Proceedings of the 2006 IEEE/RSJ
International Conference on Intelligent Robots and Systems
October 9 - 15, 2006, Beijing, China

Biped Walking Pattern Generator allowing Auxiliary ZMP Control

Shuuji KAJITA, Mitsuharu MORISAWA, Kensuke HARADA, Kenji KANEKO,
Fumio KANEHIRO, Kiyoshi FUJIWARA and Hirohisa HIRUKAWA
AIST,Tsukuba, Ibaraki 305–8568
Email: {s.kajita,m.morisawa,kensuke.harada,k.kaneko,
f-kanehiro,k-fujiwara,hiro.hirukawa}@aist.go.jp

***Abstract*— A biped walking pattern generator which allows an additional ZMP control (auxiliary ZMP) is presented. An auxiliary ZMP is realized by an inverse system added to a pattern generator based on the ZMP preview control. To compensate the effect of the auxiliary ZMP, we apply virtual time shifting of the reference ZMP. As an application of the proposed method, a walking control on uneven terrain is simulated. The simulated robot can walk successfully by changing its walking speed as the side effect of the auxiliary ZMP control.**

I. Introduction

A number of successful biped humanoid robots have been adapting the Zero-Moment Point (ZMP) [1], [2] for their walking pattern generation and control. However, there exist criticisms that ZMP based walking controls lack the adaptivity which is observed in the passive dynamic walk [3], [4], the compliance based walking control [5], the central pattern generator (CPG) [6] or the hybrid zero dynamics [7].

As criticized, many ZMP based walking control had relied on off-line pattern generation due to its heavy computational demands, but there are works which had proposed adaptive walking control based on ZMP [8]–[11]. In this context, a concept of auxiliary ZMP is introduced for an adaptive walking pattern generation. The auxiliary ZMP modifies an instantaneous behavior of the walking robot and as its side effect the long term parameters like walking cycle or step width are modified to hold consistency of the system dynamics. By using the auxiliary ZMP as a control input, a pattern generator can be involved in a feedback loop, which potentially creates a robust and adaptive walking motion.

The rest of this paper is organized as follows. In Section II, a concept of auxiliary ZMP and its applications are explained. In Section III, we derive a ZMP based pattern generator and investigate its internal structure. In Section IV, we expand the pattern generator to realize the auxiliary ZMP. To compensate the effect of auxiliary ZMP, virtual time shifting of the reference ZMP is introduced in Section V. In Section VI, the auxiliary ZMP is used to stabilize a simulated humanoid robot walking on an uneven floor and its effectiveness is demonstrated. We conclude this paper and address the future remarks in Section VII.

II. Introducing auxiliary ZMP

Figure 1 illustrates the basic control schema we assume in this paper. A walking pattern generator creates reference joint angles from given walking parameters, for example, average speed, step width, ground profile etc. At the same time, the pattern generator calculates an expected Zero-Moment Point which is dynamically consistent with the reference motion.

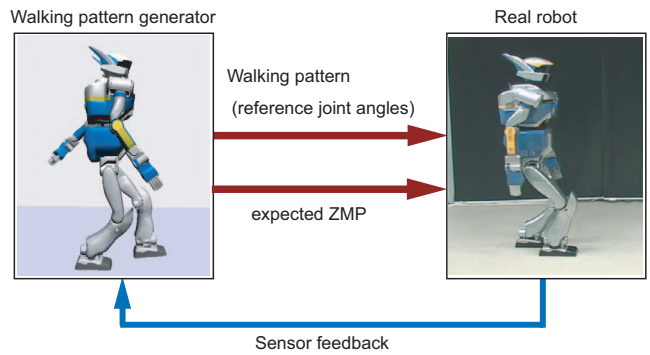


Fig. 1. Control schema of this paper

Now, let us assume this expected ZMP can be separated as following,

$$\text{(expected ZMP)} = p^{ref} + p^{aux}. \tag{1}$$

The first component p^{ref} represents global walking behavior such as average speed and step width. Since we must take into account of the global behavior and long term stability, we cannot flexibly change p^{ref}.

The second component p^{aux} is introduced as a supplement which can be changed in real-time. Let us call p^{aux} the *auxiliary ZMP*.

We can use the auxiliary ZMP for the following purposes.

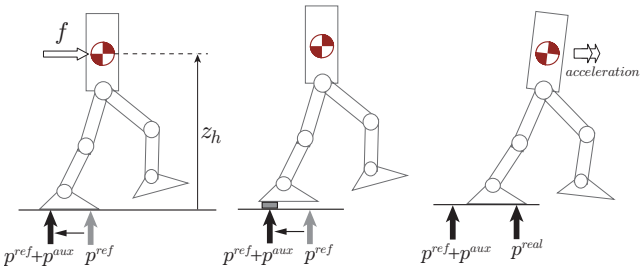
A. Absorbing external force

An external horizontal force f applied on the center of mass (CoM) is equivalent with the auxiliary ZMP of

$$p^{aux} = -f z_h/(Mg),$$

where M is the total mass of the robot, z_h is the height of CoM and g is gravity acceleration (Fig.2(a)). Therefore, a robot can absorb the effect of the external force by using this equation.

1-4244-0259-X/06/$20.00 ©2006 IEEE



(a) External force (b) Uneven floor (c) Posture control

Fig. 2. Applications of auxiliary ZMP

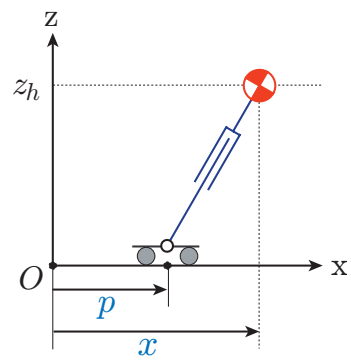


Fig. 3. Linear inverted pendulum on a moving cart

B. Adapting to ground unevenness

On an uneven terrain, it can happen that a robot cannot realize the reference ZMP p^{ref} because of the ground shape as shown in Fig.2(b). In such case, the pattern generator should move the ZMP into the actual support polygon by using the auxiliary ZMP.

C. Posture control

When the body of a robot gets inclined, we can use a *Model ZMP control* which was proposed by Hirai, Hirose, Haikawa and Takenaka to control the Honda humanoid robot P2 [9]. To recover the body posture, the robot strongly accelerates forward and it makes the expected ZMP move away from the real ZMP (p^{real}) [1]. The posture recovery torque is generated in proportional to the difference of the ZMPs(Fig.2(c)).

III. Dynamics of Pattern Generator

In this section, we review a pattern generator as a preparation of auxiliary ZMP. The pattern generator is derived by applying a preview control theory to an inverted pendulum model [12].

A. Preview controller based on linear inverted pendulum

Biped dynamics in sagittal plane can be approximated by the model of Fig.3 [13] whose dynamics is given by

$$\ddot{x} = \frac{g}{z_h}(x - p), \tag{2}$$

where x is the horizontal displacement of the center of mass (CoM), z_h is the height of CoM, g is gravity acceleration and p is the location of ZMP.

By assuming a cart carrying ZMP (ZMP-cart) instead of actual feet, we can concisely model the dynamics of a biped robot. For example, cart motion can represent the weight shifting in double support phase as well as the change of the foot torque in single support phase.

As the system input, let us take the speed of the ZMP-cart v. The dynamics of the ZMP-cart is given as

$$\dot{p} = v. \tag{3}$$

[1]Since $p^{ref} + p^{aux}$ is a nominal value planned by the pattern generator, it can leave the support polygon.

From eqs. (2) and (3), we have the following system representation.

$$\begin{aligned} \frac{d}{dt}\begin{bmatrix} x \\ \dot{x} \\ p \end{bmatrix} &= \begin{bmatrix} 0 & 1 & 0 \\ \frac{g}{z_h} & 0 & -\frac{g}{z_h} \\ 0 & 0 & 0 \end{bmatrix}\begin{bmatrix} x \\ \dot{x} \\ p \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} v \\ p &= \begin{bmatrix} 0 & 0 & 1 \end{bmatrix}\begin{bmatrix} x \\ \dot{x} \\ p \end{bmatrix} \end{aligned} \tag{4}$$

By discretizing eq.(4) with sampling time of Δt, we obtain the following system.

$$\begin{cases} \boldsymbol{x}_{k+1} = \boldsymbol{A}_0 \boldsymbol{x}_k + \boldsymbol{b}_0 v_k \\ p_k = \boldsymbol{c}_0 \boldsymbol{x}_k \end{cases} \tag{5}$$

where

$$\begin{aligned} \boldsymbol{x}_k &\equiv [\ x(k\Delta t) \quad \dot{x}(k\Delta t) \quad p(k\Delta t)\ ]^T, \\ v_k &\equiv v(k\Delta t), \\ p_k &\equiv p(k\Delta t). \end{aligned}$$

To obtain a preview controller, we define a performance index as

$$J = \sum_{j=1}^{\infty}\{Q(p_j^{ref} - p_j)^2 + Rv_j^2\}, \tag{6}$$

where Q and R are non-negative weights. This index J can be minimized by using the following controller [14].

$$v_k = -\boldsymbol{K}\boldsymbol{x}_k + [f_1, f_2, \cdots f_N]\begin{bmatrix} p_{k+1}^{ref} \\ \vdots \\ p_{k+N}^{ref} \end{bmatrix} \tag{7}$$

where $\boldsymbol{K}$ and f_j are calculated as

$$\begin{aligned} \boldsymbol{K} &\equiv (R + \boldsymbol{b}_0^T \boldsymbol{P} \boldsymbol{b}_0)^{-1} \boldsymbol{b}_0^T \boldsymbol{P} \boldsymbol{A}_0 \\ f_i &\equiv (R + \boldsymbol{b}_0^T \boldsymbol{P} \boldsymbol{b}_0)^{-1} \boldsymbol{b}_0^T (\boldsymbol{A}_0 - \boldsymbol{b}_0 \boldsymbol{K})^{T*(i-1)} \boldsymbol{c}_0^T Q. \end{aligned} \tag{8}$$

$\boldsymbol{P}$ is a solution of the following Riccati equation.

$$\boldsymbol{P} = \boldsymbol{A}_0^T \boldsymbol{P} \boldsymbol{A}_0 + \boldsymbol{c}_0^T Q \boldsymbol{c}_0 - \boldsymbol{A}_0^T \boldsymbol{P} \boldsymbol{b}_0 (R + \boldsymbol{b}_0^T \boldsymbol{P} \boldsymbol{b}_0)^{-1} \boldsymbol{b}_0^T \boldsymbol{P} \boldsymbol{A}_0 \tag{9}$$

The parameters we used in this paper are shown in the following table.

g	9.8 [m/s^2]	Q	1
z_h	0.8 [m]	R	1×10^{-10}
Δt	0.005 [s]	N	320
		$N\Delta t$	1.6 [s]

B. Structure of the pattern generator

Figure 4 illustrates the structure of the pattern generator.

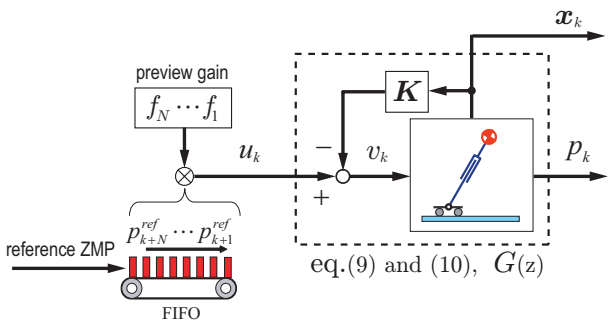


Fig. 4. Structure of the pattern generator

For further analysis we rewrite the pattern generator. First, by combining the inverted pendulum system with the state feedback gain we have

$$\boldsymbol{x}_{k+1} = \boldsymbol{A}\boldsymbol{x}_k + \boldsymbol{b}u_k, \tag{10}$$

$$p_k = \boldsymbol{c}\boldsymbol{x}_k, \tag{11}$$

where $\boldsymbol{A} \equiv \boldsymbol{A}_0 - \boldsymbol{b}_0\boldsymbol{K}, \boldsymbol{b} \equiv \boldsymbol{b}_0, \boldsymbol{c} \equiv \boldsymbol{c}_0$.

The system input u_k is calculated by the reference ZMP pattern p^{ref} and the preview gain.

$$u_k = \sum_{j=1}^{N} f_j p^{ref}_{k+j} \tag{12}$$

This convolution process is illustrated by a belt conveyer in Fig.4. Finally, by feeding u_k to the system of eq.(10) and (11), we obtain a proper trajectory of the CoM $\boldsymbol{x}$ and the resulted ZMP p.

Figure 5 shows an example pattern generation. First, we determine a reference ZMP for the desired walking. Fig.5(a) is the reference ZMP corresponding a walk of two 30[cm] steps with 0.7[s] single support and 0.1[s] double support for each step. Next, by a convolution between the reference ZMP and the preview gain $[f_1, \ldots, f_N]$ (Fig.5(b)) the internal signal u is calculated (Fig.5(c)). Finally, the internal signal is transformed into the CoM and the ZMP (Fig.5(d)).

IV. ZMP MODIFICATION USING AN INVERSE SYSTEM

A. Inverse system

The pattern generator's output p is used to control the actual walking robot since it gives a ZMP consistent with the robot dynamics of eq.(2). The pattern generator guarantees its output to be

$$p_k = p^{ref}_k + \epsilon \tag{13}$$

where ϵ is the tracking error of the preview controller whose magnitude can be negligibly small by selecting the weights of eq.(6).

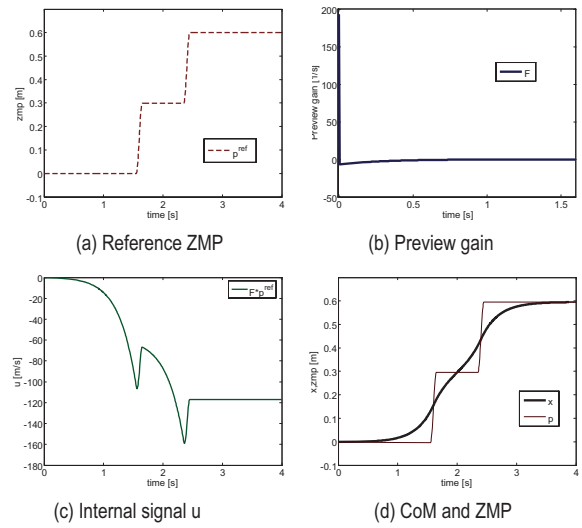


Fig. 5. Process of pattern generation

Our goal is to modify the ZMP from the reference trajectory as

$$p_k = p^{ref}_k + p^{aux}_k + \epsilon \tag{14}$$

where p^{aux}_k is the auxiliary ZMP specified in real-time.

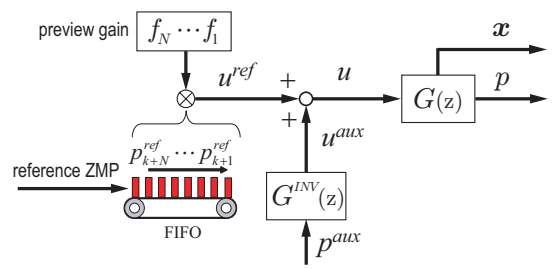


Fig. 6. Preview pattern generator with inverse system

To realize the ZMP modification of eq.(14), we extend the pattern generator as Fig.6. Newly added $G^{INV}(z)$ must have the following property to realize the real-time ZMP modification of eq.(14).

$$G(z) \cdot G^{INV}(z) = 1 \tag{15}$$

where $G(z)$ is a transfer function from internal signal u to the ZMP p determined by eq.(10) and (11). Using the parameters shown in Section III.A, $G(z)$ is given as

$$G(z) = \frac{0.005(z-1.018)(z-0.9827)}{(z-0.9827)^2(z-0.8539)}. \tag{16}$$

From eq.(15) we get

$$G^{INV}(z) = \frac{(z-0.9827)^2(z-0.8539)}{0.005(z-1.018)(z-0.9827)}. \tag{17}$$

This transfer function can not be realized since its numerator has higher order than its denominator. This problem can be

solved by multiplying an operator of one step delay z^{-1}, and we get a realizable transfer function as

$$G^{INV}(z) = \frac{(z-0.9827)^2(z-0.8539)}{0.005z(z-1.018)(z-0.9827)}. \tag{18}$$

The drawback of this $G^{INV}(z)$ is that we have one sample time delay from p^{aux} to p. Because now we have,

$$G(z) \cdot G^{INV}(z) = z^{-1}. \tag{19}$$

As shown here, we can obtain an inverse transfer function in discrete time domain by accepting delay. See Sain et al. [15] for more rigorous discussion.

B. Inverse system in state space

Let us derive the inverse system $G^{INV}(z)$ in state space. By letting eq.(10) one sampling step forward, we get

$$\begin{aligned} p_{k+1} &= \boldsymbol{c}\boldsymbol{x}_{k+1} \\ &= \boldsymbol{c}\boldsymbol{A}\boldsymbol{x}_k + \boldsymbol{c}\boldsymbol{b}u_k. \end{aligned}$$

Since $\boldsymbol{cb} \neq 0$, we can derive the output equation of the inverse system, which gives u_k from $\boldsymbol{x}_k$ and p_{k+1}.

$$\begin{aligned} u_k &= -(\boldsymbol{cb})^{-1}\boldsymbol{c}\boldsymbol{A}\boldsymbol{x}_k + (\boldsymbol{cb})^{-1}p_{k+1} \\ &\equiv \boldsymbol{c}_1\boldsymbol{x}_k + d_1 p_{k+1} \end{aligned} \tag{20}$$

The state transition of the inverse system is obtained by substituting eq.(20) into eq.(10).

$$\boldsymbol{x}_{k+1} = (\boldsymbol{A} + \boldsymbol{b}\boldsymbol{c}_1)\boldsymbol{x}_k + \boldsymbol{b}d_1 p_{k+1}. \tag{21}$$

From above results, we can form the state space representation of $G^{INV}(z)$ as the following equations.

$$\begin{cases} \boldsymbol{x}_{k+1}^{INV} = \boldsymbol{A}_1\boldsymbol{x}_k^{INV} + \boldsymbol{b}_1 p_k^{aux} \\ u_k^{aux} = \boldsymbol{c}_1\boldsymbol{x}_k^{INV} + \boldsymbol{d}_1 p_k^{aux} \end{cases} \tag{22}$$

where

$$\begin{aligned} \boldsymbol{A}_1 &\equiv \boldsymbol{A} + \boldsymbol{b}\boldsymbol{c}_1, \\ \boldsymbol{b}_1 &\equiv \boldsymbol{b}d_1. \end{aligned}$$

$\boldsymbol{x}_k^{INV}$ gives the state of the inverse system. In the following simulations, $G^{INV}(z)$ is implemented by eq.(22).

C. Nature of the inverse system

Figure 7 shows an example of applying an auxiliary ZMP. From 1.0[s] to 1.3[s], an auxiliary ZMP of -0.05[m] was added to the walking pattern of Fig.5. Although the ZMP is correctly modified (upper graph, thin line), the internal signal $u^{ref} + u^{aux}$ (lower graph, gray line) and the CoM trajectory (upper graph, bold line) quickly diverges. It happens since the inverse system $G^{INV}(z)$ is unstable.

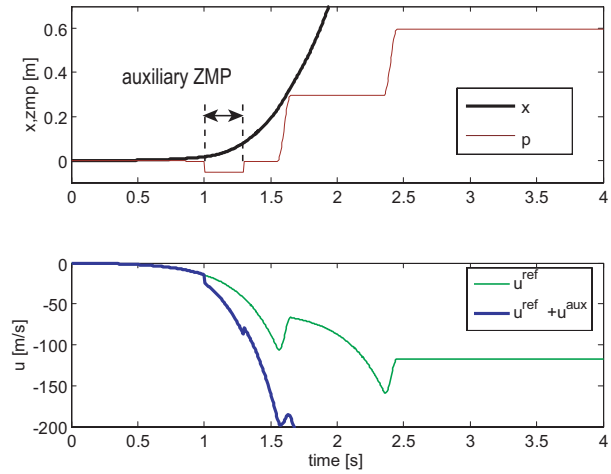


Fig. 7. Adding auxiliary ZMP using an inverse system

D. Stabilized inverse system

To avoid the divergence of the inverse system, we can stabilize it by a feedback. The auxiliary ZMP input is modified by

$$p^{aux2} = p^{aux} + \boldsymbol{K}^{INV}\boldsymbol{x}^{INV}, \tag{23}$$

where $\boldsymbol{K}^{INV}$ is a feedback gain matrix which stabilizes eq.(22) and we determined it by a pole assignment. p^{aux2} becomes the input of the inverse system. Fig.8 shows the block diagram.

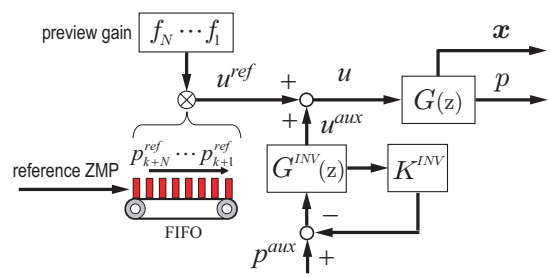


Fig. 8. Preview pattern generator with a stabilized inverse system

Fig. 9(a) shows the result when the same auxiliary ZMP ($p^{aux} = -0.05$, $t \in [1.0, 1.3]$) is applied to this pattern generator. Thanks to the state feedback, the CoM trajectory does not diverge but the auxiliary ZMP was not correctly realized. The distortion becomes smaller when the auxiliary ZMP has high frequency component. Fig.9(b) shows the result when an auxiliary ZMP of $p^{aux} = 0.05\sin(50t)$, $t \in [1.0, 1.3]$ was applied. We can observe the ZMP is well modified as expected.

V. Adjusting Future ZMP

A. Reset of the inverse system

To avoid the divergence of the inverse system $G^{INV}(z)$, let us *reset* the system instead of the stabilization. We can reset $G^{INV}(z)$ by letting

$$\boldsymbol{x}_k^{INV} = \boldsymbol{0}. \tag{24}$$

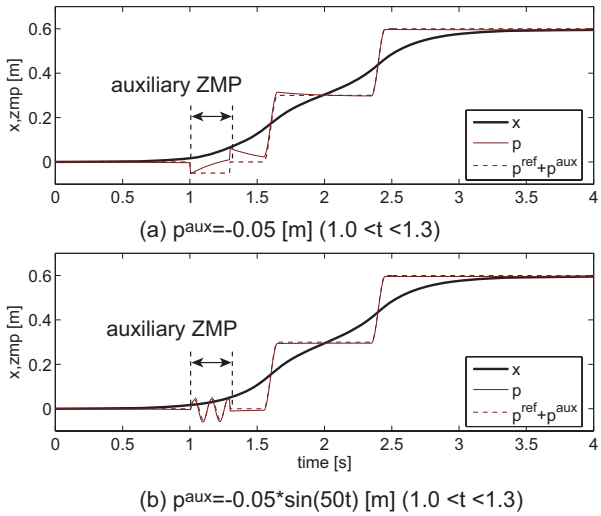


Fig. 9. Adding auxiliary ZMP using a stabilized inverse system

In Figure 10, the auxiliary ZMP is given without stabilization, but the inverse system is reset at the end of the ZMP modification (1.3[s]).

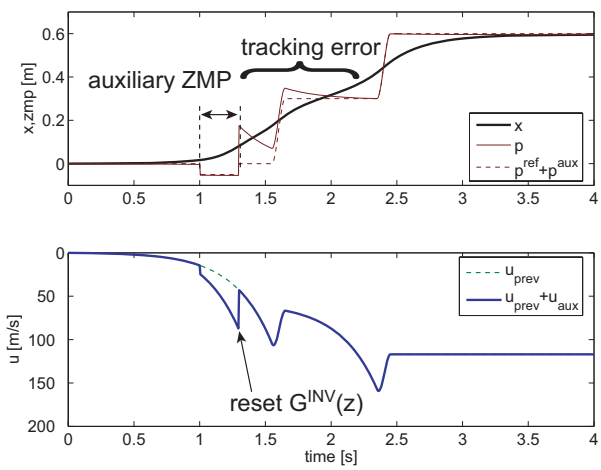


Fig. 10. Reseting an inverse system to avoid divergence

While the divergence of the CoM motion was avoided, we observe a huge tracking error of the ZMP in the upper graph of Fig.10. The tracking error is inappropriate for walking control and it appears at 1.3[s] when u discontinuously changed by the reset of $G^{INV}(z)$ (lower graph of Fig.10). From this observation, let us assume the following working hypothesis.

> *Working hypothesis:* By keeping continuity of the internal signal u, the preview pattern generator can generate a pattern with small ZMP tracking error.

Although we do not have a solid theoretical basis for this hypothesis, it is confirmed by simulations in the rest of this paper. To keep the continuity of the internal signal u, we modify the reference ZMP by time shifting.

B. Time shift of the reference ZMP

Suppose the reference ZMP trajectory is defined as a function of virtual time τ. The preview feedforward term of eq.(12) can be rewritten as

$$g_p(\tau) \equiv \sum_{j=1}^{N} f_j p^{ref}(\tau + j\Delta t). \tag{25}$$

The normal preview controller is given by specifying the virtual time for each sampling τ_k as

$$\tau_{k+1} = \tau_k + \Delta t. \tag{26}$$

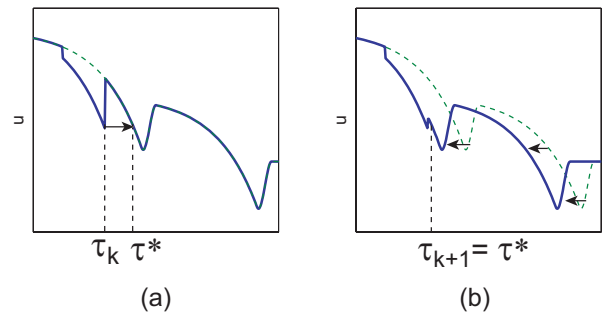


Fig. 11. Absorbing discontinuous change of the internal signal u. (a)Find τ^* to keep current u. (b)Update virtual time as $\tau_{k+1} = \tau^*$. This results the time shift of the reference ZMP.

When we reset the inverse system at time $(k+1)\Delta t$, we have discontinuous change of the internal signal as

$$u_k^{aux} \neq 0 \qquad and \qquad u_{k+1}^{aux} = 0. \tag{27}$$

To ease this abrupt change, let us introduce an active control of virtual time τ as shown in Fig.11. This time shifting procedure is formalized as following.

$$\tau_{k+1} = \tau^*, \text{ such that } g_p(\tau^*) = u_k^{ref} + u_k^{aux}. \tag{28}$$

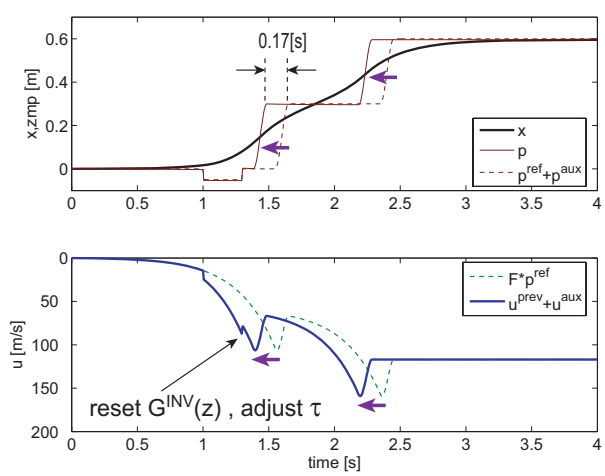


Fig. 12. Modified ZMP by shifting virtual time

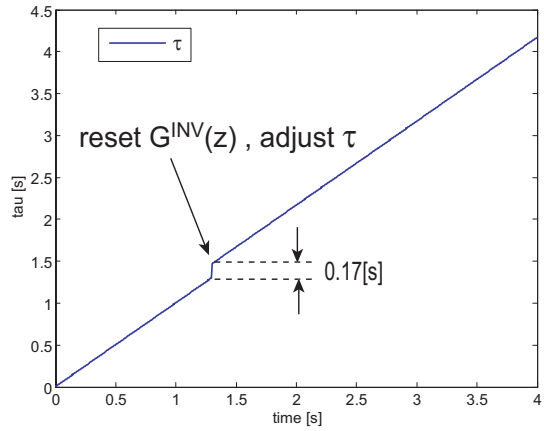


Fig. 13. Virtual time τ

Figure 12 shows the effect of such adjustment. At the moment of reseting the inverse system (1.3[s]), τ is adjusted by eq.(28). As the result the reference ZMP is quickened and the walking pattern finishes 0.17[s] earlier than planned. Though the steps were quickened, their lengths were kept as planned. We can see how virtual time was modified in Fig. 13.

VI. Walking simulation on uneven floor

By using auxiliary ZMP we designed a simple controller to stabilize walking on uneven floor.

$$p^{aux} = k_1\theta + k_2\dot{\theta} + k_3(p^{ref} - p), \tag{29}$$

where θ is the body pitch angle and $\dot{\theta}$ is the body pitch angular velocity. p^{ref} is the reference ZMP and p is the actual ZMP i.e. the center of pressure generated by the contact between the feet and the ground. k_1, k_2, k_3 are feedback gains. eq.(29) is used for the sagittal stabilization and the same controller is used for the lateral stabilization.

To evaluate the proposed pattern generator with the control algorithm of eq.(29), we performed dynamic simulations using OpenHRP [16] with hardware model of HRP-2 which is a 30 DOF humanoid robot with 1.54[m] height and 58[kg] weight [17]. We used an uneven floor made of flat plates whose heights randomly changes within ± 30 [mm]. Both the length of plates and the step length of the walking pattern were set to be 30 [cm]. Therefore, the robot only suffers touchdown height error at each step. More difficult problems on uneven terrain, for example, inclination of the support foot, unstable contact etc. were avoided.

To realize the auxiliary ZMP, we used the stabilized inverse system of Fig.8. We also used τ adjust algorithm of eq.(28) when the output of the inverse system u^{aux} reached predetermined threshold. Figure 14 shows the simulated HRP-2 walking over the step of 26 [mm]. The unexpected floor height created the posture error and the ZMP error, which were translated into auxiliary ZMP by eq.(29).

Fig. 15 shows the time profile of the CoM and the ZMP on the uneven floor. Thin lines show ZMP profiles which change irregularly due to the ground unevenness and the auxiliary

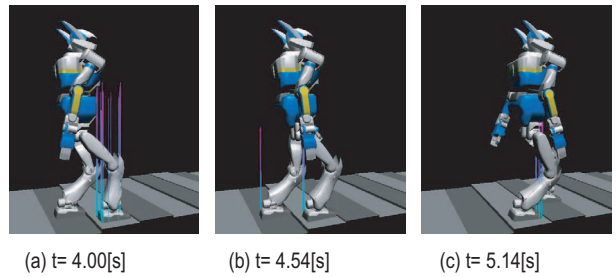


Fig. 14. Walking on uneven floor with τ adjust

ZMP. The vertical dotted lines indicate the change of support phase. We can see the walking cycle is not constant as the result of time shifting. Fig. 16 shows how the virtual time τ was modified in the simulation.

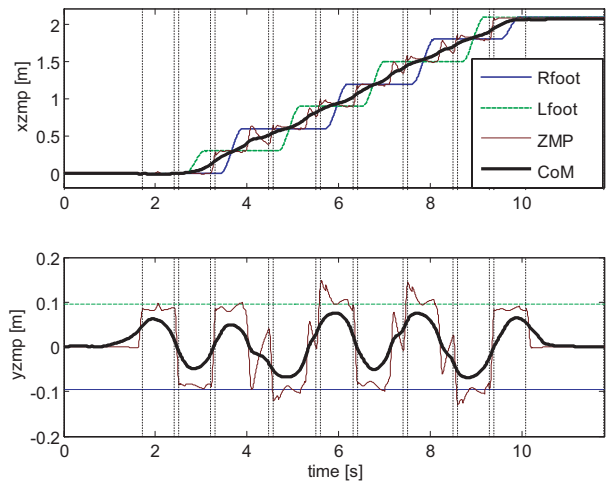


Fig. 15. Time profile of CoM and ZMP generated on uneven floor

On the other hand, the ground projected CoM and ZMP shown in Fig. 15 illustrate that the step length was kept constant since the walking pattern was modified only in the time domain.

To see the effectiveness of the virtual time adjustment, we also simulated a walk on the uneven floor without using eq.(28). In this simulation, the robot had lost balance and fell down at the second step, as shown in Fig. 18.

VII. Conclusion

In this paper, we discussed a method to control auxiliary ZMP by expanding a pattern generator based on ZMP preview control. We can generate auxiliary ZMP by using an inverse system and its effect can be canceled by shifting virtual time of the reference ZMP. In other words, a biped robot must change its walking pattern to realize the ZMP manipulation. As an application of auxiliary ZMP, a walking on uneven floor was simulated. In the simulation a humanoid robot could walk on uneven terrain by using auxiliary ZMP, and the walking speed was changed adaptively as its side effect.

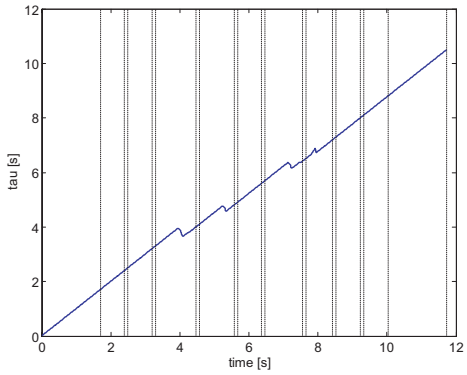


Fig. 16. Virtual time τ

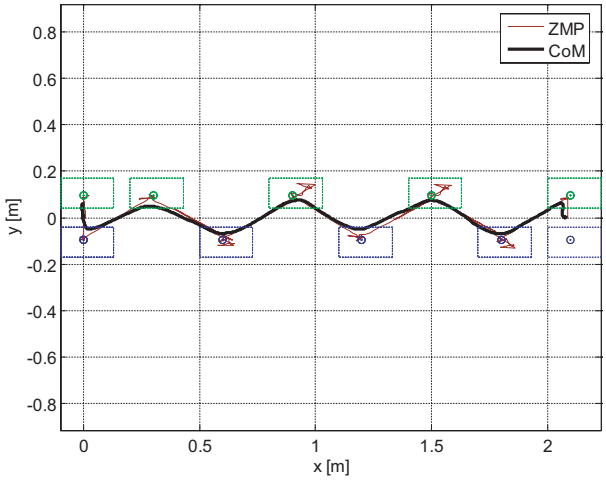
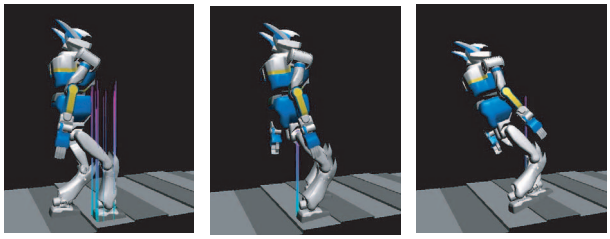


Fig. 17. Ground projection of CoM and ZMP generated on uneven floor

The auxiliary ZMP can also be used to handle external disturbances. We will examine such possibility as well as implementing the algorithm on the actual humanoid robot.



(a) t= 4.00[s] (b) t= 4.54[s] (c) t= 5.14[s]

Fig. 18. Walking on uneven floor without τ adjust

References

[1] Vukobratović, M. and Stepanenko, J., "On the Stability of Anthropomorphic Systems," Mathematical Biosciences, Vol.15 pp.1-37, 1972.

[2] Vukobratović, M. and Borovac, B., "Zero-Moment Point — Thirty Five Years of Its Life," International Journal of Humanoid Robotics, Vol.1, No. 1, pp.157-173, 2004.

[3] McGeer, T., "Passive Dynamic Walking," Intr. J. of Robotics Research, Vol.9, No.2, pp.62-82, 1990.

[4] Collins, S., Ruina, A., Tedrake, R. and Wisse, M., "Efficient Bipedal Robots Based on Passive-Dynamic Walkers," Science, Vol.307, pp.1082-1085, 2005.

[5] Pratt, J., Dilworth, P. and Pratt, G., "Virtual Model Control of a Bipedal Walking Robot," Proc. of the 1997 ICRA, pp.193-198, 1997.

[6] Tsuchiya, K., Aoi, S., Tsujita, K., "Locomotion Control of Biped Locomotion Robot using Nonlinear Oscillators," Proc. of ICRA2003, pp.1745-1750, 2003.

[7] Westervelt, E.R., Grizzle, J.W. and Koditschek, D.E., "Hybrid Zero Dynamics of Planar Biped Walkers," IEEE Trans. on Automatic Control, Vol.48, No.1, pp.42-56, 2003.

[8] Takanishi, A., Lim, H.,Tsuda, M. and Kato, I., "Realization of Dynamic Biped Walking Stabilized by Trunk Motion on a Sagittally Uneven Surface," Proceedings of IEEE International Workshop on Intelligent Robots and Systems (IROS '90), pp.323-330, 1990.

[9] Hirai, K., Hirose, M., Haikawa, Y. and Takenaka, T., "The Development of Honda Humanoid Robot," Proc. of the 1998 ICRA, pp.1321–1326, 1998.

[10] Harada, K., Kajita, S., Kanehiro, F., Fujiwara, K., Kaneko, K., Yokoi, K. and Hirukawa, H., "Real-Time Planning of Humanoid Robot's Gait for Force Controlled Manipulation," Proc. of ICRA2004, pp.616-622, 2004.

[11] Wieber, P.B. and Chevallereau, C., "Online adaptation of reference trajectories for the control of walking systems," http://www.inrialpes.fr/bipop/pub/publications/wieber.html

[12] Kajita, S., Kanehiro, F., Kaneko, K., Fujiwara, K., Harada, K., Yokoi, K. and Hirukawa, H., "Biped Walking Pattern Generation by using Preview Control of Zero-Moment Point," Proc. of ICRA2003, pp.1620-1626, 2003.

[13] Sugihara, T., Nakamura, Y. and Inoue, H., "Realtime Humanoid Motion Generation through ZMP Manipulation based on Inverted Pendulum Control," Proc. of the 2002 ICRA, pp.1404-1409, 2002.

[14] Katayama, T., Ohki, T., Inoue, T. and Kato, T., "Design of an Optimal Controller for a Discrete Time System Subject to Previewable Demand," Int. J. Control, Vol.41, No.3, pp.677-699, 1985.

[15] Sain, M.K. et al., "Invertibility of Linear Time-Invariant Dynamical Systems," IEEE Trans. on Automatic Control, Vol.14, No.2, pp.141-149, 1969.

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

[17] Kaneko, K., Kanehiro, F., Kajita, S., Hirukawa, H., Kawasaki, T., Hirata, M., Akachi, K. and Isozumi, T., "Humanoid Robot HRP-2," Proc. of ICRA2004, pp.1083-1090, 2004.