2014 IEEE International Conference on Robotics & Automation (ICRA)
Hong Kong Convention and Exhibition Center
May 31 - June 7, 2014. Hong Kong, China

Manipulation Strategy Decision and Execution based on Strategy Proving Operation for Carrying Large and Heavy Objects

Masaki Murooka, Shintaro Noda, Shunichi Nozawa, Yohei Kakiuchi, Kei Okada and Masayuki Inaba

***Abstract*— In case that a robot carries large and heavy objects with unknown physical parameters such as mass automatically, the autonomous decision and execution of the manipulation strategy are necessary. The method to decide the proper strategy from the various candidates depending on the object is a difficult problem and not researched widely. We consider the operation as the mapping from the physical parameter space to the object motion space. Based on the concept of mapping, we define the strategy proving operation (SPO) for determination of strategy feasibility. We introduce two examples of SPO and construct the system for deciding strategy from lifting, pushing, and pivoting. Executing the strategy in the situation that physical parameters are not known is also necessary. We construct the generator and controller for the full-body manipulation, which can be employed regardless of strategy. The controller enables the robot to exert adequate force while keeping balance. We clarify the applicable scope of the proposed method and show that a life-sized humanoid decides the strategy and carries various large and heavy objects autonomously through the experiment.**

I. INTRODUCTION

Carrying objects is one of the tasks which robots are expected to perform in daily life. Especially, the high autonomy in the motion is necessary for human utility.

One of the difficulties of the carrying task is selection of the feasible strategy from a lot of choice, and the choice of these strategies depends on the physical parameters of the target object. For example if the object is heavy, large and high friction, then pivoting behavior might be appropriate one, whereas if the object is heavy, large but low friction, then the robot can push the object as shown in Fig.1. The problem is that measuring physical parameters of the objects is very difficult unless the robot manipulates them, however, the robot requires physical parameters in order to manipulate the object.

In this paper, we solve the above problem by Manipulation Strategy Decision (MSD) and Execution (MSE) based on Strategy Proving Operation (SPO). SPO is the object operation for determining the feasibility of the strategy based on mapping between the physical parameter space, the object motion space, and the strategy list. We construct the system for carrying objects by integrating MSD and MSE, which is applicable to general objects, environments, and robots (Fig.2). The features of this system are following two points: (i) deciding strategy autonomously and (ii) requiring no physical parameters of the objects. We confirm the effectiveness of our method in an experiment in which a humanoid robot carries large furniture autonomously.

M. Murooka, S. Noda, S. Nozawa, Y. Kakiuchi, K. Okada and M. Inaba are with Department of Mechano-Infomatics, The University of Tokyo, 7-3-1 Hongo, Bunkyo-ku, Tokyo 113-8656, Japan murooka at jsk.t.u-tokyo.ac.jp

Pushing Pivoting

Fig. 1. Objects Manipulation by a Robot.

II. CARRYING LARGE AND HEAVY OBJECTS WITH A ROBOT

A. Problem Establishment

In this section, we organize the premise and clarify the definition of the problem for carrying heavy objects with the robot.

1) Parameters of Objects, Environment, and Robot: Fig.3 shows the parameters of the three element: Objects (O), Environment (E), and Robot (R). Let Θ_{O}, Θ_{E}, and Θ_{R} be the parameters of each element. The parameters can be divided into the kinematic parameters $\Theta_{*\mathrm{K}}$ and the physical parameters $\Theta_{*\mathrm{P}}$. The kinematic parameters represent the geometrical structure and the physical parameters are related to the dynamics.

In this paper, we assume Θ_{OK} and Θ_{EK} are known. Even if those are unknown, the robot can generate the model online from vision data such as images and point clouds[1]. On the other hand, Θ_{OP} and Θ_{EP} are unknown because they are rarely known in general situation, and difficult to measure. $\Theta_{\mathrm{R}*}$ is treated as known.

2) Definision of Manipulation Strategy: There are various manipulation methods for heavy objects. Pushing[2], Pivoting[3], [4], Lifting[5], Holding[6], and Tumbling[7], are researched and achieved in experiments with real robots. We define the manipulation strategy S as this manipulation method, and express the strategy S with the object's pose sequence $\boldsymbol{r}(t)$ for general expression.

$$\mathrm{S} : \boldsymbol{r}(t) = (x(t), y(t), z(t), \alpha(t), \beta(t), \gamma(t))^{\mathrm{T}} \ (t \geq 0) \quad (1)$$

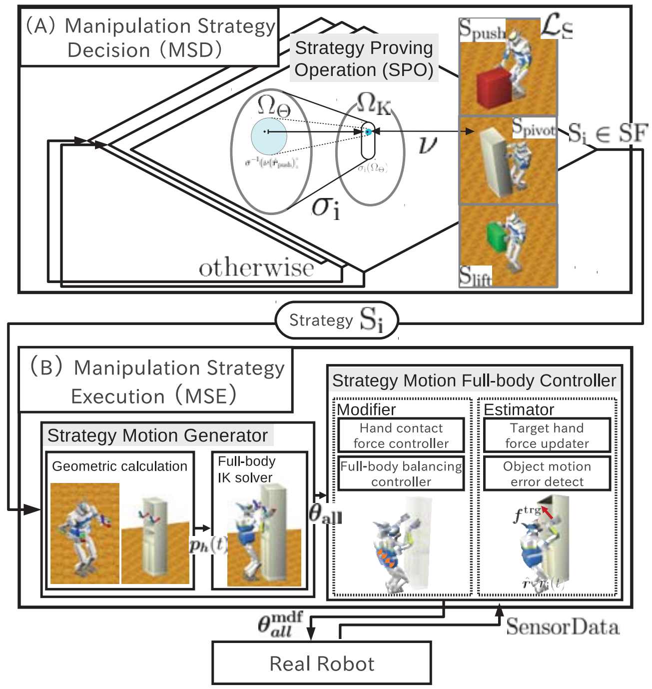


Fig. 2. Propsed System Configuration
(A) MSD decides the feasible strategy $\mathrm{S_i}$, and (B) MSE executes $\mathrm{S_i}$.

Object(O)		Environment(E)		Robot (R)	
Θ_{O}		Θ_{E}		Θ_{R}	
Kinematic Θ_{OK}	Physical Θ_{OP}	Kinematic Θ_{EK}	Physical Θ_{EP}	Kinematic Θ_{RK}	Physical Θ_{RP}
·Collision model ·Candidate of grasping points	·Mass ·COG ·Moment of inertia ·Coefficient of friction	·Collision model	·Coefficient of friction	·Collision model ·Joint model ·Range of motion	·Mass ·COG ·Moment of inertia ·Limit of torque

Fig. 3. Table of the O,E,R Parameters
Gray cells mean the unknown parameters.

The characters, x, y, z represent the position of COG in the world coordinate and α, β, γ represent the attitude in roll, pitch, yaw direction. For generating method of the manipulation strategy S, it is possible to define major general strategy based on heuristics, or generate automatically with learning algorithm by applying random sequence of external force and moment on dynamics simulation.

3) Problem: In the following, we solve the problem to decide and execute the manipulation strategy applicable for each object.

B. Our Proposed System

The Fig.2 shows the proposed system for carrying the objects with unknown physical parameter. The system consists of two components: (A) Manipulation Strategy Decision (MSD) and (B) Manipulation Strategy Execution (MSE). In (A), the robot decides the proper strategy based on the information got by strategy proving operation. Strategy proving operation are generated by considering mapping between physical parameter space and strategy list. In (B), the robot generate the full-body motion from strategy and execute it by full-body control. The generator and controller, which need no physical parameters, are common for various strategy. In the following sections, we introduce (A) in Sec.III, and (B) in Sec.IV. We perform the verification experiments to evaluate each component in the last of Sec.III and IV respectively, and the effectiveness of whole system is confirmed in the experiment with HRP-2 robot in Sec.V.

C. Related Works and Contributions of this Paper

1) The Framework for Strategy Decision: Although a lot of researches deal with the motion generation and control for individual strategy, the decision procedure of strategy is hardly researched. In the research of Nishide et al.[8], the robot predicts the dynamics by manipulating and learning, but the strategy decision is not discussed. The researches on the industrial arm robot are achieving the sophisticated object manipulations, but the working situation are mostly unchanged and the strategy switching is not necessary.

The framework of the autonomous decision becomes absolutely necessary for the robots to perform the carrying task in the general situation. In this research, we propose the general principle and the simple and heuristic detection methods. We believe the proposed system has the high comprehensiveness and this research becomes the base in this research field.

2) The Approach to deal with the physical parameters: Another feature of the proposed method is the point that we do not deal with the physical parameter directly. The model-based approach is possible by measuring some physical parameters such as mass, COG, and friction[9], [10], but there are following demerits: the system inevitably has the model error, the procedure becomes complicated, and the measured parameters are low-accuracy. We do not employ the learning-based approach[8], because learning for carrying heavy objects in the real world takes time and is sometimes dangerous. We believe the heuristic-based approach is more stable and powerful in practical use if the method has the enough basis and the generality.

III. MANIPULATION STRATEGY DECISION

A. General Theorem of the Method

In this subsection, we introduce the general principle justifying the Manipulation Strategy Decision (MSD).

1) The Relation between the Manipulation Strategy and the Parameter of Objects and Environment: We define the symbol of space and list as follows: the parameter space Ω_Θ, the object motion space Ω_{M}, and the strategy list $\mathcal{L}_{\mathrm{S}}$.

$$\Omega_\Theta = \{(\boldsymbol{\theta_O}, \boldsymbol{\theta_E}) \mid \ \boldsymbol{\theta_O} \in \Theta_{\mathrm{O}}, \boldsymbol{\theta_E} \in \Theta_{\mathrm{E}}\} \tag{2}$$

$$\Omega_{\mathrm{M}} = \{\boldsymbol{v} = (v_x, v_y, v_z, v_\alpha, v_\beta, v_\gamma)^{\mathrm{T}} \mid \ \boldsymbol{v} \in \mathbf{R}^6\} \tag{3}$$

$$\mathcal{L}_{\mathrm{S}} = \{\mathrm{S_1}, \mathrm{S_2}, \mathrm{S_3}, \cdots, \mathrm{S_i}, \cdots, \mathrm{S_n}\} \tag{4}$$

The brief operation can be considered as the mapping σ from Ω_Θ to Ω_{M}.

$$\sigma : \Omega_\Theta \longrightarrow \sigma(\Omega_\Theta) \subset \Omega_{\mathrm{M}} \ ; \quad (\boldsymbol{\theta_O}, \boldsymbol{\theta_E}) \longmapsto \boldsymbol{v} \tag{5}$$

The dimension and the range of $\sigma(\Omega_\Theta)$ are much smaller than those of Ω_Θ, and $\boldsymbol{v}$ is easy to measure compared to $(\boldsymbol{\theta_O}, \boldsymbol{\theta_E})$. It is also easy to classify $\sigma(\Omega_\Theta)$ by the moving direction . Fig.4 represents the example of the mapping σ_{i}.

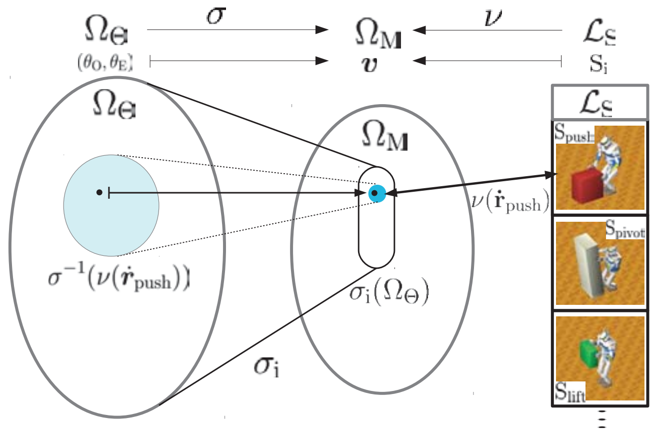


Fig. 4. The Relationship between Space $(\Omega_\Theta, \Omega_M)$, List $(\mathcal{L}_S)$, and Mapping (σ, ν)

If the object motion signified by $\boldsymbol{r_i(t)}$ can be achieved, the manipulation strategy S_i is feasible as the following formula.

$$\exists\sigma \text{ s.t. } \sigma(\boldsymbol{\theta_O}, \boldsymbol{\theta_E}) \in \nu(\dot{\boldsymbol{r}}_{\mathbf{i}}(\hat{t})) \implies S_i \in \text{FS} \quad (t = \hat{t}) \tag{6}$$

FS is the set of the feasible strategy for the object and the environment $(\boldsymbol{\theta_O}, \boldsymbol{\theta_E})$. ν is the neighborhood function independent of $(\boldsymbol{\theta_O}, \boldsymbol{\theta_E})$, which is defined heuristically. ν corresponds to the mapping between Ω_M and $\mathcal{L}_S$. The parameter set in which S_i is feasible can be expressed as $\sigma^{-1}(\nu(\dot{\boldsymbol{r}}_{\mathbf{i}}(\hat{t})))$ using the composite mapping (Fig.4).

2) The Definition of the Strategy Proving Operation: We define the Strategy Proving Operation (SPO) as follows.

$$\text{SPO}: \ \sigma \ \text{ s.t. } \ \widetilde{\boldsymbol{v}}_{\mathbf{i}} \in \sigma(\Omega_\Theta) \ \ (\widetilde{\boldsymbol{v}}_{\mathbf{i}} \in \nu(\dot{\boldsymbol{r}}_{\mathbf{i}}(\hat{t}))) \tag{7}$$

$\widetilde{\boldsymbol{v}}_{\mathbf{i}}$ is the key motion which is easy to prove. It is desirable that $\sigma^{-1}(\widetilde{\boldsymbol{v}}_{\mathbf{i}})$ approximates $\sigma^{-1}(\nu(\dot{\boldsymbol{r}}_{\mathbf{i}}(\hat{t})))$ because the sufficiency of the SPO result becomes higher. In this paper, we decide the strategy at the starting point, and let $\hat{t}$ be 0. SPO can deal with several key motions at one time in the following case.

$$\widetilde{\boldsymbol{v}}_{\mathbf{i}}, \widetilde{\boldsymbol{v}}_{\mathbf{j}}, \cdots \in \text{SPO}(\Omega_\Theta) \tag{8}$$

3) The Construction and Practice Method of the MSD using SPO: We introduce the method for constructing and practicing MSD. In the following, i) and ii) are the phase that constructing the system which are common for any objects and environment. In iii) and iv), we practice the strategy decision for each $(\boldsymbol{\theta_O}, \boldsymbol{\theta_E})$.

i) Generate $\mathcal{L}_S$.
ii) Define $\widetilde{\boldsymbol{v}}_{\mathbf{i}} (\in \nu(\dot{\boldsymbol{r}}_{\mathbf{i}}(\hat{t})))$ and SPO for each S_i.
iii) Execute SPO until finding the feasible strategy.
iv) If the following inequality is satisfied, S_i is feasible.

$$\|\text{SPO}(\boldsymbol{\theta_O}, \boldsymbol{\theta_E}) - \widetilde{\boldsymbol{v}}_{\mathbf{i}}\| < \varepsilon_{th} \tag{9}$$

B. Object Tilt / Slide Detection for the Selection of Pushing and Pivoting

In the following three subsections, we introduce the example of the implementation and the integration of general SPOs available by the dual-arm robot.

1) Statics Analysis of the Pushed Object: Pushing (σ_{push}) is one of the most basic operation. In the following, we analyze the statics of the pushed object on the floor. The object shape is assumed to be rectangular in this case, but many objects are approximated in this assumption. Setting variables as shown in left of Fig.5, equilibrium equations of forces and moments are as follows:

$$f_{hx} = f \tag{10}$$
$$f_{hz} + n = mg \tag{11}$$
$$cmg = lf_{hz} + hf_{hx} + pn \tag{12}$$

The object starts to move in the following two conditions: eq.(13) is the condition of tilt and eq.(14) is that of slide.

$$p \le 0 \iff f_{hx} \ge \frac{cmg - lf_{hz}}{h} \tag{13}$$
$$f > \mu n \iff f_{hx} > \mu(mg - f_{hz}) \tag{14}$$

Therefore, the movement of the pushed object $(\sigma_{push}(\Omega_\Theta))$ is classified into "tilt", "slide", and "still" (Fig.6 left).

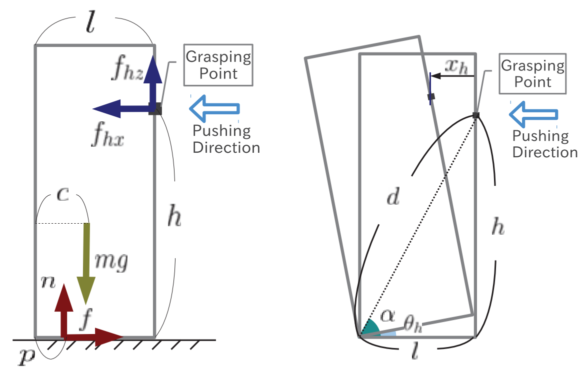


Fig. 5. Establishment of Variables
Left figure: Variables for statics analysis of a pushed object
Right figure: Variables for kinematic analysis of a tilted object

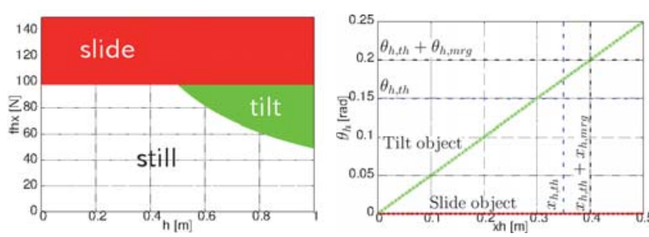


Fig. 6. The Movement of the Pushed Object
Left figure: The classification of the pushed object movement
Settings of the variables are as follows:
$f_{hz} = 0, l = 1.0, c = 0.5, m = 10, u = 1.0, g = 9.8$
Right figure: The threshold of the Object Tilt / Slide Detection
Setting of the variable is as follows: $h = 0.5$

2) Strategy and Key Motion: We set the strategy S_i and key motion $\widetilde{\boldsymbol{v}}_{\mathbf{i}}$ as Table.I . Eq.(15) is derived from the statics analysis. From eq.(15), the operation σ_{push} is SPO, which determines the feasibility of S_{push} and S_{pivot}. We call this SPO the Object Tilt / Slide Detection.

$$\widetilde{\boldsymbol{v}}_{\mathbf{push}}, \widetilde{\boldsymbol{v}}_{\mathbf{pivot}} \in \sigma_{push}(\Omega_\Theta) \tag{15}$$

3) Procedure of the Object Tilt / Slide Detection: The purpose of the Object Tilt / Slide Detection is the deter-

TABLE I

KEY MOTION FOR THE OBJECT TILT / SLIDE DETECTION

i (name)	push	pivot
$\mathrm{S_i}$	$\mathrm{S_{push}}$	$\mathrm{S_{pivot}}$
$\widetilde{\boldsymbol{v}}_{\mathbf{i}}$ (* : "don't care")	$(\dot{x}_{th}, *, 0, 0, 0, *)^{\mathrm{T}}$ $(\dot{x}_{th} > \varepsilon_{th})$	$(\dot{x}_{th}, *, 0, 0, \dot{\beta}_{th}, *)^{\mathrm{T}}$ $(\dot{x}_{th}, \dot{\beta}_{th} > \varepsilon_{th})$

TABLE II

KEY MOTION FOR THE OBJECT LEAVING FLOOR DETECTION

i (name)	lift
$\mathrm{S_i}$	$\mathrm{S_{lift}}$
$\widetilde{\boldsymbol{v}}_{\mathbf{i}}$ (* : "don't care")	$(*, *, \dot{z}_{th}, 0, 0, 0)^{\mathrm{T}}$ $(\dot{z}_{th} > \varepsilon_{th})$

The following is satisfied for each strategy: $\widetilde{\boldsymbol{v}}_{\mathbf{i}} \in \nu(\dot{\boldsymbol{r}}_{\mathbf{i}}(0))$
$\dot{y}$ and $\dot{\gamma}$ are $*$ in $\mathrm{S_{push}}$ and $\mathrm{S_{pivot}}$ because they do not affect the $\dot{x}$ and $\dot{\beta}$.
Also, $\dot{x}$ and $\dot{y}$ are $*$ in $\mathrm{S_{lift}}$.

mination that the object motion is "tilt" ($\widetilde{\boldsymbol{v}}_{\mathbf{pivot}}$) or "slide" ($\widetilde{\boldsymbol{v}}_{\mathbf{push}}$). Although there are a lots of determination methods, we propose the following procedure in this paper.

1) Apply the impedance control to the both hands.
2) Reach for the grasping point of the object.
3) Push the object forward quasi-statically.
4) Continue to push while $x_h < x_{h,th}$ and $\theta_h < \theta_{h,th}$.
5) $x_h > x_{h,th} \Rightarrow$ "slide"; $\theta_h > \theta_{h,th} \Rightarrow$ "tilt"; otherwise $\Rightarrow$ "still" or "error"

x_h and θ_h are x position and pitch angle of the hands. The string th in the index means threshold.

We assume that the hands keep from slipping because of grabbing and adaptation. By applying the impedance control[11], the motion of the hands becomes adaptive to the object position and posture (Fig.7). Therefore, we consider the position and posture of the hands approximate those of the object as eq.(16), and determine as 5) in the procedure.

$$\dot{x}_h \approx \dot{x}, \quad \dot{\theta}_h \approx \dot{\beta} \tag{16}$$

The visual recognition of the large object is difficult because the object is close to the robot and covers the field of view. Therefore, we used the kinematic information of the hands in this case.

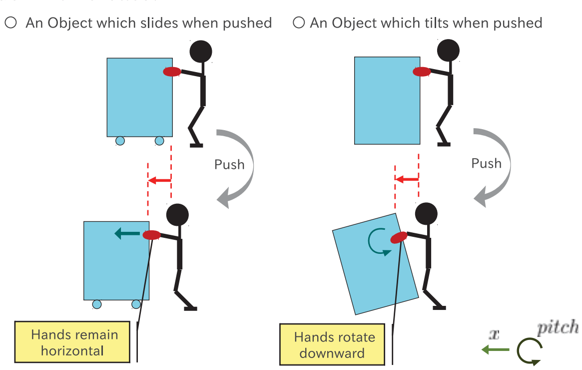


Fig. 7. Schematic View of Object Manipulation by Robot in Tilt / Slide Detection
When pushing a sliding object, hands remain horizontal (left figure), but when pushing a tilting object, hands rotate downward (right figure).

4) Calculation of Tilt / Slide Detection Threshold: The threshold $x_{h,th}, \theta_{h,th}$ should be changed depending on the height of the grasping point h because the assumed relation between x_h and θ_h changes during object tilting. As shown in right of Fig.5, d and α are defined by

$$d = \sqrt{l^2 + h^2}, \quad \alpha = \arctan \frac{h}{l} \tag{17}$$

Supposing that θ_h is small enough, x_h is given by

$$\begin{aligned} x_h &= d\cos\alpha - d\cos(\alpha + \theta_h) \\ &= d(\cos\alpha - \cos\alpha\cos\theta_h + \sin\alpha\sin\theta_h) \\ &= d\ \theta_h \sin\alpha \\ &= \theta_h\ h \end{aligned} \tag{18}$$

Using the constant margins $x_{h,mrg}(> 0)$ and $\theta_{h,mrg}(> 0)$, the threshold is defined as follows.

$$x_{h,th} = x_h - x_{h,mrg}, \quad \theta_{h,th} = \theta_h - \theta_{h,mrg} \tag{19}$$

When $\theta_{h,th}$ is set to be constant, $x_{h,th}$ is derived as follows from eq.(18) (Fig.6 right).

$$x_{h,th} = (\theta_{h,th} + \theta_{h,mrg})h - x_{h,mrg} \tag{20}$$

C. Object Leaving Floor Detection for the Selection of Lifting

In this subsection, we propose another SPO.

1) Statics Analysis of the Lifted Object: Lifting (σ_{lift}) is also basic operation. The object which is added the upward force f_z is lifted if the eq.(21) is satisfied. Otherwise the object keeps still.

$$f_z > mg \tag{21}$$

2) Strategy and Key Motion: We set the strategy $\mathrm{S_i}$ and key motion $\widetilde{\boldsymbol{v}}_{\mathbf{i}}$ as Table.II . From eq.(22), σ_{lift} is SPO for $\mathrm{S_{lift}}$. We call this SPO the Object Leaving Floor Detection.

$$\widetilde{\boldsymbol{v}}_{\mathbf{lift}} \in \sigma_{\mathrm{lift}}(\Omega_\Theta) \tag{22}$$

3) Procedure of the Object Leaving Floor Detection: The Object Leaving Floor Detection determines "lifted" ($\widetilde{\boldsymbol{v}}_{\mathbf{lift}}$) in the following procedure:

1) Reach for the grasping point of the object.
2) Add the upward force f_z quasi-statically until the saturation of the reaction force is detected or f_z exceeds the threshold $f_{z,th}$.
3) the saturation is detected $\Rightarrow$ "lifted"; otherwise $\Rightarrow$ "still" or "error"

$f_{z,th}$ is the threshold, which is the max force that the robot can exert. When the robot increases f_z, the reaction force that the hands receive saturates at the time when the object starts to leave the floor[12]. In the case that the saturation is not detected within $f_{z,th}$, the robot can not add the required force to lift the object. The saturation of the reaction force is detected by determining whether the difference of the reaction force which the humanoid robot measures becomes lower.

4) Calculation of Leaving Floor Detection Threshold: We calculate the hand contact force threshold from the torque limit as follows. Eq.(23) is the dynamics condition for the limbs of the robot.

$$(J^T \boldsymbol{f} - \boldsymbol{\tau_0})_i \leq (\boldsymbol{\tau_{max}})_i \quad \text{for all joints} \tag{23}$$

J, $\boldsymbol{\tau_0}$ and $\boldsymbol{\tau_{max}}$ are Jacobian of the arm, torque vector of arm joints during $\boldsymbol{f} = \boldsymbol{0}$, and maximum limit torque vector. $\boldsymbol{f}$ is the contact force of the hand. Let s and $\boldsymbol{r}$ be the norm and the unit direction vector of $\boldsymbol{f}$.

$$\boldsymbol{f} = s\boldsymbol{r} \quad (\|\boldsymbol{r}\| = 1) \tag{24}$$

From eq.(23), the maximum of the contact force is calculated as follows:

$$s \leq \min_i \left[\frac{(\boldsymbol{\tau_{max}} + \boldsymbol{\tau_0})_i}{(J^T \boldsymbol{r})_i} \right] \tag{25}$$

D. The Construction and the Experiment of MSD by Integrating SPO

In this subsection, we construct the system for Manipulation Strategy Decision by integrating Object Tilt / Slide Detection and Object Leaving Floor Detection. By the experiment in which a humanoid robot decides the strategy for carrying the various types of objects, we confirm the effectiveness of the constructed system.

1) The Priority Order of the Strategy: The priority order of the strategy can be considered by operability based on $\boldsymbol{r_i}$. In this system, we give the heuristic priority which is applicable to many situations based on the experience. In general, when the object is light, $\mathrm{S_{lift}}$ has higher operability than graspless manipulation[13] such as $\mathrm{S_{push}}$ and $\mathrm{S_{pivot}}$. If the object is heavy, $\mathrm{S_{push}}$ is more stable and faster than $\mathrm{S_{pivot}}$, and therefore $\mathrm{S_{push}}$ takes precedence over $\mathrm{S_{pivot}}$. Therefore, we decide the priority order as following: $\mathrm{S_{lift}}$, $\mathrm{S_{push}}$, $\mathrm{S_{pivot}}$.

2) The Planning of the Grasping Point when Switching the Strategy: The effective grasping point (GP) is different depending on the strategy. For example, from Fig.6, we can say that pushing becomes more stable as the GP becomes lower, and the required force for tilting becomes smaller as the GP becomes higher. When lifting, the GP close to the COG is more stable. We approximate the center of the shape as the COG because COG ($\in \Theta_{\mathrm{OP}}$) is unknown. In this system, Θ_{OK} includes the candidates of GPs, and the robot selects the most effective GP when the strategy is decided.

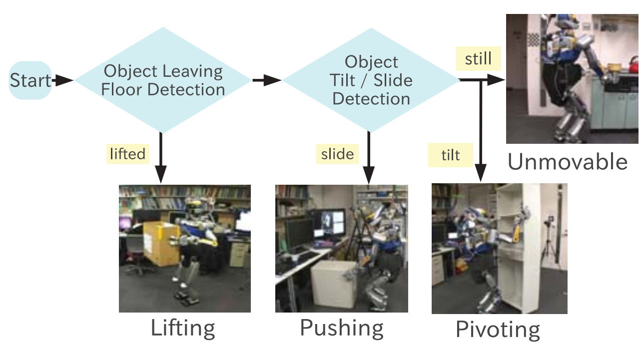


Fig. 8. System Flow of the Manipulation Strategy Decision
Determine the feasibility of lifting, pushing, and pivoting based on object leaving floor detection and object tilt / slide detection

3) The Construction and the Experiment of MSD: We constructed the system integrating Object Tilt / Slide Detection and Object Leaving Floor Detection, which decides the strategy from lifting, pushing, and pivoting (Fig.8).

We carried out the experiment in which the life-sized humanoid HRP-2 decides strategy for 10 objects by this system. We assigned the variables as follows: $x_{h,mrg} = 25$[mm],$\theta_{h,mrg} = 0.025$[rad], $\theta_{h,th} = 0.175$[rad], $f_{z,th} = 20$[N] and $x_{h,th}$ is calculated by eq.(20). Hand force is measured by 6-axis wrist force sensor, and the hand pose is calculated by the value of the IMU sensor attached to the root link and the encoder of each joint.

First, a small cardboard box and a garbage box were determined as "lifted" by Object Leaving Floor Detection (Fig.9). For a garbage box, the saturation of the reaction force was detected at the point that f_z was $5.18[N]$ for one hand, which is approximately equal with the value calculated from the object mass (Fig.10). The other 8 objects were determined as "still". Second, the robot determined 4 objects as "slide", and the other 4 objects as "tilt" by Object Tilt / Slide Detection (Fig.9). Fig.11 shows x_h and θ_h during the detection. This decision of strategy agrees with intuitive handling by human. We confirm the strategy is achieved by the execution system introduced in the next section through the experiment in Sec.V.

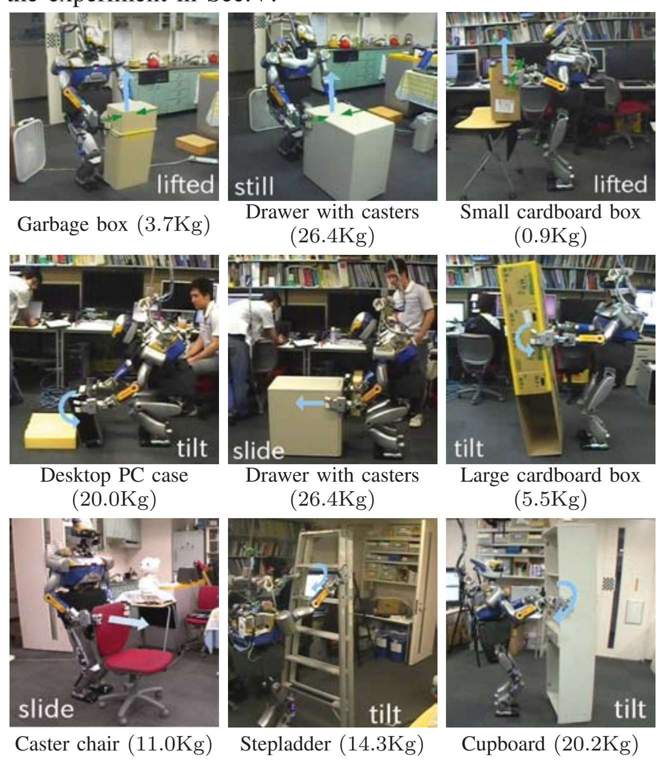
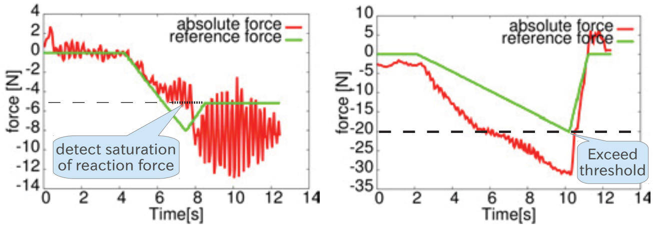


Fig. 9. Manipulation Strategy Decision by a Humanoid
The robot grabbed the object from the both sides and executed the detection.
Top 3 figures: Object Leaving Floor Detection.
Bottom 6 figures: Object Tilt / Slide Detection.

E. Applicable Scope of MSD

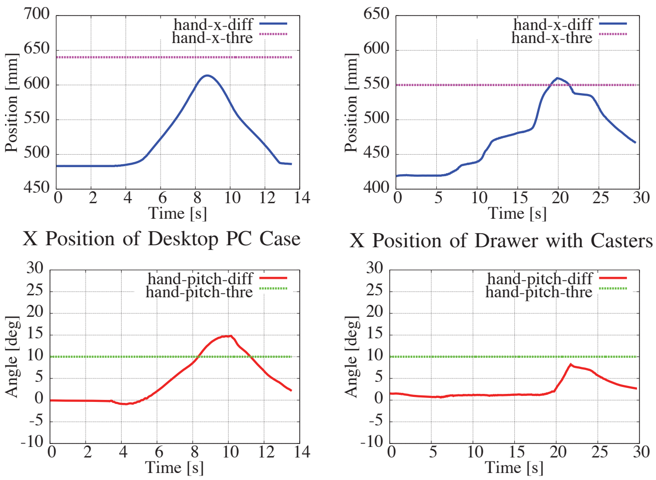
We clarify the scope of the proposed method for MSD. This section is mainly divided into two parts: the general theorem (III-A) and the implemented system (III-B,III-C,III-D).

1) The General Theorem: The general theorem is discussed on the basis of basic assumption of kinematics and



Garbage box Drawer

Fig. 10. Z-axis Hand Contact Force in Object Leaving Floor Detection
The reaction force saturates at $5N$ for the garbage box, and does not saturate within $20N$ for the drawer.



Pitch Angle of Desktop PC Case Pitch Angle of Drawer with Casters

Fig. 11. X Position (x_h) and Pitch Angle (θ_h) in Object Tilt / Slide Detection
Estimate desktop PC case as "tilt" because θ_h exceeds threshold and x_h does not exceeds. On the other hand, estimate drawer with casters as "slide" because x_h does and θ_h does not exceed threshold.

dynamics: (e.g. the motion of the operated object depends on the parameters of the object and the environment.) Therefore, the proposed concept is applicable in very wide scope (even outside of the earth). There are some heuristic variables and functions such as $\mathcal{L}_{\mathrm{S}}$, $\widetilde{\boldsymbol{v}}_{\mathbf{i}}$ and ν, and further researches based on the approach from the analytical or learning side is necessary.

2) The Implemented System: The implemented system is one that embodies the general theorem. The system can find the feasible strategy for the objects which satisfy eq.(26).

$$(\boldsymbol{\theta_O}, \boldsymbol{\theta_E}) \in \sigma^{-1}_{\mathrm{push}}(\widetilde{\boldsymbol{v}}_{\mathbf{push}}) \cup \sigma^{-1}_{\mathrm{push}}(\widetilde{\boldsymbol{v}}_{\mathbf{pivot}}) \cup \sigma^{-1}_{\mathrm{lift}}(\widetilde{\boldsymbol{v}}_{\mathbf{lift}}) \quad (26)$$

σ_{i} and $\widetilde{\boldsymbol{v}}_{\mathbf{i}}$ are properly selected so that the applicable range of objects, environment, and robots is wide. Although we draw the rectangular objects in the figures explaining the methods, the system can also deal with the objects made up of surfaces, edges and points (e.g. tables and stools). By considering the convex hull of the objects, we can approximate two point contact as line contact, and three or more point contact as surface contact. In many cases, asymmetry of the objects is not a problem because the decision does not care the motion in y and yaw direction (Table.I and Table.II).

There is not much specification for which a robot is required. The robot needs dual-arm and hands which can grasp the object from both side but the lower part is not limited: environment-fixed, wheeled, or legged type can execute this method. For legged robot, full-body balancing controller mentioned in Sec.IV-B can be employed. The kinematic and dynamic difference among robots is not a problem because the threshold $x_{h,th}, \theta_{h,th}$ and $f_{z,th}$ are automatically calculated from Θ_{R},

IV. MANIPULATION STRATEGY EXECUTION

A. The Generation of the Strategy Motion

As shown in Fig.2, the robot generates the motion depending on the decided strategy, and then sends the motion to the controller. The generator and controller we constructed need no Θ_{OP} and Θ_{EP}, and are common for various strategy. We define the motion generation as generating the sequence of the robot posture $\boldsymbol{\theta}_{\mathbf{all}}(t)$. As shown in Fig.12, (i) the geometric calculator converts $\boldsymbol{r}_{\mathbf{i}}(t)$ and the grasping point to the sequence of the hand's pose $\boldsymbol{p_h}(t)$, and then (ii) the full-body inverse kinematics solver converts $\boldsymbol{p_h}(t)$ to $\boldsymbol{\theta}_{\mathbf{all}}(t)$. The generation method uses only $\Theta_{*\mathrm{K}}$ and $\Theta_{\mathrm{R}*}$.

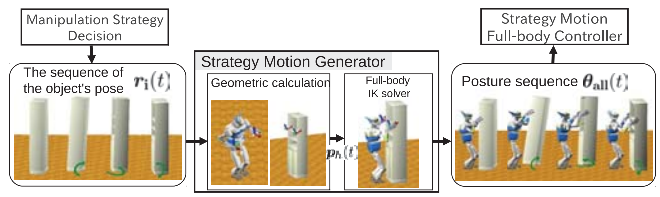


Fig. 12. System Flow of the Generation of the Strategy Motion
The figure shows an example in $\mathrm{S}_{\mathrm{pivot}}$.

B. The Full-body Control for the Strategy Motion

1) The Composition of the Control System: In this subsection, we introduce the composition of the control system for the two-legged robot, which is difficult to control because of necessity for balancing. For the robot of other types, we can use the part of the system.

Fig.13 left shows that the system is divided into modifier and estimator. The modifier modifies the target posture by the physical target value F^{trg} (e.g. target contact force). The estimator estimates F^{trg} based on sensor feedback. F^{trg} depends on Θ_{OP} and Θ_{EP}, but we do not estimate those parameters directly. In this way, the system can deal with unknown Θ_{OP} and Θ_{EP} implicitly.

2) The Modifier: Because two-legged robot falls down when the hands receive excessive force, the controller needs to treat the hand force explicitly. The modifier receives target hand contact force $\boldsymbol{f}^{\mathbf{trg}}$ as F^{trg} and modifies $\boldsymbol{\theta}_{\mathbf{all}}$. First, the robot controls hand contact force by feedback of measured hand reaction force $\boldsymbol{f}^{\mathbf{msr}}$, and exerts $\boldsymbol{f}^{\mathbf{trg}}$ on the object. The hand contact force controller employs the position-commanded impedance control[11] and modifies the reference hand pose $\boldsymbol{\xi}^{\mathbf{ref}}$ based on the error between $\boldsymbol{f}^{\mathbf{trg}}$ and $\boldsymbol{f}^{\mathbf{msr}}$ (eq.(27)). M_v, D_v and K_v in eq.(27) mean a virtual inertia coefficient matrix, a virtual viscosity coefficient matrix, and a virtual stiffness coefficient matrix respectively.

$$M_v \ddot{\boldsymbol{\xi}} + D_v(\dot{\boldsymbol{\xi}} - \dot{\boldsymbol{\xi}}^{\mathbf{ref}}) + K_v(\boldsymbol{\xi} - \boldsymbol{\xi}^{\mathbf{ref}}) = \boldsymbol{f}^{\mathbf{msr}} - \boldsymbol{f}^{\mathbf{trg}} \quad (27)$$

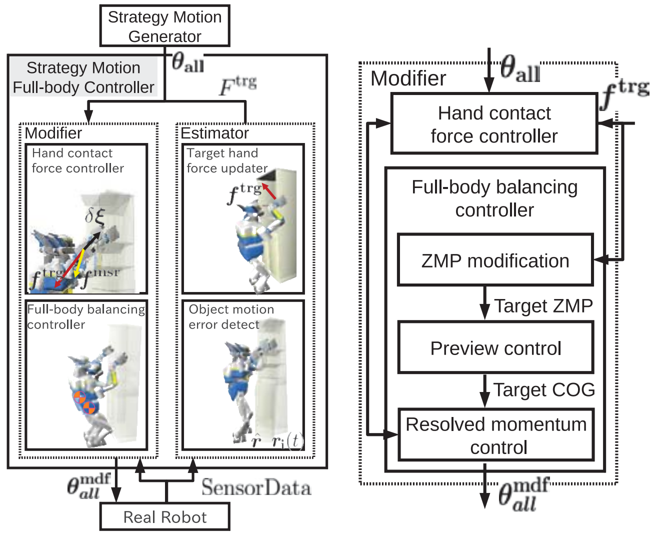


Fig. 13. System Flow of the Controller for the Strategy Motion
Left figure: The whole controller
Right figure: The detail of the modifier

The procedure of full-body balancing is shown in Fig.13 right. ZMP modification calculates the modified ZMP by eq.(28)[14].

$$x_{\text{zmp}} = \frac{Mx_g g - Mz_g \ddot{x}_g + \sum_{l/r}\{z_h f^{\text{trg}}_{h,x} - x_h f^{\text{trg}}_{h,z} + \tau^{\text{trg}}_{h,y}\}}{Mg - \sum_{l/r} f^{\text{trg}}_{h,z}} \quad (28)$$

Preview control[15] derives the target COG, and then resolved momentum control[16] calculates the modified posture $\boldsymbol{\theta}^{\text{mdf}}_{\boldsymbol{all}}$. The robot can exert $\boldsymbol{f}^{\text{trg}}$ while keeping balance by $\boldsymbol{\theta}^{\text{mdf}}_{\boldsymbol{all}}$.

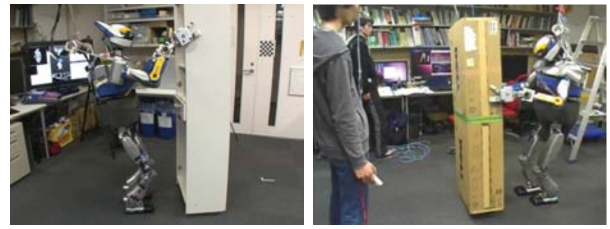
3) The Estimator: $\boldsymbol{f}^{\text{trg}}$ is estimated by sensor feedback in the estimator. We update $\boldsymbol{f}^{\text{trg}}$ periodically by eq.(29) to decrease the operational direction component of the force error.

$$\boldsymbol{f}^{\text{trg,j+1}} = \boldsymbol{f}^{\text{trg,j}} + K_u \left(\boldsymbol{f}^{\text{msr,j}} - \boldsymbol{f}^{\text{trg,j}}, \boldsymbol{r}^{\text{j}}\right) \boldsymbol{r}^{\text{j}} \quad (29)$$

K_u is a feedback gain, and $\boldsymbol{r}^{\boldsymbol{j}}$ is the unit vector representing the operational direction. The index, j, means the number of cycles. By this update, the proper $\boldsymbol{f}^{\text{trg}}$ is calculated automatically depending on the Θ_{OP} and Θ_{EP}, and follow-up performance of the object manipulation is improved.

The estimator also detects the error of the object motion. Based on the assumption that the hands keep from slipping, the estimated object motion $\hat{\boldsymbol{r}}$ can be calculated from the actual hand pose. By comparing $\hat{\boldsymbol{r}}$ with target motion $\hat{\boldsymbol{r}}_{\text{i}}(t)$, the robot can detect the motion error such as collision with obstacles. The separation of the hands from the object also can be detected from $\boldsymbol{f}^{\text{msr}}$.

4) The Verification Experiment: In order to confirm the effectiveness of the control system, we conducted the experiment in which the life-sized humanoid HRP-2 tilts the objects (Fig.14). Table.III is the result of the target and actual angle of tilting. The actual angle was measured by recognizing the checkerboard attached to the object using the vision sensor. The follow-up performance of tilting the heavy cupboard was improved by updating $\boldsymbol{f}^{\text{trg}}$. Fig.15 shows that the force error becomes small by update.

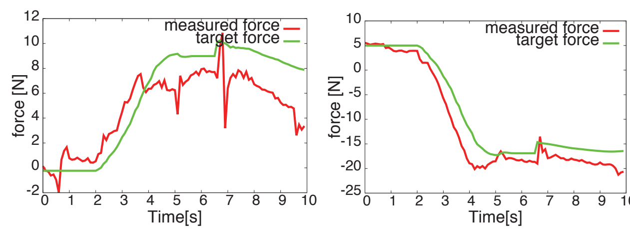


Cupboard　　Cardboard box

Fig. 14. Verification Experiment of Full-body Control System by Tilting the Object

TABLE III
OBJECT TILT ANGLE BY TILTING MANIPULATION

	Cupboard (20.2Kg)	Cardboard box (3.7Kg)
Target tilt angle [deg]	7.00	7.00
Actual tilt angle (without estimator) [deg]	1.71 (error 5.29)	6.29 (error 0.71)
Actual tilt angle (with estimator) [deg]	5.25 (error 1.75)	6.72 (error 0.28)



x-axis force of right hand　　y-axis force of right hand

Fig. 15. The Hand Contact Force during Tilting the Object

V. EXPERIMENT ON CARRYING FURNITURE

Using HRP-2[17], we performed the experiment on carrying the furniture, switching the manipulation strategy depending on the object autonomously by the integrated system of MSD and MSE. HRP-2 carried 10 objects, which are dealt with in the experiment of Sec.III-D. The objects have different physical parameters, which are unknown, but the robot achieved to estimate the required information directly for deciding and executing the strategy. Fig.16 shows that the robot carried the garbage box, the drawer with casters, and the cupboard by lifting, pushing, and pivoting respectively. Furthermore, we confirmed MSE enabled to execute the strategy motion robustly by the error detector and the force updater in the controller. (Fig.16(13)-(16)).

VI. CONCLUSIONS

In this paper, we proposed the method for deciding and executing the manipulation strategy of the objects with unknown physical parameters. Considering the operation as the mapping from the physical parameter space to the object motion space, we defined the strategy proving operation (SPO). By implementing SPO, we constructed the system deciding the feasibility of lifting, pushing, and pivoting. For executing the strategy, we constructed the generator and controller which various strategy can share, and enabled the robot to

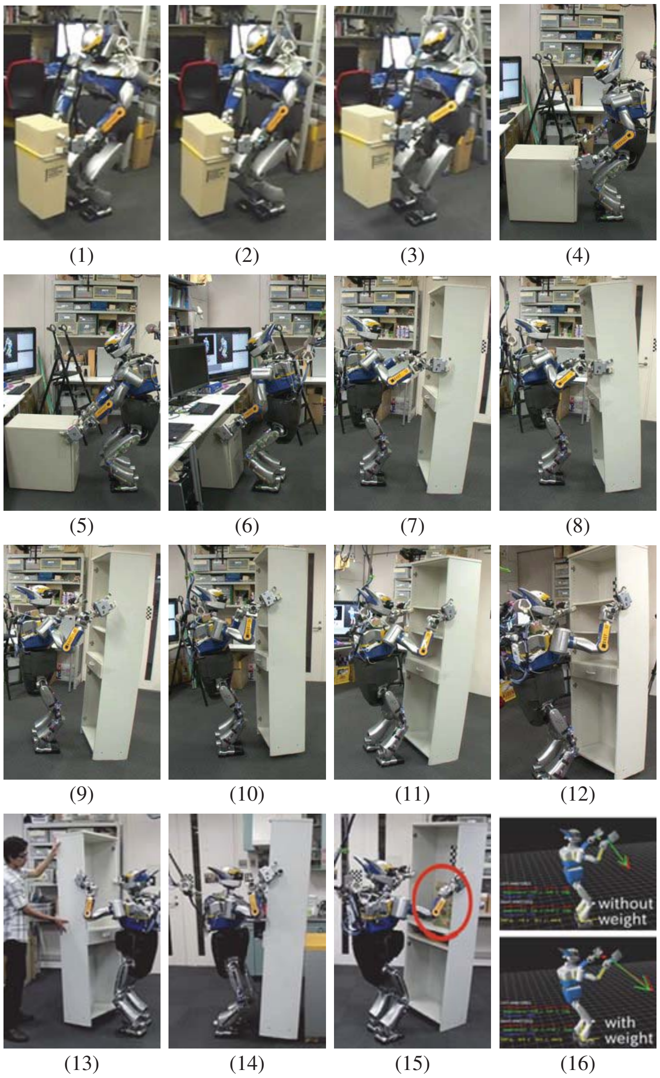


Fig. 16. Snapshots of the Experiment in which HRP-2 Carries Furniture. The robot decides the manipulation strategy autonomously based on the feasibility.
(1)-(3): Carry a garbage box (200mm x 310mm x 540mm, 3.7Kg) by lifting.
(4)-(6): Carry a drawer with caster (600mm x 390mm x 610mm, 26.4Kg, low friction) by pushing.
(7)-(12): Carry a cupboard (353mm x 586mm x 1816mm, 20.2Kg, high friction) by pivoting.
(13),(14): Detect the error of object motion by human interference or collision with obstacles.
(15),(16): Add a 8Kg weight (within the red circle) to the 20Kg cupboard. Target hand force is updated depending on the object weight automatically.

manipulate objects with unknown physical parameters. The feature of these system components is the point that required information is estimated directly, and the physical parameters are treated implicitly. We confirmed the effectiveness of each component and whole system through experiments. The life-sized humanoid robot decided and executed the strategy for carrying large furniture automatically. We can use this system in wide scope of objects, environments, and robots.

REFERENCES

[1] Yohei Kakiuchi, Ryohei Ueda, Kazuya Kobayashi, Kei Okada, Masayuki Inaba. Working with Movable Obstacles Using On-line Environment Perception Reconstruction Using Active Sensing and Color Range Sensor. In *Proceedings of the 2010 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 1969–1701, October 2010.

[2] Shunichi Nozawa, Yohei Kakiuchi, Kei Okada, and Masayuki Inaba. Controlling the planar motion of a heavy object by pushing with a humanoid robot using dual-arm force control. In *Proceedings of The 2012 IEEE International Conference on Robotics and Automation*, pp. 1428–1435, 5 2012.

[3] Eiichi Yoshida, Mathieu Poirier, Jean-Paul Laumond, Oussama Kanoun, Florent Lamiraux, Rachid Alami and Kazuhito Yokoi. Whole-body motion planning for pivoting based manipulation by humanoids. In *Proceedings of The 2008 IEEE International Conference on Robotics and Automation*, pp. 1712–1717, 2008.

[4] Eiichi Yoshida, Mathieu Poirier, Jean-Paul Laumond, Oussama Kanoun, Florent Lamiraux, Rachid Alami and Kazuhito Yokoi. Pivoting based manipulation by a humanoid robot. *Autonomous Robots*, Vol. 28, No. 1, pp. 77–88, 2010.

[5] Kensuke Harada, Shuuji Kajita, Hajime Saito, Mitsuharu Morisawa, Fumio Kanehiro, Kiyoshi Fujiwara, Kenji Kaneko, and Hirohisa Hirukawa. A Humanoid Robot Carrying a Heavy Object. In *Proceedings of The 2005 IEEE International Conference on Robotics and Automation*, pp. 1724 – 1729, April, 2005.

[6] Ohmura Yoshiyuki and Kuniyoshi Yasuo. Humanoid robot which can lift a 30kg box by whole body contact and tactile feedback. In *Proceedings of the 2007 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 1136–1141, October 2007.

[7] Naoyuki Sawasaki, Masayuki Inaba, and Hirochika Inoue. Tumbling Objects Using a Multi-Fingered Robot. In *Proceedings of the 20th International Symposium on Industrial Robots and Robot Exhibition*, 1989.

[8] Shun Nishide, Tetsuya Ogata, Jun Tani, Kazunori Komatani, and Hiroshi G. Okuno. Predicting object dynamics from visual images through active sensing experiences. *Advanced Robotics*, Vol. 22, No. 5, pp. 527–546, 2008.

[9] Yong Yu, Kenro Fukuda, and Showzow Tsujio. Estimation of mass and center of gravity of graspless unknown object based on gravity equi-effect planes. In *Proceedings of the 1998 IEEE/RSJ International Conference on Intelligent Robots and Systems*, Vol. 3, pp. 1497–1502, October 1998.

[10] Yoshikawa Tsuneo, and Kurisu Masamitsu . Indentification of the center of friction from pushing an object by a mobile robot . In *Proceedings of the 1991 IEEE/RSJ International Conference on Intelligent Robots and Systems*, November 1991.

[11] Shunichi Nozawa, Iori Kumagai, Yohei Kakiuchi, Kei Okada, and Masayuki Inaba. Humanoid full-body controller adapting constraints in structured objects through updating task-level reference force. In *Proceedings of the 2012 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 3417–3424, 10 2012.

[12] Shunichi Nozawa, Ryohei Ueda, Yohei Kakiuchi, Kei Okada, and Masayuki Inaba. Sensor-based integration of full-body object manipulation based on strategy selection in a life-sized humanoid robot. *Journal of Robotics and Mechatronics*, Vol. 23, No. 2, pp. 239–248, (2011).

[13] Y. Aiyama, M. Inaba, and H. Inoue. Pivoting: A new method of graspless manipulation of object by robot fingers. In *Proceedings of the 1993 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 136–143, 1993.

[14] K. Harada, S. Kajita, K.Kaneko, and H.Hirukawa. Zmp analysis for arm/leg coordination. In *Proceedings of the 2003 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 75–81, (2003).

[15] Shuuji Kajita, Fumio Kanehiro, Kenji Kaneko, Kiyoshi Fujiwara, Kensuke Harada, Kazuhito Yokoi, and Hirohisa Hirukawa. Biped walking pattern generation by using preview control of zero-moment point. In *Proceedings of The 2003 IEEE International Conference on Robotics and Automation*, pp. 1620–1626, Sep (2003).

[16] S.Kajita, F.Kanehiro, K.Kaneko, K.Fujiwara, K.Harada, K.Yokoi, and andH.Hirukawa. Resolved Momentum Control:Humanoid Motion Planning based on the Linear and Angular Momentum. In *Proceedings of the 2003 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 1644–1650, October, 2003.

[17] Kei Okada, Takashi Ogura, Atsushi Haneda, Junya Fujimoto, Fabien Gravot, and Masayuki Inaba:. Humanoid Motion Generation System on HRP2-JSK for Daily Life Environment. In *International Conference on Mechatronics and Automation*, pp. 1772 – 1777, July, (2005).