectory

where

$$\boldsymbol{y} = [A_0^{(1)} \; A_1^{(1)} \; A_2^{(1)} \; V^{(1)} \; W^{(1)} \; \cdots \; V^{(m)} \; W^{(m)}]^T$$

$$\boldsymbol{Z} = \begin{bmatrix} \boldsymbol{Z}_0 & \boldsymbol{0} & \boldsymbol{0} & \cdots & & \boldsymbol{0} \\ \boldsymbol{Z}_{11} & \boldsymbol{Z}_{12} & \boldsymbol{0} & \cdots & & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{Z}_2 & \boldsymbol{0} & \cdots & & \boldsymbol{0} \\ & \boldsymbol{0} & \ddots & & & \vdots \\ \vdots & \vdots & & \boldsymbol{Z}_j & & \\ & & & & \ddots & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{0} & \cdots & & \boldsymbol{0} & \boldsymbol{Z}_{m-1} \\ \boldsymbol{0} & \boldsymbol{0} & \cdots & & \boldsymbol{0} & \boldsymbol{z}_{m-1} \end{bmatrix}$$

$$\boldsymbol{Z}_0 = \begin{bmatrix} 1 & 0 & -2/T_c^2 \\ 1 & t_1 - t_0 & (t_1 - t_0)^2 - 2/T_c^2 \end{bmatrix}$$

$$\boldsymbol{Z}_{11} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 1 & (t_1 - t_0) & (t_1 - t_0)^2 \\ 0 & 1 & 2(t_1 - t_0) \end{bmatrix}$$

$$\boldsymbol{Z}_{12} = \begin{bmatrix} 1 & 0 \\ 0 & T_c \\ \cosh(T_c(t_1 - t_0)) & \sinh(T_c(t_1 - t_0)) \\ T_c \sinh(T_c(t_1 - t_0)) & T_c \cosh(T_c(t_1 - t_0)) \end{bmatrix}$$

$$\boldsymbol{Z}_j = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ -1 & 0 \\ 0 & -T_c \end{bmatrix}$$

$$\boldsymbol{Z}_j = \begin{bmatrix} \cosh(T_c(t_j - t_{j-1})) & \sinh(T_c(t_j - t_{j-1})) \\ T_c \sinh(T_c(t_j - t_{j-1})) & T_c \cosh(T_c(t_j - t_{j-1})) \\ -1 & 0 \\ 0 & -T_c \end{bmatrix}$$

$$\boldsymbol{z}_{m-1} = [0 \; 0 \; \cosh(T_c(t_f - t_{m-1})) \; \sinh(T_c(t_f - t_{m-1}))]$$

$$\boldsymbol{w} = [\bar{x}_{zmp}^{(1)}(t_0) \;\; \bar{x}_{zmp}^{(2)}(t_1) \;\; \bar{x}_G^{(1)}(t_0) \;\; \dot{\bar{x}}_G^{(1)}(t_0) \;\; \tilde{A}_0^{(2)} \;\; \tilde{A}_1^{(2)} \;\; \cdots \tilde{A}_0^{(j+1)} - \sum_{i=0}^{n} \tilde{A}_i^{(j)} (t_j - t_{j-1})^i \;\; \tilde{A}_1^{(j+1)} - \sum_{i=1}^{n} i \tilde{A}_i^{(j)} (t_j - t_{j-1})^{i-1}$$

$$\cdots \bar{x}_G^{(m)}(t_f) - \sum_{i=0}^{n} \tilde{A}_i^{(m)}(t_f - t_{m-1})^i]^T$$

Here, we numerically confirmed that the matrix $\boldsymbol{Z}$ in eq.(17) is invertible. We note that, by using the proposed method, the COG trajectory can be calculated very fast. Only the time-cunsuming calculation is the inverse of the matrix $\boldsymbol{Z}$. Assuming the ZMP trajecotry for 3 steps and $m = 9$, the size of the matrix $\boldsymbol{Z}$ becomes 20×20. By using the PC whose processer is XEON 2.2Mhz, the calculation of $\boldsymbol{Z}^{-1}$ takes about 0.3[msec] which is short enough. We also note that the idea of connecting the new COG trajectory to the current one is originally proposed by Nishiwaki et al.[11]. Different from their approach, we used the analytical solution of COG position which can be calculated fast enough, and we consider the smooth connection between two COG trajectories. In the simulation, in addition to the Initial Condition for Position of ZMP (eq.(14)) and the Terminal Condition for ZMP Position of the 1st Section (eq.(15)), we also consider the Intial Condition for the Velocity of ZMP and the Terminal Condition for ZMP velocity of the 1st Section.

VI. Simulation

A. Controller

The controller used in the simulation and the experiment is shown in Fig.6. The positions of the hands, the COG, and the feet are calculated by using the proposed algorithm. By using these position data as inputs, the joint angle of the humanoid robot is calculated by using the Resolved Momentum Control[16]. The humanoid robot HRP-2 is driven by the joint angle command. Since the error of ZMP position may cause due to the unknown error of physical parameters or due to the external disturbance, the stabilizing controller which simultaneously controls the ZMP position and the body posture is installed.

B. Results

We performed simulation by using OpenHRP[14], [15]. We set the parameters of the arm impedance control as $m = 10.0$, $c = 500.0$, and $f_{xd} = 70$[N]. We also set the time for the pushing and the stepping phases as $T_p = 2.7$[sec] and $T_s = 0.8$[sec], respectively. The amount of pushing is limited to 0.17 [m]. The friction coefficient at each contact point is set as $\mu = 0.5$.

Simulation results are shown in Figs.7,8,9, and 10. Figs.7 and 8 show the case where the weight of the object is $m_o = 5$[kg], and Figs.9 and 10 show the case where the $m_o = 15$[kg]. As shown in the figures, the robot can keep the dynamical balance even if the weight of the object is changed. If the object is light, the humanoid robot pushes an object with walking fast. And the heavier the object becomes, the slower humanoid robot walks.

VII. Experiment

Then we performed experiment. We used the humanoid robot HRP2[17] whose height and weight are 1.54[m] and 58 [kg], respectively. As an object, we used a table whose weight is about 10[kg]. The result of experiment is shown in Fig.11. While the motion of the table is disturbed between $t = 12$[sec] and $t = 20$[sec], we can see that the robot can keep balance by adaptively changing the gait pattern depending on the amount of pushing an object.

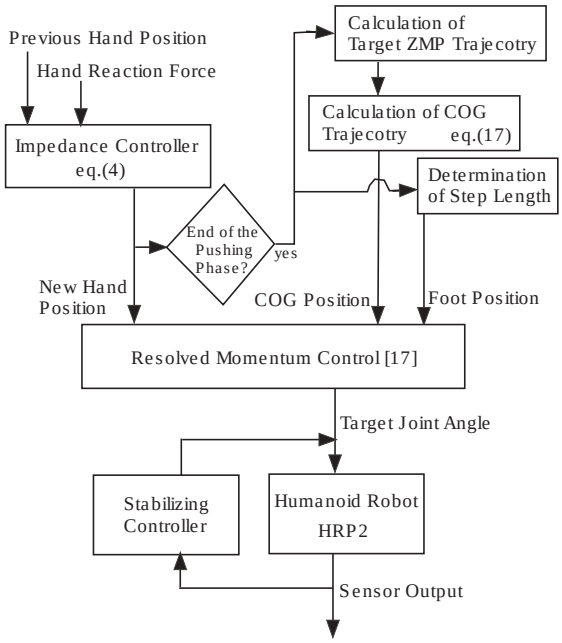


Fig. 6. Block Diagram of Controller

VIII. Conclusion

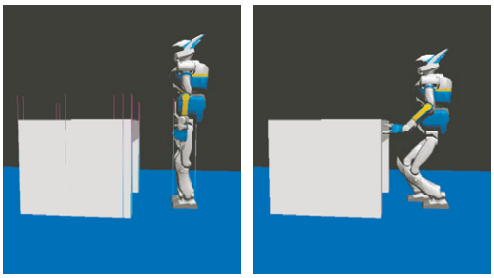
In this paper, we proposed a new style of manipulation by a humanoid robot where the gait is planned in real-time based on the information from the force controlled arms. Focusing on the pushing manipulation, we have succeeded in pushing an object stably by a humanoid robot even if the weight of the object is changed. The effectiveness of the proposed method is confirmed by simulation and experiment.

To control the arms of HRP2, we used the position-control based impedance control. Here, due to the error comming from the descretization of the desired impedance, the error is caused in the actual impedance at the tip of the arms. As a fiture research topic, we will refine the performance of the impedance control.

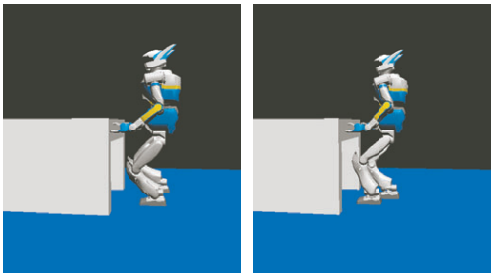
As for the gait planning, the planned ZMP trajectory will not be a straight line if the change of the foot position is very significant. The real-time gait generation for the significant change of the foot position, for example by increasing the order of the polynomial and imposing another constraint on the ZMP trajecotry, is considered to be our future research topic.

References

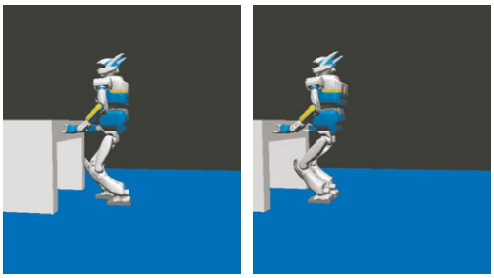
[1] K. Harada, S. Kajita, K. Kaneko, and H. Hirukawa: "*Pushing Manipulation by Humanoid considering Two-Kinds of ZMPs*", Proc. of IEEE Int. Conf. on Robotics and Automation, pp. 1627-1632, 2003.



(a) t=0.0 [sec] (b) t=5.4 [sec]

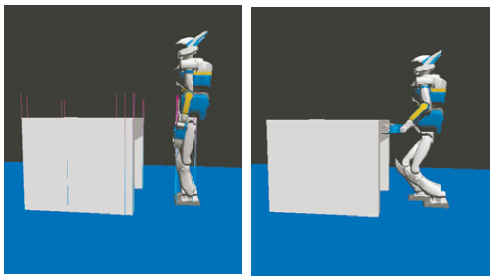


(c) t=10.8 [sec] (d) t=16.1 [sec]

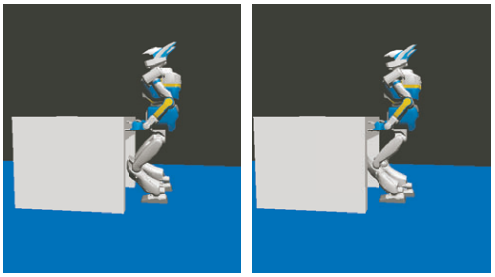


(e) t=21.6 [sec] (f) t=32.3 [sec]

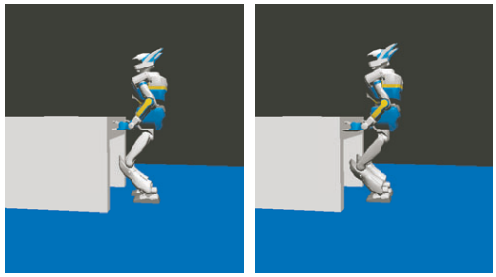
Fig. 7. Simulation Result($m_o = 5$[kg])



(a) t=0.0 [sec] (b) t=5.4 [sec]

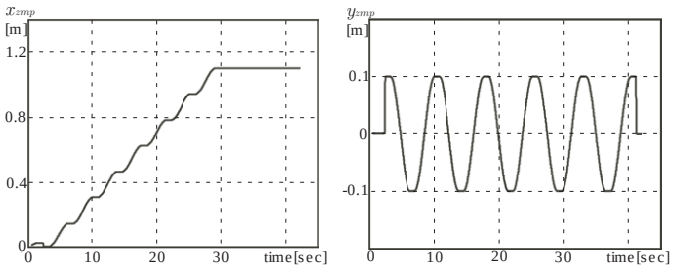


(c) t=10.8 [sec] (d) t=16.1 [sec]



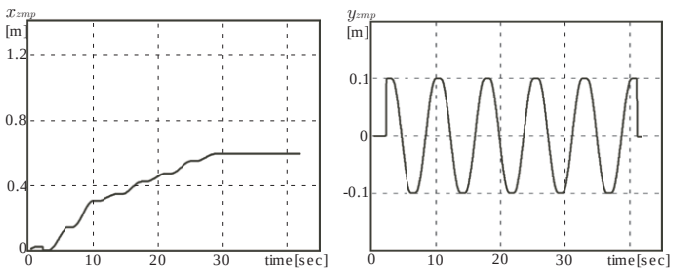
(e) t=21.6 [sec] (f) t=32.3 [sec]

Fig. 9. Simulation Result($m_o = 15$[kg])



(a) ZMP position in x direction (b) ZMP position in y direction

Fig. 8. ZMP Trajectories($m_o = 5$[kg])



(a) ZMP position in x direction (b) ZMP position in y direction

Fig. 10. ZMP Trajectories($m_o = 15$[kg])

[2] K. Harada, S. Kajita, K. Kaneko, and H. Hirukawa: "*ZMP Analysis for Arm/Leg Coordination*", Proc. of IEEE/RSJ Int. Conf. on Intelligent Robots and Systems, pp. 73-79, 2003.

[3] 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*", Proc. of IEEE Int. Conf. on Robotics and Automation, pp. 2985-2991, 2003.

[4] K. Inoue, H. Yoshida, T. Arai, and Y. Mae: "*Mobile Manipulation of Humanoids –Real-Time Control Based on Manipulability and Stability–*", Proc. of IEEE Int. Conf. on Robotics and Automation, pp. 2217-2222, 2000.

[5] Y. Nishihara, K. Inoue, T. Arai, and Y. Mae: "*Mobile Manipulation of Humanoid Robots –Control method for Accurate Manipulation–*", Proc. of IEEE/RSJ Int. Conf. on Intelligent Robots and Systems, pp. 1914-1919, 2003.

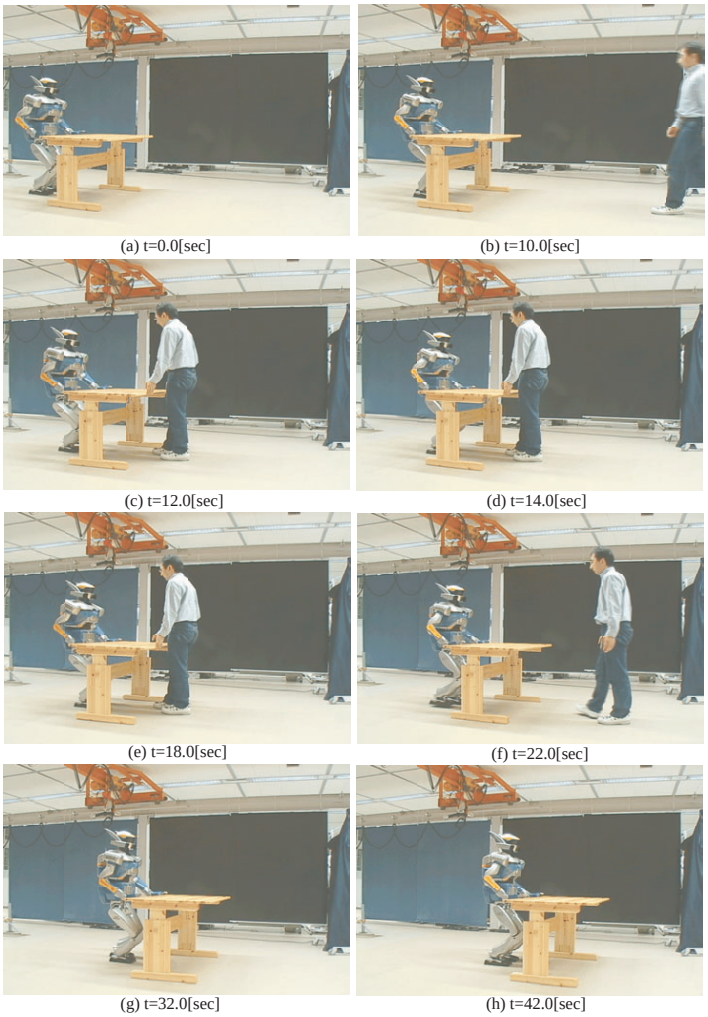
[6] Y. Hwang, A. Konno, and M. Uchiyama: "*Whole Body Cooperative Tasks and Static Stability Evaluations for a Humanoid Robot*", Proc. of IEEE/RSJ Int. Conf. on Intelligent Robots and Systems, pp. 1901-1906, 2003.

[7] Y. Yokokohji, S. Nomoto, and T. Yoshikawa: "*Static Evaluation of Humanoid Robot Postures Consrained to the Surrounding Environment through their Limbs*", Proc. of IEEE Int. Conf. on Robotics and Automation, pp. 1856-1863, 2002.

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

[9] S. Kajita, O. Matsumoto, and M. Saigo: "*Real-time 3D Walking Pattern Generation for a Biped Robot with Telescopic Legs*", Proc. of IEEE Int. Conf. on Robotics and Automation, pp. 2299-2036, 2001.

[10] H.-O. Lim, Y. Kaneshima, and A. Takanishi: "*Online Walking Pattern*



(a) t=0.0[sec] (b) t=10.0[sec]
(c) t=12.0[sec] (d) t=14.0[sec]
(e) t=18.0[sec] (f) t=22.0[sec]
(g) t=32.0[sec] (h) t=42.0[sec]

Fig. 11. Experimental Result

Generation for Biped Humanoid Robot with Trunk", 'Proc. of IEEE Int. Conf. on Robotics and Automation, pp. 3111-3116, 2002.

[11] K. Nishiwaki, S. Kagami, Y. Kuniyoshi, M. Inaba, and H. Inoue: "*Online Generation of Humanoid Walking Motion based on a Fast Generation Method of Motion Pattern that Follows Desired ZMP*", 'Proc. of IEEE/RSJ Int. Conf. on Intelligent Robots and Systems, pp. 2684-2689, 2002.

[12] R. Kurazume, T. Hasegawa, and K. Yoneda: "*The Sway Compensation Trajectory for a Biped Robot*", 'Proc. of IEEE Int. Conf. on Robotics and Automation, pp. 925-931, 2003.

[13] H. Inoue et al.: "*HRP: Humanoid Robotics Project of MITI*", Proc. of the First IEEE-RAS Int. Conf. on Humanoid Robots, 2000.

[14] F. Kanehiro et al.: "*Virtual humanoid robot platform to develop controllers of real humanoid robots without porting*", Proc. of IEEE/RSJ Int. Conf. on Intelligent Robots and Systems, 2001.

[15] H. Hirukawa et al.: "*OpenHRP: Open Architecture Humanoid Robot Platform*", Proc. of Int. Symp. on Robotics Research, 2001.

[16] S.Kajita et al.: "*Resolved Momentum Control: Humanoid Motion Planning based on the Linear and Angular Momentum*", Proc. of IEEE/RSJ Int. Conf. Intelligent Robots and Systems, pp. 1644-1650, 2003.

[17] K. Kaneko et al.: "*The Humanoid Robot HRP2*", Proc. of IEEE Int. Conf. on Robotics and Automation, 2004(To Appear).