Optimization of Impact Motions for Humanoid Robots
Considering Multibody Dynamics and Stability

Teppei Tsujita, Atsushi Konno and Masaru Uchiyama

Abstract—In order to exert large force on an environment, it is effective to apply impulsive force. We describe the motions that perform tasks by applying impulsive force as “impact motion.” The objective of an impact motion is to exert large force on an environment, however if the impulsive force is too large, the robot may fall down due to the reaction force. This paper presents an optimization scheme to generate impact motions for humanoid robots. The advantage of the proposed scheme is that impulsive force exerted on a target by a humanoid robot’s whole body is maximized while guaranteeing the stability. A punching motion is generated by the scheme as an example and evaluated by performing simulations.

I. INTRODUCTION

When a robot applies force statically on an environment, the magnitude of the force is limited by the maximum torque of its actuators. In order to exert large force on the environment beyond this limitation, it is effective to apply impulsive force. For example, if a robot pushes a nail statically by a hammer, the nail cannot be inserted into a wood. However, the nail can be driven by hitting by the hammer since momentum of a hammer and a robot is exerted in a short time. We describe the motions that perform tasks by applying impulsive force as “impact motion.” There are difficult problems introduced by impacts between robots and environments.

Uchiyama proposed a control algorithm constitution method and dynamic control modes for performing a nailing task by a 3 DOF manipulator [1]. Asada and Ogawa proposed the virtual mass for analyzing dynamic behavior of a manipulator arm and its end effector that interacts with the environment [2]. Around the same time, Khatib and Burdick proposed the effective mass [3]. These works mentioned above used robotic manipulators fixed on the ground. Effective mass is maximized while guaranteeing the stability. In order to avoid falling during the impact motion, relationship between ZMP(Zero-Moment Point) [7] and a support polygon is evaluated in all phase in the optimization process.

II. OUTLINE OF MOTION GENERATION SCHEME

An impact motion is divided into three phases as follows in this paper.

Acceleration phase: A robot accelerates its joints to hit a target object during this phase.
Impact phase: During this phase, the robot hits the target. This phase is short time just before and after contact.
Slowdown phase: The robot decelerates its joints after hitting the target and stands still during this phase.

Fig. 1 shows an outline of the proposed motion generating method. This method consists of two parts.

i) Optimizing an impact posture and joint velocities at the time of impact.
ii) Generating motions for acceleration and slowdown phases.

In the first stage, the posture and joint velocities are decided under constraints to maximize the impulsive force or impulse exerted on a target. By using the simplified
dynamics computation (SDC) model [6], the impulsive force and impulse can be estimated. Moreover, behavior of ZMP during the impact phase can be obtained. However, it is not known how stable the robot is during the acceleration and slowdown phases at this stage. Hence, there is a possibility of generating motion which is unstable during the acceleration and slowdown phase even if the impact posture and velocities are stable during the impact phase. Therefore, the following steps are considered to overcome the problem.

1) Setting decision variables (posture, end effector’s velocity, momentum, angular momentum and velocity of CoM (Center of Mass)) at the time of impact and estimating impulsive force and ZMP during impact phase by the SDC model.

2) Interpolating the decision variables from zero and to zero and computing ZMP during acceleration and slowdown phases.

3) If the impulsive force is maximum under constraints, e.g., stability, the motion generation process proceeds to Step 4). If the force is not maximum, the decision variables are slightly changed and motion generation process returns to Step 1).

4) Generating motions for acceleration/slowdown phases to satisfy the interpolated trajectories in Step 2).

Since Step 1) and 2) are executed in optimization process, these steps are computed numerous times. The advantage of this scheme is that ZMP trajectory during acceleration and slowdown phases can be estimated in process 2) without computing the inverse dynamics.

ZMP can be expressed in the form.

\[
P_{ZMP_x} = \frac{M_g P_{Gx} - \dot{L}_y}{M_g + P_z}, \quad P_{ZMP_y} = \frac{M_g P_{Gy} + \dot{L}_x}{M_g + P_z},
\]

where \(P_{ZMP_x}, P_{ZMP_y}, P_{Gx} \) and \(P_{Gy} \) are position of ZMP in \(X_x \) and \(Y_y \) directions and GCoM (Ground projection of Center of Mass) in \(X_b \) and \(Y_b \) directions, respectively. \(\dot{L}_x, \dot{L}_y \) and \(P_z \) are differentiations of angular momentums around \(X_x \) and \(Y_y \) axes and momentum in \(Z_b \) direction, respectively. \(M \) and \(g \) are respectively total mass of the system and acceleration of gravity. This formula indicates that a time series ZMP trajectory can be computed using the CoM, angular momentum and momentum trajectories. Hence, by interpolating these values, ZMP trajectory can be obtained with low computation cost. Therefore, stability during the acceleration and slowdown phase can be evaluated in the optimizing process. By considering stability based on ZMP and a support polygon as a constraint condition, the optimizing problem outputs a stable motion in all phases.

III. IMPACT MOTION OPTIMIZATION

In this section, detail of the proposed impact motion optimization scheme is presented. Notations of the decision variables in Step 1) are denoted in Subsection III-A and III-B and a motion generation scheme in Step 4) is presented in Subsection III-C. Formulation of the optimization problem in Step 1) \(\sim 3)\) is presented in Subsection III-D. Superscripts \(rl, ll, ra\) and \(la\) express the robot’s right leg, left leg, right arm and left arm numbers, respectively and the total degree of freedom of the robot is \(n\).

A. Posture (Joint Angles)

Let’s consider a situation when the humanoid robot hits a free floating target box in \(X_0 \) direction as shown in Fig. 2. In the figure, \(\Sigma_0\) is the world coordinate system, \(\Sigma_0\) is the robot body coordinate system, \(\beta_{fr_0}\) is the position vector of the right foot with respect to \(\Sigma_0\), \(\beta_{sl_i}\) is the position vector of the left foot with respect to the right foot coordinate system and \(\beta_{fr_i}\) is the \(Z-Y-X\) Euler angle vector of the right foot with respect to \(\Sigma_0\). Left superscripts 0 and \(fr\) denote \(\Sigma_0\) and the right foot coordinate system, respectively. Elements of \(\beta_{fr}, \beta_{sl_i}\) and \(\beta_{fr_i}\) are denoted as follows.

\[
\begin{align*}
\beta_{fr} &= [\beta_{fr_x}, \beta_{fr_y}, \beta_{fr_z}]^T, \\
\beta_{sl_i} &= [\beta_{sl_i,x}, \beta_{sl_i,y}, \beta_{sl_i,z}]^T, \\
\beta_{fr_i} &= [\beta_{fr_i,x}, \beta_{fr_i,y}, \beta_{fr_i,z}]^T,
\end{align*}
\]

where \(\beta_{fr}, \beta_{sl_i}\) and \(\beta_{fr_i}\) are the position vector of the right foot in \(X_0\) direction with respect to \(\Sigma_0\) and the others are defined similarly.

The position and orientation of \(\Sigma_0\) with respect to \(\Sigma_0\) is expressed as follows on the assumption that the right foot is fixed on the ground.

\[
p_0 = -\beta_{fr}^T \beta_{fr}, \quad R_0 = \beta_{fr}^T
\]

where \(p_0\) and \(R_0\) are the position vector and the rotation matrix of \(\Sigma_0\) with respect to \(\Sigma_0\). \(\beta_{fr}^T \beta_{fr}\) expresses orientation of the right foot with respect to \(\Sigma_0\). In this paper, position vectors and rotation matrices which does not have a left superscript are expressed with respect to \(\Sigma_0\).

By this operation, the origin of the right foot is same with the origin of \(\Sigma_0\). The joint angles of the right leg \(\phi^{rl} = [\phi_{rl}^1, \ldots, \phi_{rl}^i, \ldots, \phi_{rl}^n]_T\) can be obtained by inverse kinematics using \(\beta_{fr}\) and \(\beta_{fr_i}\). The subscript \(i\) expresses the joint number which is numbered from a root link to a tip link and \(n_{rl}\) expresses maximum number of the right leg joints. The position of the left foot \(\beta_{fl}\) can be calculated in the form:

\[
\beta_{fl} = \beta_{fr} + \beta_{fl} \beta_{sl_i}
\]

The joint angles of the left leg \(\phi^{ll} = [\phi_{ll}^1, \ldots, \phi_{ll}^n]_T\) can be obtained on the assumption that the orientation of the left foot is the same with the right foot. \(n_{ll}\) expresses...
maximum number of the left leg joints. Therefore, the both legs’ postures are defined by \( p_{fr}, \alpha_{fr} \) and \( f_{sfl} \) under the kinematic closure constraint. In addition, by defining the joint angles except for the leg joints, the posture can be defined at the impact. The remaining joint angle vector is expressed as \( \phi^{em} \).

B. Joint Velocities

The joint angle velocities are computed from the following values.
- Velocity of the end effector
- Momentum and angular momentum
- Velocity of the center of mass

The relationship between the joint angle velocities and the above mentioned values are expressed in the form:

\[
\begin{bmatrix}
    v_c \\
    \omega_c \\
    P \\
    L \\
    v_g
\end{bmatrix} =
\begin{bmatrix}
    J_c(\phi) \\
    J_{PL}(\phi) \\
    J_{TL}(\phi)
\end{bmatrix} \dot{\phi}, \quad (4)
\]

where

\[
\dot{\phi} \equiv \begin{bmatrix}
    \dot{\phi}^{vl} \\
    \dot{\phi}^{vl} \\
    \dot{\phi}^{em}
\end{bmatrix},
\quad J_{PL} \equiv \begin{bmatrix}
    J_{P_x} & J_{P_y} & J_{P_z}
\end{bmatrix},
\quad J_{TL} \equiv \begin{bmatrix}
    J_{T_{gx}} \\
    J_{T_{gy}} \\
    J_{T_{gz}}
\end{bmatrix},
\]

\( \dot{\phi} \in R^{n \times 1} \) : joint velocity vector,
\( v_c \in R^{3 \times 1} \) : velocity of the collision point,
\( \omega_c \in R^{3 \times 1} \) : angular velocity of the collision point,
\( P \in R^{3 \times 1} \) : momentum of the whole system in \( \Sigma_b \),
\( L \in R^{3 \times 1} \) : angular momentum of the whole system in \( \Sigma_b \),
\( v_g \in R^{3 \times 1} \) : velocity vector of the center of mass,
\( \omega_g \in R^{3 \times 1} \) : angular velocity of the center of mass.

The velocity of CoM is dependent to the momentum of the whole system. The velocity can be obtained by dividing the momentum by the total mass. Therefore, the rows of \( P \) and \( v_g \) are not independent. From (1), only the velocity of GCoM, momentum in \( Z_b \) direction and angular momentum around \( X_b \) and \( Y_b \) affect the ZMP. Therefore, the dimensions of (4) can be reduced as follows:

\[
\begin{bmatrix}
    v_c \\
    \omega_c \\
    P_z \\
    L_x \\
    L_y \\
    v_gx \\
    v_gy
\end{bmatrix} =
\begin{bmatrix}
    J_c \\
    J_{P_z} \\
    J_{L_x} \\
    J_{L_y} \\
    J_{T_{gx}} \\
    J_{T_{gy}}
\end{bmatrix} \dot{\phi} \equiv \mathbf{J}_{hu} \dot{\phi}, \quad (6)
\]

In addition, the constraint conditions must be considered when there are kinematic closures.

\[
\begin{bmatrix}
    v_c \\
    \omega_c \\
    \mathbf{M} \\
    \mathbf{v}_{T_{gy}}
\end{bmatrix} =
\begin{bmatrix}
    J_c \\
    J_{P_z} \\
    P_z \\
    L_y \\
    v_gx \\
    v_gy
\end{bmatrix} \dot{\phi} \equiv \mathbf{J}_{hu} \dot{\phi}, \quad (7)
\]

where

\[
\mathbf{V} \equiv \begin{bmatrix}
    v_c \\
    \omega_c \\
    \mathbf{M} \\
    \mathbf{v}_{T_{gy}}
\end{bmatrix},
\quad \mathbf{M} \equiv \begin{bmatrix}
    P_z \\
    L_y \\
    v_gx \\
    v_gy
\end{bmatrix},
\]

and \( C_\alpha \in R^{n \times n} \) is the constraint formula and the size of column depends on the situation. For example, the kinematics closure of the legs is expressed by this constraint formula as follows to avoid generating internal forces in the situation where a humanoid robot stands with both feet as shown in Fig. 2. In this case, the relative velocities which are generated by both legs must be zero. Therefore, this constraint condition can be formulated in the form:

\[
0 = \begin{bmatrix}
    \mathbf{J}_0 & -\mathbf{J}_0' & 0 & \cdots & 0
\end{bmatrix} \dot{\phi} \equiv \mathbf{C}_\alpha \dot{\phi}, \quad (8)
\]

where \( \mathbf{J}_0 \) expresses relationship between the velocities of \( \Sigma_0 \) generated by the right leg and the right leg joints and \( \mathbf{J}_0' \) expresses relationship between the velocities of \( \Sigma_0 \) generated by left leg and the left leg joints.

\[
\mathbf{J}_0 \equiv \begin{bmatrix}
    -\mathbf{E} & 0 & \mathbf{p}_{fr}^T \\
    0 & -\mathbf{E} & -\mathbf{E}
\end{bmatrix},
\quad \mathbf{J}_0' \equiv \begin{bmatrix}
    -\mathbf{E} & 0 & \mathbf{p}_{fl}^T \\
    0 & -\mathbf{E} & -\mathbf{E}
\end{bmatrix},
\]

\( \mathbf{0} \) express relationship between the joints and the velocities of the right foot with respect to \( \Sigma_0 \) and \( \mathbf{0} \) express relationship between the joints and the velocities of the left foot with respect to \( \Sigma_0 \). And \( \mathbf{E} \in R^{3 \times 3} \) denotes an identity matrix. \( \mathbf{p} \) is the skew-symmetric matrix to rewrite the cross product by matrix multiplication as \( \mathbf{p} \times \mathbf{b} = \mathbf{p}^T \mathbf{b} \) where \( \mathbf{b} \) is an arbitrary vector.

The simultaneous equation (7) describes the relationships between the joint angle velocities and the end effector velocities, the CoM velocity and the momentums under the
Discontinuous change caused by impact
Returning to the reference by the servo controller... and the others are defined similarly. The position of the target box relative to the robot is illustrated in Fig. 4.

Based on a result of optimization problem described in the next subsection. The assumed actual behavior of the interpolated trajectories is shown in Fig. 3. The discontinuous change is occurred at the time of impact by impulsive force. However, the actual trajectory returns to the reference one immediately by the servo controller.

### D. Formulation of the Optimization Problem

By using the simplified contact dynamics computation model [6], peak force or impulse exerted on a target object can be obtained. Therefore, the object of this optimization problem is to maximize the peak force or impulse by varying robot’s posture and angle velocities. As stated in Subsection III-A and III-B, the posture and angle velocities can be obtained by giving the following values:

- The position vector of the right foot with respect to $\Sigma_0$.
- The right foot’s Z-Y-X Euler angle with respect to $\Sigma_0$.
- The position vector of the left foot with respect to the right foot coordinate system.
- The joint angles except for the leg joints.
- The collision velocity $v_c$ in $X_b$ direction.
- The momentum in $Z_b$ direction, the angular momentum around $X_b$ and $Y_b$ axes.
- The velocity of the center of mass.

Therefore, the decision variables are expressed as follows:

$$\mathbf{x} = \begin{bmatrix} \mathbf{p}_{fr}^T & v_{fr}^T & s_{fr}^T & \phi_{\text{rem}}^T & v_c & \mathbf{M}^T & v_{T_{grv}}^T \end{bmatrix}^T,$$

and the ranges of the variables are denoted as follows:

$$\begin{align}
\min \phi_{\eta,fr} & \leq \phi_{\eta,fr} \leq \max \phi_{\eta,fr} \quad (\eta = x, y, z), \\
\min \alpha_{\eta,fr} & \leq \alpha_{\eta,fr} \leq \max \alpha_{\eta,fr} \quad (\eta = x, y, z), \\
\min \phi_{\eta,fr} & \leq \phi_{\eta,fr} \leq \max \phi_{\eta,fr} \quad (\eta = x, y, z), \\
\min \mathbf{r} & \leq \mathbf{r} \leq \max \mathbf{r}, \\
\min \mathbf{P} & \leq \mathbf{P} \leq \max \mathbf{P}, \\
\min \mathbf{L} & \leq \mathbf{L} \leq \max \mathbf{L}, \\
\min \mathbf{v} & \leq \mathbf{v} \leq \max \mathbf{v},
\end{align}$$
The objective function for impact motions is obviously the peak force $f_{\text{max}}$ or impulse $f_n$. The objective value is decided based on a task whether the force or impulse. This optimization problem is expressed as follows and this optimizing problem can be solved by SQP (Sequential Quadratic Programming).

$$\begin{align*}
\text{minimize} & \quad f(x) = -f_{\text{max}} \quad \text{or} \quad -f_n, \\
\text{subject to} & \quad \begin{cases}
(z_{ta} - z_{te}) \leq n_{ce}^T R_{ce} n_{h} \leq (z_{ta} + z_{te}), \\
\min R_{ce} \leq n_{ce}^T R_{ce} n_{h} \leq \max R_{ce}, \\
-v_{n,ce} \leq v_{n,ce} \leq v_{n,ce}, \\
-\omega_{n,ce} \leq \omega_{n,ce} \leq \omega_{n,ce}, \\
-z_{fe} \leq n_{fe}^T R_{fe} n_{fe} \leq z_{fe}, \\
\min R_{fe} \leq n_{fe}^T R_{fe} n_{fe} \leq \max R_{fe}, \\
||p_f|| \leq \eta_{fe}, \\
||\omega_f|| \leq \omega_{fe}, \\
l_{z,\text{imp}} \geq l_{z,\text{imp}}, \\
l_{z,ax} \geq l_{z,ax}, \\
(\phi_{k,\text{lim}} + C_6) \leq \phi_k \leq (\phi_{k,\text{ulim}} - C_6), \quad (k = r, l, t) \\
-\phi_k \leq \phi_{k,\text{lim}} \leq \phi_k \leq \phi_{k,\text{ulim}}, \\
\min P_k \leq P_k \leq \max P_k, \quad (\eta = x, y), \\
\min L_k \leq L_k \leq \max L_k.
\end{cases}
\end{align*}$$

In (14), the kinematics and ZMP are computed on the assumption that the right foot is fixed on the ground. The part 1 expresses constraints depended on the task. In this case (Fig. 2), this part defines behavior of the hand and the box at the time of impact. Fig. 5 explains allowed range of the colliding direction. In the figure, $\eta_n$ is an unit vector pointing the collision direction in the local coordinate of the end effector. In this case, since the robot punches the box to negative direction in $Z$ direction with respect to the local coordinate of the end effector, $\eta_n$ is $n = [0 \quad 0 \quad -1]^T$. The part 2 expresses constraints for the both legs and avoids generating internal forces. Through the kinematic closure is defined by (8), the solution (12) using pseudoinverse does not guarantee the constraints. Therefore, the constraint is defined here. The parts 3 and 4 express stability during impact phase and acceleration/slowdown phases, respectively. These parts keep ZMP inside of a support polygon. $l_{z,imp}$ is computed by SDC model and $l_{z,ax}$ is computed by the ZMP estimation scheme for acceleration/slowdown phases stated in Section II CoM, angular momentum and momentum at the time of impact are interpolated by third-order polynomial as shown in Fig.3. By using the interpolated trajectories, ZMP during acceleration/slowdown phases are computed by (1) without computing the inverse dynamics and distance between support polygon and ZMP is evaluated. Therefore, the part 5 guarantees the stability during acceleration/slowdown phases in the optimization process.

The part 6 defines hardware limitation of the robot’ legs. The part 7 expresses limitation of momentums in $X_b$ and $Y_b$ directions and angular momentum around $Z_b$ axis. These are related to friction between a floor and the feet.

IV. SIMULATION RESULT

In order to evaluate the proposed scheme, a punching motion is generated to perform by the humanoid robot HRP-
TABLE I
DECISION VARIABLES $p_{scr}, \alpha_{scr}, f_{scr}, v_c, M$ and $v_{T_{xy}}$

<table>
<thead>
<tr>
<th>init.</th>
<th>min.</th>
<th>max.</th>
<th>result</th>
<th>init.</th>
<th>min.</th>
<th>max.</th>
<th>result</th>
</tr>
</thead>
<tbody>
<tr>
<td>$p_{x,scr}$ (m)</td>
<td>$p_{y,scr}$ (m)</td>
<td>$p_{x,scr}$ (m)</td>
<td>$p_{y,scr}$ (m)</td>
<td>$p_{x,scr}$ (m)</td>
<td>$p_{y,scr}$ (m)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>$\phi_{x,sc}$ (°)</td>
<td>$\phi_{y,sc}$ (°)</td>
<td>$\phi_{x,sc}$ (°)</td>
<td>$\phi_{y,sc}$ (°)</td>
<td>$\phi_{x,sc}$ (°)</td>
<td>$\phi_{y,sc}$ (°)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>$f_{scr}$ (m/s)</td>
<td>$v_c$ (m/s)</td>
<td>$v_{T_{xy}}$ (Nms)</td>
<td>$P_a$ (Ns)</td>
<td>$L_x$ (Nms)</td>
<td>$L_y$ (Nms)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>$v_{x,y}$ (m/s)</td>
<td>$v_{y,z}$ (m/s)</td>
<td>$v_{z,x}$ (m/s)</td>
<td>$v_{x,y}$ (m/s)</td>
<td>$v_{y,z}$ (m/s)</td>
<td>$v_{z,x}$ (m/s)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>$v_{x,y}$ (m/s)</td>
<td>$v_{y,z}$ (m/s)</td>
<td>$v_{z,x}$ (m/s)</td>
<td>$v_{x,y}$ (m/s)</td>
<td>$v_{y,z}$ (m/s)</td>
<td>$v_{z,x}$ (m/s)</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

TABLE II
DECISION VARIABLE $\phi^{cen}$

<table>
<thead>
<tr>
<th>init. (°)</th>
<th>$\phi_{x,a}^{cen}$</th>
<th>$\phi_{y,a}^{cen}$</th>
<th>$\phi_{z,a}^{cen}$</th>
<th>$\phi_{a}^{cen}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\phi_{x,a}^{cen}$</td>
<td>$\phi_{y,a}^{cen}$</td>
<td>$\phi_{z,a}^{cen}$</td>
<td>$\phi_{a}^{cen}$</td>
<td></td>
</tr>
<tr>
<td>$\phi_{x,a}^{cen}$</td>
<td>$\phi_{y,a}^{cen}$</td>
<td>$\phi_{z,a}^{cen}$</td>
<td>$\phi_{a}^{cen}$</td>
<td></td>
</tr>
<tr>
<td>$\phi_{x,a}^{cen}$</td>
<td>$\phi_{y,a}^{cen}$</td>
<td>$\phi_{z,a}^{cen}$</td>
<td>$\phi_{a}^{cen}$</td>
<td></td>
</tr>
<tr>
<td>$\phi_{x,a}^{cen}$</td>
<td>$\phi_{y,a}^{cen}$</td>
<td>$\phi_{z,a}^{cen}$</td>
<td>$\phi_{a}^{cen}$</td>
<td></td>
</tr>
</tbody>
</table>

TABLE III
THE JOINT ANGLES OF THE NECK, LEFT ARM AND RIGHT HANDS.

<table>
<thead>
<tr>
<th>Joint</th>
<th>$\phi_{x,a}^{cen}$</th>
<th>$\phi_{y,a}^{cen}$</th>
<th>$\phi_{z,a}^{cen}$</th>
<th>$\phi_{a}^{cen}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\phi_{x,a}^{cen}$</td>
<td>$\phi_{y,a}^{cen}$</td>
<td>$\phi_{z,a}^{cen}$</td>
<td>$\phi_{a}^{cen}$</td>
<td></td>
</tr>
<tr>
<td>$\phi_{x,a}^{cen}$</td>
<td>$\phi_{y,a}^{cen}$</td>
<td>$\phi_{z,a}^{cen}$</td>
<td>$\phi_{a}^{cen}$</td>
<td></td>
</tr>
<tr>
<td>$\phi_{x,a}^{cen}$</td>
<td>$\phi_{y,a}^{cen}$</td>
<td>$\phi_{z,a}^{cen}$</td>
<td>$\phi_{a}^{cen}$</td>
<td></td>
</tr>
<tr>
<td>$\phi_{x,a}^{cen}$</td>
<td>$\phi_{y,a}^{cen}$</td>
<td>$\phi_{z,a}^{cen}$</td>
<td>$\phi_{a}^{cen}$</td>
<td></td>
</tr>
</tbody>
</table>

2 [13]. A HRP-2 model of this simulation is modified from an original HRP-2 model. The arrangement of wrist axes is modified original HRP-2 [14]. Fig. 4 illustrates the HRP-2 model and its notations. The objective function is chosen as follows as an example.

$$f(x) = -f_{max}.$$  \hspace{1cm} (15)

In order to simplify the problem, the joint angles of the left arm, neck and hand are not included in decision variables. Tables I and II show the initial, minimum and maximum values of the decision variables. $\phi_{i}^{cen}$ expresses the chest and neck joint angles. The range of these decision variables are decided based on hardware limitation of HRP-2 and the initial values are chosen heuristically. If $P_x$, $L_x$ and $L_y$ are huge, the robot cannot accelerate the body during acceleration phase. Therefore, $P_x$, $L_x$ and $L_y$ are limited. The ranges of these moments are decided in moderation. It is a future work to decide these limits theoretically considering the robot’s actuator performance. Table III shows the joint angles which are not included in the decision variables. $\phi_{i}^{cen}$, $\phi_{x,a}^{cen}$ and $\phi_{y,a}^{cen}$ are joint angles, right hand joint angle and left arm joint angles including its hand, respectively. The angles are decided intuitively to avoid collision between the left arm and the body.

The constraint conditions are shown in Table IV. The height and mass of the target box are 0.7 (m) and 5 (kg), respectively. Therefore, $z_{bot}$ is 0.7 (m). The other constraint conditions are examples of the conditions for the punching motion.

This optimization problem is solved by using the technical computing language MATLAB (The MathWorks, Inc.) and Optimization Toolbox. The motion is generated in 633 (s) by two 3 (GHz) Intel Xeon X5365 CPUs.

Fig. 6 shows transition from the initial posture to an optimal posture. The initial posture does not satisfy the constraint conditions. For example, the height of the hand is lower than the required position. The optimal posture satisfy all the constraint conditions. As shown in Fig. 6 (b), the robot’s torso shifts forward. ZMP moves backward when the robot receives impulsive force. In order to extend the margin of the support polygon backward, the robot’s center of mass shifts forward. Table I and II show the optimized result of the decision variables. The hand collides with the box at 3.0 (s). In order to extend the support polygon in front-back direction, $f_{rx}$ reaches its maximum limit 0.3 (m). To hit the box strongly, $L_x$ and $L_y$ reach their maximum limit.

In order to generate motions for acceleration and slowdown phases, interpolation time $t_a$ and $t_s$ of $v_c$ are $0.74$ (s) and $0.74$ (s), respectively. And $t_a$ and $t_s$ of $P_x$, $L_x$, $L_y$, $v_{x,y}$ and $v_{y,z}$ are 0.7 (s) and 0.7 (s), respectively. In Fig. 7, the dotted lines show the momentum and angular momentum of the generated motion. The momentum and angular momentum are obtained by inverse dynamics on the assumption that the right foot is fixed on the ground from the generated motion. The computational results of $P_x$, $L_x$ and $L_y$ at the time of impact are respectively 2.7 (Ns), 3.0 (Nms) and 3.0 (Nms) as shown in Table I. In Fig. 7, the dotted lines show that $P_x$, $L_x$ and $L_y$ are smoothly interpolated from zero and to zero during the acceleration and slowdown phases.

Fig. 8 shows snapshots of the OpenHRP3 [15] simulation result. The simulation condition, e.g. contact model between the hand and the box, is the same with [6]. As shown in these figures, the robot dexterously accelerates and slows down the body by twisting its torso. Since the momentum in $Z_b$ direction is positive at time of impact, the robot unbends the body after the impact as shown in Fig. 8. Fig. 9 shows the estimated impulsive force in the optimizing process and OpenHRP3 simulation result. The estimated and actual peak force are 734.9 (N) and 722.0 (N) respectively and the error is $-1.8$ (%). The estimated and actual impulses are 15.1 (Ns) and 15.0 (Ns) as shown in Table II.
TABLE IV
CONSTRAINT CONDITIONS.

<table>
<thead>
<tr>
<th>$z_{ta}$ (m)</th>
<th>$z_{te}$ (m)</th>
<th>$\min R_{ce}$</th>
<th>$\min R_{fe}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.7</td>
<td>0.01</td>
<td>0.9962</td>
<td>1.0</td>
</tr>
<tr>
<td>$\omega_{y,ce}$ (°/s)</td>
<td>$\omega_{z,ce}$ (°/s)</td>
<td>$\omega_{y,fe}$ (°/s)</td>
<td>$\omega_{z,fe}$ (°/s)</td>
</tr>
<tr>
<td>0.3</td>
<td>0.3</td>
<td>5.0</td>
<td>5.0</td>
</tr>
<tr>
<td>$v_{fe}$ (m/s)</td>
<td>$\omega_{fe}$ (°/s)</td>
<td>$l_{Z_{imp}}$ (m)</td>
<td>$l_{Z_{as}}$ (m)</td>
</tr>
<tr>
<td>5.0</td>
<td>0.003</td>
<td>0.9962</td>
<td>1.0</td>
</tr>
<tr>
<td>$C_{o}$ (°)</td>
<td>$C_{o}$</td>
<td>$\min P_{z}$ (Ns)</td>
<td>$\max P_{z}$ (Nms)</td>
</tr>
<tr>
<td>5.0</td>
<td>8.0</td>
<td>−3.0</td>
<td>5.0</td>
</tr>
<tr>
<td>$\min P_{y}$ (Ns)</td>
<td>$\max P_{y}$ (Ns)</td>
<td>$\min L_{z}$ (Nms)</td>
<td>$\max L_{z}$ (Nms)</td>
</tr>
<tr>
<td>−5.0</td>
<td>−5.0</td>
<td>5.0</td>
<td>5.0</td>
</tr>
</tbody>
</table>

**Fig. 6.** Comparison between the initial posture and optimal posture.

The results indicate this motion uses the support polygon fully.

In order to see the trajectory in time domain, Fig. 11 shows the relationship between ZMP and time. In the figure, the blue line and red line indicate the predicted ZMP trajectory and OpenHRP3 result, respectively. The blue line computed by inverse dynamics from the reference motion. Therefore, the behavior of ZMP during the impact phase does not consider. The dotted lines indicate the impact phase. Except for the impact phase, the trend of the estimated ZMP trajectories during the acceleration and slowdown phase is similar to the OpenHRP3 simulation. The behavior of ZMP during the impact phase is predicted by the proposed SDC model. Fig. 11 shows the detail during the impact phase. The blue line is computed by SDC. The dotted line indicates the end of the impact phase. The trend of the estimated ZMP trajectories during the impact phase is similar to the OpenHRP3 simulation. Therefore, this scheme predicts ZMP trajectories accurately for all phases and optimizes the motion.

**Fig. 7.** Reference and actual momentum and angular momentum in $\Sigma_b$.

**Fig. 8.** Simulation of the optimized motion.

In order to maximize the impulsive force exerted on a target, the impact motion is generated through optimization.

V. CONCLUSIONS

In order to see the trajectory in time domain, Fig. 11 shows the relationship between ZMP and time. In the figure, the blue line and red line indicate the predicted ZMP trajectory and OpenHRP3 result, respectively. The blue line computed by inverse dynamics from the reference motion. Therefore, the behavior of ZMP during the impact phase does not consider. The dotted lines indicate the impact phase. Except for the impact phase, the trend of the estimated ZMP trajectories during the acceleration and slowdown phase is similar to the OpenHRP3 simulation. The behavior of ZMP during the impact phase is predicted by the proposed SDC model. Fig. 11 shows the detail during the impact phase. The blue line is computed by SDC. The dotted line indicates the end of the impact phase. The trend of the estimated ZMP trajectories during the impact phase is similar to the OpenHRP3 simulation. Therefore, this scheme predicts ZMP trajectories accurately for all phases and optimizes the motion.

**Fig. 7.** Reference and actual momentum and angular momentum in $\Sigma_b$.

**Fig. 8.** Simulation of the optimized motion.

In order to maximize the impulsive force exerted on a target, the impact motion is generated through optimization.
method considering postural stability during acceleration, impact and slowdown phases. With this scheme, the stability during the acceleration and slowdown phase can be evaluated in the optimizing process without computing inverse dynamics. The impact motion generated by this scheme is evaluated by performing simulations. The estimated force and ZMP trajectories are similar to OpenHRP3 simulation result. This scheme will be evaluated by a real humanoid robot in a future study.

REFERENCES