

A Fast Generation Method of a Dynamically Stable Humanoid Robot Trajectory with Enhanced ZMP Constraint

Satoshi Kagami¹, Koichi Nishiwaki¹, Tomonobu Kitagawa²,
Tomomichi Sugihara¹, Masayuki Inaba¹, and Hirochika Inoue¹

¹ Dept. of Mechano-Informatics, Univ. of Tokyo, Bunkyo-ku, Tokyo 113-8656, JAPAN
 [{kagami,nishi,zhidao,inaba,inoue}@jsk.t.u-tokyo.ac.jp,](mailto:{kagami,nishi,zhidao,inaba,inoue}@jsk.t.u-tokyo.ac.jp)
<http://www.jsk.t.u-tokyo.ac.jp>

² Keyence Co. Ltd., Osaka, JAPAN

Abstract. This paper describes a fast dynamically stable trajectory generation method for a humanoid robot. From a given input motion and the desired ZMP trajectory, the algorithm generates a dynamically stable trajectory using the relationship between the robot's center of gravity and the ZMP. Three key issues are denoted: 1) an enhanced ZMP constraint which enables the calculation of robot stability even if several limbs are contacting the environment, 2) a simplified robot model is introduced that represents the relationship between its center of gravity and ZMP, 3) a convergence method is adopted to eliminate approximation errors arising from the simplified model. Combining these three key issues together with online ZMP compensation method, humanoid robot H5 have succeeded to walk, step down and so on. Experimental results using humanoid robot H5 are described.

1 Introduction

Recently research on a humanoid type robot is active field in robotics society, and many remarkable elemental functions are proposed. Especially soft skin, 3D vision, motion planning and other topics are very much progressing. However in order to achieve a humanoid robot which works in a human world together with human being, not only these elemental functions but also dynamic walking function is required. Therefore algorithms for maintaining dynamic stability are central to humanoid robot control. Recent advances in computing hardware have enabled increasingly sophisticated physically-based simulation techniques to be utilized for the offline generation of dynamically-stable motions for complex robots, such as humanoid robots(ex. [YN00]).

This paper presents an offline algorithm for generating dynamically-stable compensation motions for humanoid robots. Given input motion and the desired ZMP trajectory, the algorithm generates a modified dynamically-stable motion for a humanoid robot. The system consists of three parts: 1) enhanced ZMP constraint which enables to calculate robot stability even several limbs are attaching to the environment, and it can be applied not only single/dual leg(s) motion but also limbs motion if limbs has 6 DOFs, 2) humanoid robot dynamics model by representing its center of gravity, so that differential equation of center of gravity and ZMP can be obtained, and then this equation is solved using trinomial expression, 3) converge method is adopted to treat model error to the real robot. Combining these three key issues together with online ZMP compensation method, humanoid robot H5 have succeeded to walk, step down and so on. Experimental results by using humanoid type robot H5 are denoted.

2 A Fast ZMP Tracking Trajectory Generation Method

So far, there are remarkable biped robots have been proposed, and their trajectory generation method can be devided into two major schemes: 1) online simple model based trajectory generation method, and 2) offline position-based trajectory generation method. The former method has many advantages, especially calculation simpleness and adaptability for various environemnt. Remarkable issues have been proposed for biped type robots[KT96,PDP97].

However, since a humanoid robot has many degrees of freedom, position-based offline trajectory generation has been adopted mostly using a ZMP[VFD70] constraint. Currently, methods for calculating dynamically stable trajectories can be categorized into three ways: a) heuristic search such as GA (genetic algorithm), b) problem optimization method such as optimal gradient method, and c) model simplification with iteration. Several remarkable issues have been proposed using these schemes mostly applying to a walking pattern generation for a real humanoid type robot[YTK93,HIR97,NII99a,KKT⁺00].

Heuristics search method or problem optimization method for find out a dynamically stable motion trajectory under ZMP constraint are considered as a general method, however these methods require to calculate inverse

dynamics in every iteration, and it requires many computation time. We propose a fast trajectory generation method by using a relationship between robot center of gravity and ZMP. This algorithm is belonging to category c) in above.

2.1 Dynamics Model of Humanoid Type Robot

First, we introduce a model of humanoid type robot by representing motion and rotation of the center of the gravity (COG). Set z axis be the vertical axis, and x and y axis be the other component of sagittal and lateral plane respectively. Set m_i , $\mathbf{r}_i = (x_i, y_i, z_i)$, w_i , \mathbf{I}_i be weight, position, angle velocity, inertia moment of i th link respectively. Let total mass of the robot be m_{total} , and total center of the gravity be $\mathbf{r}_{cog} = (r_{cog_x}, r_{cog_y}, r_{cog_z})$. Then they are represented as follows:

$$m_{total} = \sum m_i \quad (1)$$

$$r_{cog_x} = \frac{\sum m_i r_{x_i}}{m_{total}} \quad (2)$$

$$r_{cog_y} = \frac{\sum m_i r_{y_i}}{m_{total}} \quad (3)$$

$$r_{cog_z} = \frac{\sum m_i r_{z_i}}{m_{total}} \quad (4)$$

Let moment around its center of gravity be \mathbf{M}_{cog} , total force that robot obtains be $\mathbf{f} = (f_x, f_y, f_z)$ and total moment around a point $\mathbf{p} = (p_x, p_y, p_z)$ be \mathbf{T} , then dynamic equation around a point \mathbf{p} is approximately represented as follows:

$$\begin{aligned} m_{total}(\mathbf{r}_{cog} - \mathbf{p}) \times (\ddot{\mathbf{r}}_{cog} + \mathbf{g}) + \mathbf{M}_{cog} - \mathbf{T} &= 0 \\ \mathbf{f} &= m_{total}(\ddot{\mathbf{r}}_{cog} + \mathbf{g}) \end{aligned} \quad (5)$$

ZMP $\mathbf{p}_{cog} = (p_{cog_x}, p_{cog_y})$ around point $\mathbf{p} = (p_x, p_y, h)$ on the horizontal place $z = h$ is defined as a point where moment around point \mathbf{p} be $\mathbf{T} = (0, 0, Tz)$, and it can be calculated from Equation. 5.

$$\begin{aligned} p_{cog_x} &= r_{cog_x} - \frac{M_{cog_y} - m_{total}(r_{cog_z} - h)(\ddot{r}_{cog_x} + g)}{m_{total}(\ddot{r}_{cog_z} + g)} \\ p_{cog_y} &= r_{cog_y} - \frac{M_{cog_x} - m_{total}(r_{cog_z} - h)(\ddot{r}_{cog_y} + g)}{m_{total}(\ddot{r}_{cog_z} + g)} \end{aligned} \quad (6)$$

Let $h = 0$ in Equation. 6 and using Equation. 5, then ZMP can be calculated as follows when desired robot motion has been achieved[NII98].

$$\begin{aligned} p_{cog_x} &= r_{cog_x} - \frac{M_{cog_y} - m_{total}r_{cog_z}(\ddot{r}_{cog_x} + g)}{f_z} \\ p_{cog_y} &= r_{cog_y} - \frac{M_{cog_x} - m_{total}r_{cog_z}(\ddot{r}_{cog_y} + g)}{f_z} \end{aligned} \quad (7)$$

2.2 Stabilization by Horizontal Center of Gravity Position Modification

Let $\mathbf{p}_{cog}^*(t)$ be the given ideal ZMP trajectory, and $\mathbf{WBT}(t)$ be the whole body trajectory (ex. walking motion trajectory). When robot moves along given $\mathbf{WBT}(t) = \mathbf{r}^o(t)$, then resulting moment \mathbf{M}^o , force \mathbf{f}^o , ZMP \mathbf{p}_{cog}^o , center of gravity \mathbf{r}_{cog}^o is calculated.

Problem statement and compensation scheme are defined as follows:

Problem Statement: For given ideal ZMP trajectory $\mathbf{p}_{cog}^*(t)$ and given input body trajectory $\mathbf{WBT}(t) = \mathbf{r}^o(t)$, calculate an approximate new trajectory $\mathbf{r}_{cog}^*(t)$ that causes a new ZMP trajectory $\mathbf{p}_{cog}(t)$ which is close enough to the given ideal ZMP trajectory $\mathbf{p}_{cog}^*(t)$.

From Equation. 7, following equations is obtained for both in ideal and current \mathbf{p}, \mathbf{r} respectively.

$$\begin{aligned}\mathbf{p}_{cog_x}(t) &= \mathbf{r}_{cog_x}(t) - \frac{\mathbf{M}_{cog_x}^o(t) - m_{total}\mathbf{r}_{cog_z}(t)(\ddot{\mathbf{r}}_{cog_x}^o(t) + g)}{f_z^o(t)} \\ \mathbf{p}_{cog_x}^*(t) &= \mathbf{r}_{cog_x}^*(t) - \frac{\mathbf{M}_{cog_x}^*(t) - m_{total}\mathbf{r}_{cog_z}^*(t)(\ddot{\mathbf{r}}_{cog_x}^*(t) + g)}{f_z^*(t)}\end{aligned}\quad (8)$$

Compensation Scheme: In order to simplify Equation. 7, only horizontal modification of the body trajectory is considered.

Since only horizontal compensation motion of the body is considered, $\mathbf{r}_{cog_z}^o = \mathbf{r}_{cog_z}^*$. Then, two assumptions are introduced:

Assumption 1 We assume that effect to the force $\mathbf{f}(t)$ that robot obtains from its self motion is small enough. Therefore,

$$\mathbf{f}_z^o(t) = \mathbf{f}_z^*(t) \quad (9)$$

Assumption 2 We assume that effect to the torque around center of gravity that robot obtains $\mathbf{M}_{cog}(t)$ from its self motion is small enough. Therefore

$$\mathbf{M}^o(t) = \mathbf{M}^*(t) \quad (10)$$

With these assumptions, and let $\mathbf{p}_{cog}^{err}(t)$ be an error between ideal ZMP $\mathbf{p}_{cog}^*(t)$ and current ZMP $\mathbf{p}_{cog}(t)$, and $\mathbf{r}_{cog}^{err}(t)$ be the an error between ideal center of gravity trajectory $\mathbf{r}_{cog}^*(t)$ and current trajectory $\mathbf{r}_{cog}(t)$.

$$\begin{aligned}\mathbf{p}_{cog}^{err}(t) &= \mathbf{p}_{cog}^*(t) - \mathbf{p}_{cog}(t) \\ \mathbf{r}_{cog}^{err}(t) &= \mathbf{r}_{cog}^*(t) - \mathbf{r}_{cog}(t)\end{aligned}\quad (11)$$

Therefore following result is obtained from Equation. 8 and Equation. 11.

$$\mathbf{p}_{cog}^{err}(t) = \mathbf{r}_{cog}^{err}(t) - \frac{m_{total}\mathbf{r}_{cog_z}(t)\ddot{\mathbf{r}}_{cog}^{err}(t)}{f_z^o(t)} \quad (12)$$

2.3 Solving Differential Equation

Equation. 12 can be solved as subtract approximation. By discretizing Equation. 12 with small time step Δt with iteration i , ($i = 0, 1, 2, \dots, n-1, n$),

$$\begin{aligned}\mathbf{p}_{cog}^{err}(t) &\rightarrow \mathbf{p}_{cog}^{err}(i) \\ \mathbf{r}_{cog}^{err}(t) &\rightarrow \mathbf{r}_{cog}^{err}(i) \\ \ddot{\mathbf{r}}_{cog}^{err}(t) &\rightarrow \frac{\mathbf{r}_{cog}^{err}(i+1) - 2\mathbf{r}_{cog}^{err}(i) + \mathbf{r}_{cog}^{err}(i-1)}{\Delta t^2}\end{aligned}\quad (13)$$

Then trinomial expression which satisfies $\mathbf{r}_{cog}^{err}(i)$ is obtained when $1 \leq i \leq n-1$.

$$a_i \mathbf{r}_{cog}^{err}(i-1) + b_i \mathbf{r}_{cog}^{err}(i) + c_i \mathbf{r}_{cog}^{err}(i+1) = d_i \quad (14)$$

Here,

$$\begin{aligned}
a_i &= -\frac{m_{total}r_{cog_z}^o(i)}{f_z^o(i)\Delta t^2} \\
b_i &= 1 + 2\frac{m_{total}r_{cog_z}^o(i)}{f_z^o(i)\Delta t^2} \\
c_i &= -\frac{m_{total}r_{cog_z}^o(i)}{f_z^o(i)\Delta t^2} \\
d_i &= p_{cog}^{err}(i)
\end{aligned} \tag{15}$$

Then using boundary condition of trinomial expression, boundary condition $i = 0, i = n$ is calculated. In this paper, we fix terminal position. If statically stable posture is given as the terminal posture, both end of resulted trajectory will not be moving.

- Since terminal position is fixed, x_0, x_n are given.
- From position and acceleration of center of gravity relationship, number of variables is $n - 1$ from $t = 1$ to $t = n - 1$.
- From ZMP constraint, number of variables is $n + 1$ from $t = 0$ to $t = n$.

As for boundary condition, terminal velocity is indefinite, we set the following boundary conditions.

$$\begin{aligned}
b_0 &= b_n = 1 \\
a_0 &= a_n = 0 \\
c_0 &= c_n = 0
\end{aligned} \tag{16}$$

Given coefficient matrix, trinomial expression is solved, and discrete r_{cog}^{err} is calculated.

2.4 Enhanced ZMP Constraint

Original ZMP constraint is designed for only one leg is supporting the body and the leg should be on a ground plane (vertical plane along to the gravity direction). For example in dual leg state on a floor, it is enhanced as a virtual sole which covers both legs sole like a convex hull, so that ZMP constraint is calculated on this virtual sole.

Since humanoid type robot has four limbs, body may be supported by many points. Therefore, we introduce a case that supporting points are placed only one plane which is slant to the ground plane (Fig. 1(a)). Introducing coordinate system shown in Fig. 1(a), let total torque around a point $\mathbf{p}' = (p'_x, p'_y, p'_z)$ be T' , then dynamic equation under this assumption and moment equation around a point \mathbf{p}' is approximately represented as follows:

$$m_{total}(\mathbf{r}_{cog}' - \mathbf{p}') \times (\ddot{\mathbf{r}}'_{cog} - \mathbf{g}') - \mathbf{M}'_{cog} - \mathbf{T}' = 0 \tag{17}$$

Since $\mathbf{g}' = (-g \sin \theta, 0, -g \cos \theta)$, ZMP $\mathbf{p}'_{cog} = (p'_{cog,x}, p'_{cog,y})$ around point $\mathbf{p}' = (p'_x, p'_y, h)$ on the place $z = h$ can be represent as follows when moment is $\mathbf{T}' = (0, 0, T' z)$.

$$\begin{aligned}
p'_x &= \frac{m_{total}(\ddot{r}'_{z_i} + g \cos \theta)r'_{x_i} - m_{total}(r'_{z_i} - h)(\ddot{r}'_{x_i} + g \sin \theta) + \sum_i M'_{x_i}}{m_{total}(\ddot{r}'_{z_i} + g \cos \theta)} \\
p'_y &= \frac{m_{total}(\ddot{r}'_{z_i} + g \cos \theta)r'_{y_i} - m_{total}(r'_{z_i} - h)\ddot{r}'_{y_i} + \sum_i M'_{y_i}}{m_{total}(\ddot{r}'_{z_i} + g \cos \theta)}
\end{aligned} \tag{18}$$

Then, we introduce a case that supporting points are placed on plural planes (Fig. 1(b)). In this case, there are combination of supporting points which consists of placed on one plane. When one of a ZMP constraints satisfies Equation. 18, motion trajectory will be dynamically stable. Assuming that each limb supports the body only using its end-effector, there exists at most four plane, and it is not too hard to calculate these combinations.

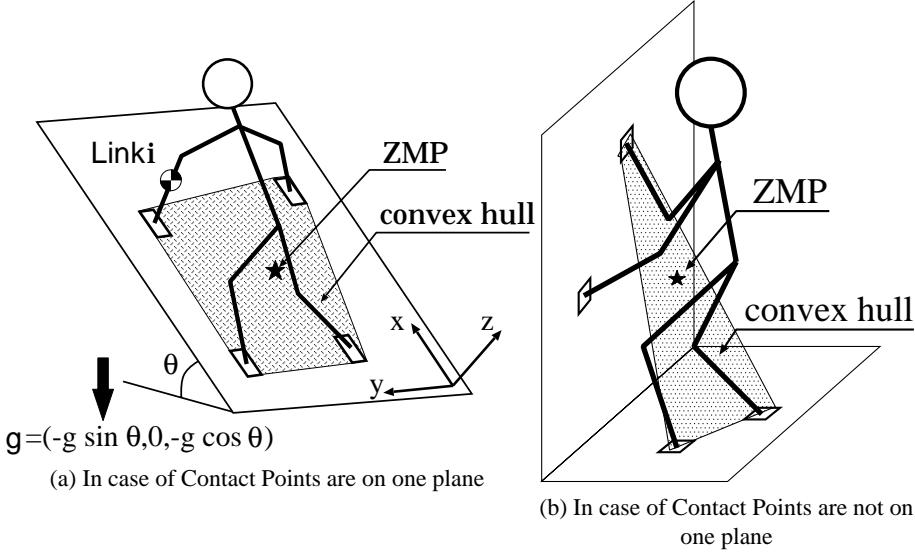


Fig. 1. Enhanced ZMP

3 Implementation

3.1 Model Error Convergence

Using approximated modelling in section above, we can calculate a error trajectory of center of gravity $\mathbf{r}_{cog}^{err}(t)$ from given ideal ZMP trajectory $\mathbf{p}_{cog}^*(t)$ and given whole body trajectory $\mathbf{WBT}(t)$. However, there exists some model error with $\mathbf{r}_{cog}^{err}(t)$, since it is a simplified approximation model. Furthurmore, given $\mathbf{WBT}(t) = \mathbf{r}_{cog}(t)$ is hard to control directly in a horzonatal plane. In order to solve this problem, $\mathbf{boldvecr}_{torso}(t)$ is selected as for a control variables of the robot by introducing a following assumption.

Assumption 3 Since horizontal motion of the torso is assumed, assume linear relationship between motion of torso $\Delta(\mathbf{r}_{torso}(t))$ and motion of the center of gravity $\Delta(\mathbf{r}_{cog}(t))$.

$$\Delta(\mathbf{r}_{torso}(t)) \cong k_{cog} \Delta(\mathbf{r}_{cog}(t)) \quad (19)$$

From this assumption, following model improvement equation is obtained using coefficient value k_{cog} .

$$\mathbf{r}_{cog}^{err}(t) = \mathbf{r}_{cog}^*(t) - k_{cog} \mathbf{r}_{torso}(t) \quad (20)$$

3.2 Implementation

We implemented the algorithm above by receiving torso, limbs and ZMP trajectories, then convert torso trajectory to track given ZMP. The algorithm has following iteration. We introduce a coefficient $K(0 < K < 1)$ for a trinomial expressions, in order to compensate several approximations. Then do a convergence to calculate a suitable dynamically stable torso trajectory.

1. Apply inverse dynamics $INV()$ to given whole body trajectory $\mathbf{r}(t)$ to calculate current $\mathbf{p}_{cog}(t)$. Then calculate the error $\mathbf{p}_{cog}^{err}(t)$ between given ideal ZMP $\mathbf{p}_{cog}^*(t)$ and current ZMP $\mathbf{p}_{cog}(t)$.
2. Calculate torso trajectory difference $\mathbf{r}_{cog}^{err}(t)$, and estimate torso trajectory $\mathbf{r}_{torso}(t)$ using Equation. 20 by applying trinomial expressions with coefficient $K(0 < K < 1)$.
3. Let error E be an average from the given ZMP $\mathbf{p}_{cog}^* * t$, and evaluate it using following equation.

$$E = \frac{1}{step} \sum_i \| \mathbf{p}_{error,i} \| \quad (21)$$

4. Let error of j^{th} iteration be E_j , and stop this algorithm when $E_j \leq CE_{j-1}$. C is a constant threshold, and in this paper $C = 0.95$.
5. Back to 1 step.

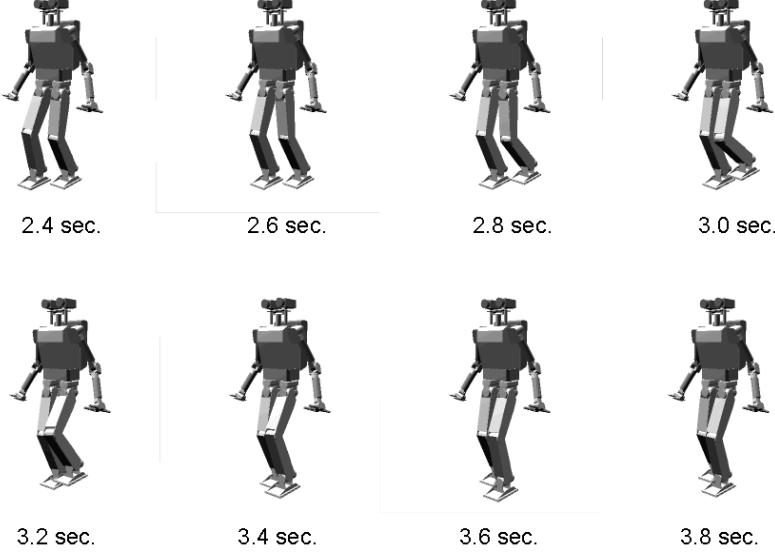


Fig. 2. Walking Pattern

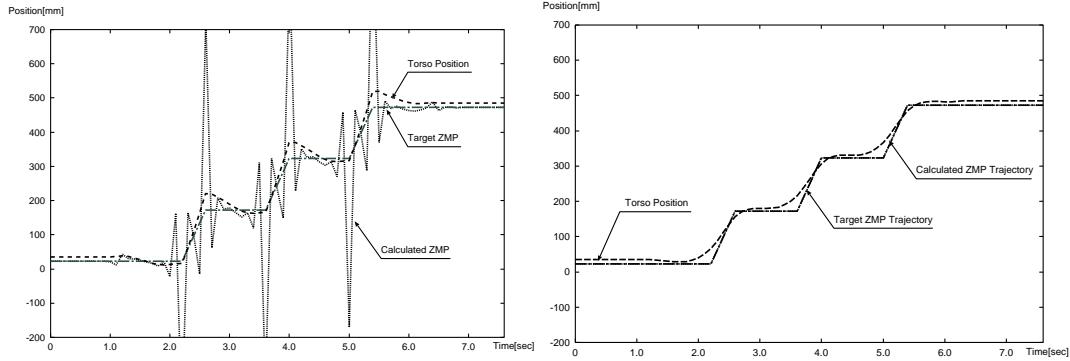


Fig. 3. Input and Output ZMP, Torso Trajectories for Front (x) Direction

3.3 Inverse Dynamics Calculation

Inverse dynamics calculation is utilized to calculate $\mathbf{p}_{cog}(t)$ from given input trajectory $\mathbf{WBT}(t)$. The inverse dynamics is the problem to find out a set of input torque of each joints $\boldsymbol{\tau}$ with given joint angle $\boldsymbol{\theta}$, joint angle velocity $\dot{\boldsymbol{\theta}}$ and joint angle acceleration $\ddot{\boldsymbol{\theta}}$. Then inverse dynamics $INV()$ is represented as follows.

$$\boldsymbol{\tau} = INV(\boldsymbol{\theta}, \dot{\boldsymbol{\theta}}, \ddot{\boldsymbol{\theta}}) \quad (22)$$

Fast and general inverse dynamics calculation library[SNII00] is adopted in order to achieve fast trajectory generation. This library is originally developed for multi-limb body and appropriate link can be fixed to the world coordinates, so that inverse dynamics of any body configuration to the environment can be efficiently calculated.

The procedure for calculating $\mathbf{p}(t)$ by this algorithm is denoted as follows:

1. Set fixed link to the world as a supporting leg terminal link (foot),
2. Set robot joint angles, angle velocities, angle accelerations,
3. Recursively calculate link translation velocity acceleration, angular velocity and acceleration respectively from root link,
4. Calculate root link translation velocity, acceleration, angular velocity and acceleration respectively, from the relative motion between fixed link and root link, then re-calculate all link motion,

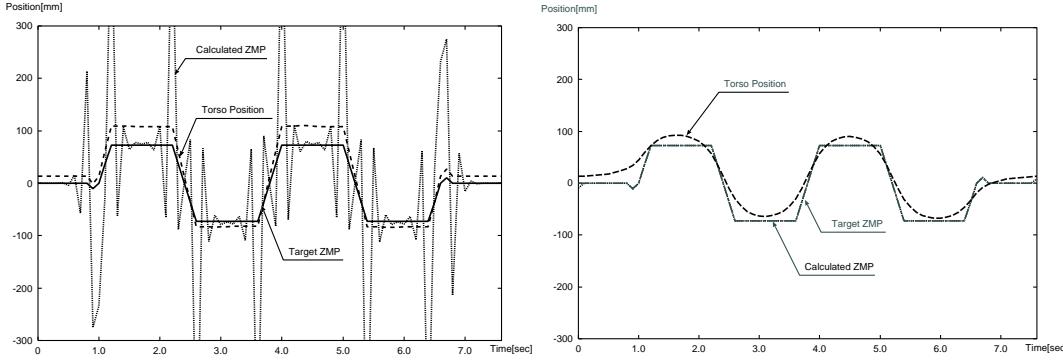


Fig. 4. Input and Output ZMP, Torso Trajectories for Side (y) Direction

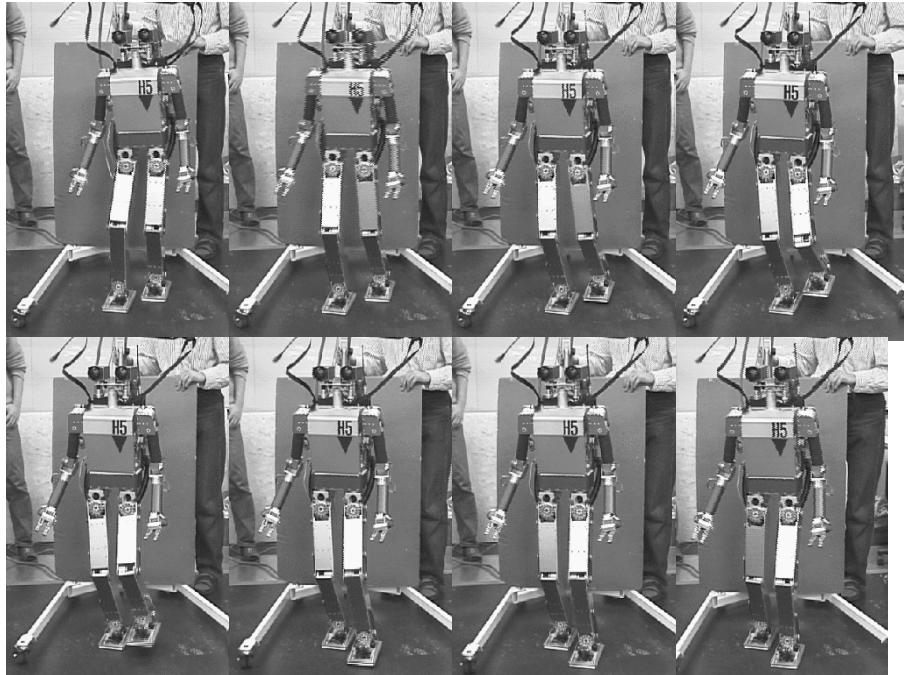


Fig. 5. H5 Walking Experiment

5. Recursively calculate link force and moment from terminal link,
6. Calculate ZMP from root link force and moment.

The algorithm is based on Newton-Euler method[JMR80], and tree-like link structure is known to be solved as follows[MD82]. Computational complexity of this part is $O(n)$, where n is number of DOFs.

3.4 Online ZMP Compensation Method by Horizontal Torso Movement

Humanoid type robot is hard to “replay” the given trajectory in openloop in real world even given trajectory satisfies the ZMP constraint. Therefore several local compliance control methods have been proposed [Hon93a,Hon93b,Hon98]. In this paper, we adopted a torso position compliance method to track a given ZMP trajectory[NII99b]. This method tries to track a given ZMP trajectory by the horizontal motion of the torso. It consists of two parts, one is ZMP tracking mechanisms, and the other is inverse pendulum control to keep its dynamic balance.

In ZMP tracking mechanisms, process is same as in a one step of trinomial expression in Equation. 14. In inverse pendulum control, think about inverse pendulum with desired ZMP as the supporting point. Let l be a distance from the supporting point to the mass center of gravity,, θ be an angle from the perpendicular to the

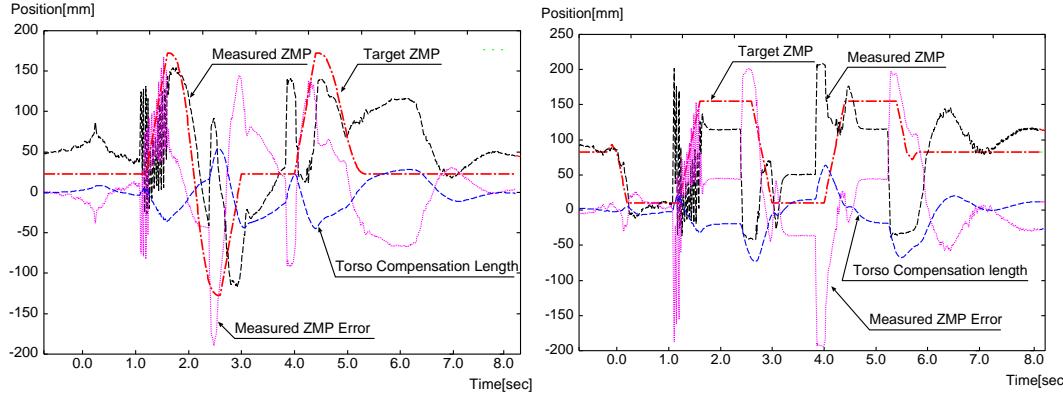


Fig. 6. Desired ZMP, Real ZMP, Error and Torso Position for Front x and Side y Direction

gravity direction, a be an acceleration to the supporting point. Then linearized dynamic equation is

$$l\theta = g\theta + a \quad (23)$$

Assuming PD control $Q(1 + T_D\theta)$, then transmission function is $Q(1 + T_D s) / (ls^2 - g)$. When $Q \geq g$ then system is stable. Therefore, let d be the distance to the destination,

$$\ddot{d} = Q(\theta + T_D\dot{\theta}) \quad (24)$$

Discretize \ddot{d} and desired movement of the center of gravity are obtained as

$$\Delta p_i^d = Q(\theta + T_D\dot{\theta}) + (\Delta p_i^d - \Delta p_{i-1}^d) + \Delta p_i^d \quad (25)$$

3.5 Humanoid Robot H5

The H5 has 1300[mm] height and 33[kg] weight. It has 6 DOFs in each leg, 6 DOFs in each arm, 1 DOF for each finger, and 4 DOFs in each neck and vergence respectively. It has 12 force sensors at the bottom of each sole and measures the ZMP. It has a PC/AT clone on body which has PentiumIII-333MHz, and motor servo loop is 1msec cycle which is written as RT-Linux kernel module.

4 Trajectory Generation for Humanoid Robot H5

We apply this fast motion generation algorithm to our dynamic humanoid robot “H5”, and show efficiency of the algorithm.

4.1 Walking Trajectory Generation and Experiment

Modified walking pattern is shown in Fig. 2, from statically stable trajectory as for input. Input and modified, torso and ZMP trajectories are shown in Fig. 3 for front x and side y direction respectively.

We applied output motion mentioned in the section above to humanoid robot H5, and confirm its performance.

Fig. 6 shows output desired ZMP trajectory, real ZMP trajectory, error between desired and real ZMP and torso position for front x and side y direction respectively. For the first step, ZMP is mostly tracking the original trajectory. However, when changing the grounding leg, free leg is touched to the ground before its original trajectory, and since then ZMP trajectory occasionally shifted from the original one, however leg compliance feedback allows ZMP to track the original one.

4.2 Squat Down Trajectory Generation and Experiment

Output squat down motion with right leg is stepping forward pattern is shown in Fig. 7, from statically stable trajectory as for input. Input and modified, torso and ZMP trajectories for front x and side y direction respectively are shown in Fig. 8, 9.

We applied output motion mentioned in the section above to humanoid robot H5, and confirm its performance.

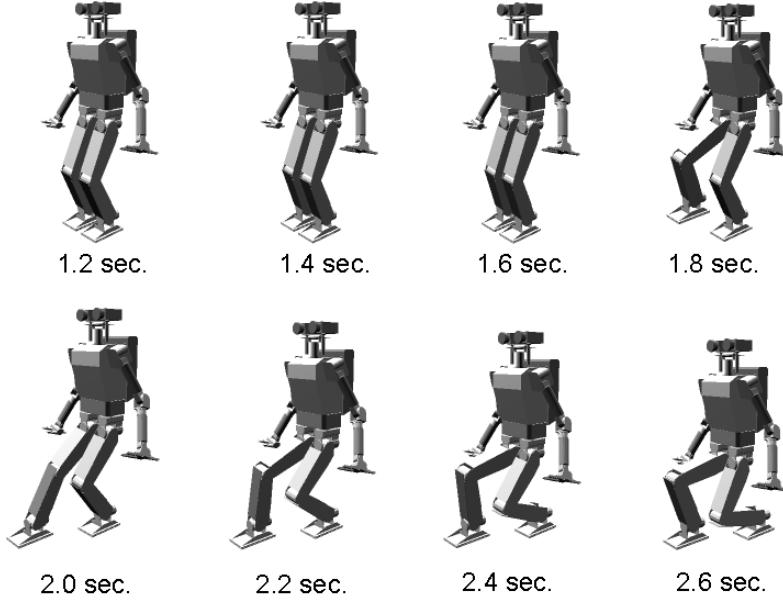


Fig. 7. Squat Down with Stepping Forward Pattern

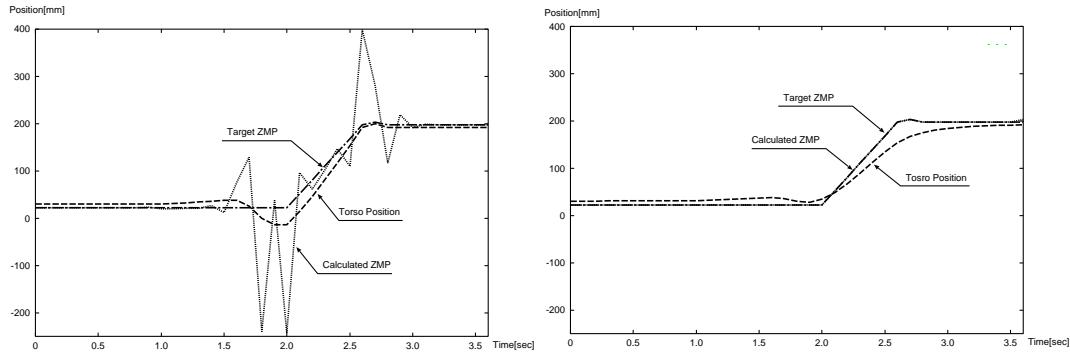


Fig. 8. Input and Output ZMP, Torso Trajectories for Front (x) Direction

4.3 Standing Up with Arms Trajectory Generation

Using enhanced ZMP constraints, not only a motion which attach to the environment with its feet, but also arm-leg cooperative motion can be generated. Output standing up with arms from sitting posture pattern is shown in Fig. 11, from statically stable trajectory as for input. Input and modified, torso and ZMP trajectories for front x direction is shown in Fig. 12.

4.4 Convergence and Performance Evaluation

Iteration number for decrease average ZMP error along trajectory E to be under 5[mm] is shown in Table 1. Fig. 13 shows a relationship between iteration number and error which is calculated in Equation. 21 for generating walking pattern. Most of the calculation time is consumed by solving inverse dynamics $INV()$, and calculation time is almost same for a real motion. For example, given walking pattern is 4sec and calculation time to make it dynamically stable is also about 4sec.

5 Concluding Remarks

In this paper, a trajectory generation method from given input to dynamically stable trajectory was proposed. This algorithm can be applied not only dual legs motion but also limbs motion if limbs has 6 DOFs. Three key issues are

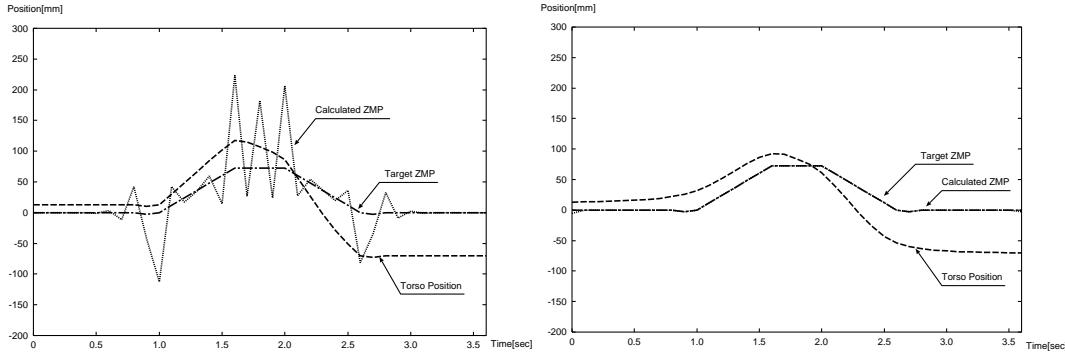


Fig. 9. Input and Output ZMP, Torso Trajectories for Side y Direction

Table 1. Gain K and Convergence

Gain K	K=0.1	K=0.3	K=0.5	K=0.7
Walk	27	9	5	26
Squat Down	18	6	4	—
Stand Up with Arms	29	9	5	—

denoted, 1) enhanced ZMP constraint which enables to calculate robot stability even several limbs are attaching to the environment, 2) simplified robot model of which mass is concentrated on its center of gravity, so that dynamic equation is solved using trinomial expression, 3) converge method is adopted to treat model error to the real robot. Furthermore, experimental results by using humanoid type robot H5 are denoted.

References

- [HIR97] Kazuo HIRAI. Current and Future Perspective of Honda Humanoid Robot. In *Proc. of 1997 IEEE Int'l. Conf. on Intelligent Robots and Systems (IROS'97)*, pages 500–508, 1997.
- [Hon93a] Honda.Co.Ltd. *Walking Control System for Legged Robot*. Japan Patent Office (A) 5-305583, 1993.
- [Hon93b] Honda.Co.Ltd. *Walking Control System for Legged Robot*. Japan Patent Office (A) 5-200682, 1993.
- [Hon98] Honda.Co.Ltd. *Walking Pattern Generation System for Legged Robot*. Japan Patent Office (A) 10-86080, 1998.
- [JMR80] J.Y.S.Luh, M.W.Walker, and R.P.C.Paul. On-Line Computational Scheme for Mechanical Manipulators. *ASME Journal of Dynamic Systems, Measurement, Control*, 102:69–76, 1980.
- [KKT⁺00] S. Kagami, F. Kanehiro, Y. Tamiya, M. Inaba, and H. Inoue. AutoBalancer: An Online Dynamic Balance Compensation Scheme for Humanoid Robots. In *Proc. of Fourth Intl. Workshop on Algorithmic Foundations on Robotics (WAFR'00)*, pages SA-79–SA-89, 2000.
- [KT96] S. Kajita and K. Tani. Experimental study of biped dynamic walking. *IEEE Control Systems*, 16(1):13–19, 1996.
- [MD82] M.W.Walker and D.E.Orin. Efficient Dynamic Computer Simulation of Robotic Mechanisms. *ASME Journal of Dynamic Systems, Measurement, Control*, 104:205–211, 1982.
- [NII98] Ken'ichirou NAGASAKA, Masayuki INABA, and Hirochika INOUE. Research on human-based genetic motion acquisition for humanoid. In *Proc. of the 16th Annual Conf. of Robotics Society of Japan*, pages 827–828, 1998.
- [NII99a] K. Nagasaka, M. Inaba, and H. Inoue. Walking Pattern Generation for a Humanoid Robot based on Optimal Gradient Method. In *Proc. of 1999 IEEE Int. Conf. on Systems, Man, and Cybernetics No. VI*, 1999.
- [NII99b] Ken'ichirou NAGASAKA, Masayuki INABA, and Hirochika INOUE. Stabilization of Dynamic Walk on a Humanoid Using Torso Position Compliance Control. In *Proceedings of 17th Annual Conference on Robotics Society of Japan*, pages 1193–1194, 1999.
- [PDP97] Jerry Pratt, Peter Delworth, and Gill Pratt. Virtual model control of a bipedal walking robot. In *Proc. of IEEE Int. Conf. Robotics and Automation*, pages 193–198, 1997.
- [SNII00] Tomomichi SUGIHARA, Koichi NISHIWAKI, Masayuki INABA, and Hirochika INOUE. Development of “z-dynaform” — library for dynamics analysis of rigid multibody. In *Proceedings of 18th Annual Conference on Robotics Society of Japan*, 2000.
- [VFD70] Vukobratović, A.A. Frank, and D.Juričić. On the Stability of Biped Locomotion. *IEEE Trans. on Biomedical Engineering*, 17(1):25–36, 1970.
- [YN00] Katsu YAMANE and Yoshihiko NAKAMURA. Dynamics Filter – Concept and Implementation of On-Line Motion Generator for Human Figures. In *Proc. of IEEE Int. Conf. on Robotics and Automation*, pages 688 – 694, 2000.



Fig. 10. H5 Squat Down Experiment

[YTK93] J. Yamaguchi, A. Takanishi, and I. Kato. Development of a Biped Walking Robot Compensating for Three-Axis Moment by Trunk Motion. *Journal of the Robotics Society of Japan*, 11(4):581–586, 1993.

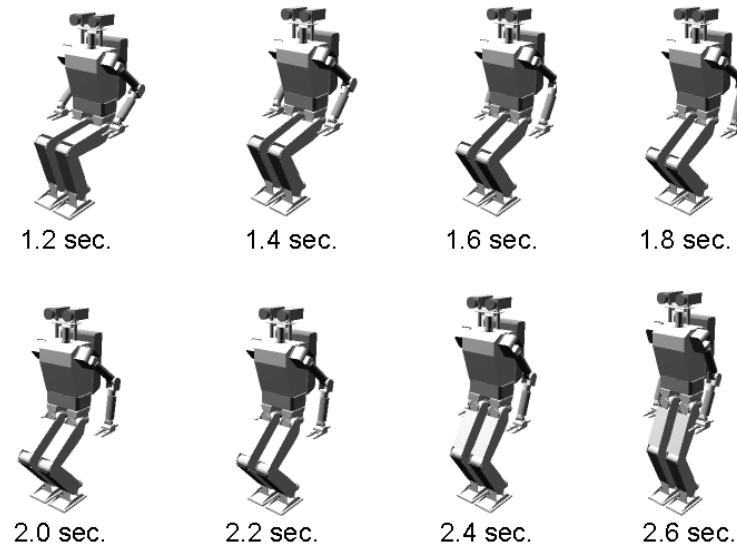


Fig. 11. Stand Up with Arms Pattern

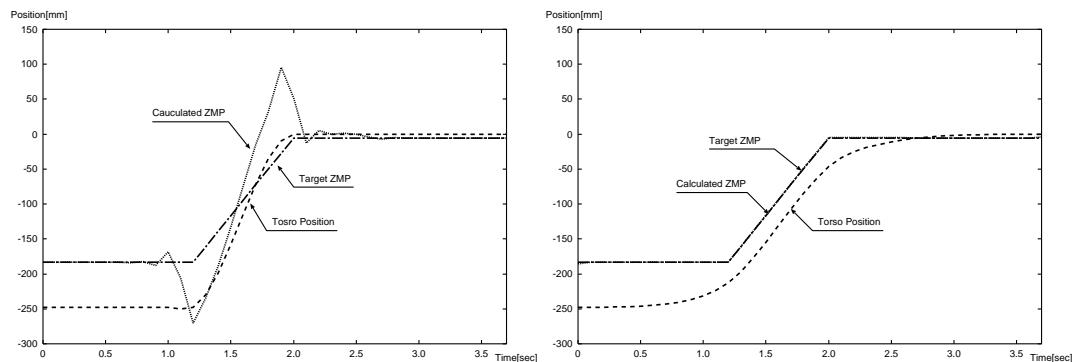


Fig. 12. Input and Output ZMP, Torso Trajectories for Front (x) Direction

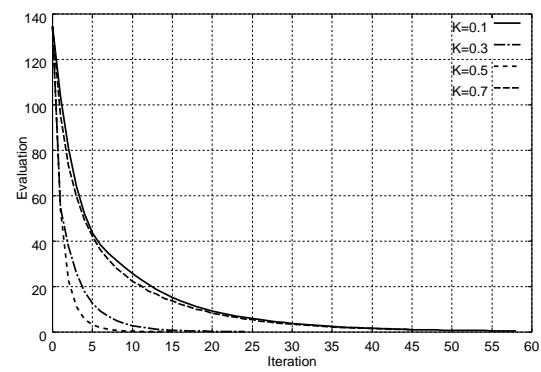


Fig. 13. Error Feedback Gain K and Convergence for Walking Pattern Generation