Real-time Jumping Trajectory Generation for a One Legged Jumping Robot

Barkan Ugurlu and Atsuo Kawamura
Department of Electrical and Computer Engineering
Yokohama National University
79-5 Tokiwadai, Hodogaya-ku, Yokohama, 240-8501, JAPAN
E-mail: {barkanu, kawamura}@kawalab.dnj.ynu.ac.jp
Tel : +81-45-339-4162, Fax: +81-45-338-1157

***Abstract*— As a recent trend in humanoid robot research, running motion appears to be a very important issue since faster and faster mobility is required. Being an essential fact, jumping phenomenon seems to be the previous step of succeeding running motion as jumping dynamics play the key role to achieve stable running motion cycles. Therefore, in this paper, authors group sought a method to generate real-time jumping patterns which may ensure the overall stability in every possible phases through a complete jumping cycle. To be able to reach this goal, we discretized ZMP equation in polar coordinates which enabled us to include angular momentum information without referring any predetermined function or set of values. Subsequently, vertical and forward one leg jumping simulations are run on a 3-D dynamic simulator applying the aforementioned technique. Simulation results indicate that stable and repetitive jumping cycles are obtained both for vertical jumping and forward jumping cases. Finally, vertical jumping experiments are performed within the result of successful jumping cycles.**

I. Introduction

In a dynamic human environment, people may change their movement from walking to running in various situations. Correspondingly, a well adapted humanoid robot should function running if necessary. In biped locomotion, the motion without double support phase is called running motion. In other words, running motion includes flight phases as a result of repetitive jumping cycles. Therefore, jumping motion appears to be an essential feature in humanoid robot technology. Moreover, humanoid robots may have to perform small jumps in order to stabilize their locomotion when they are under the effect of external forces. Considering these facts, jumping motion merits careful analysis and will give us insights in order to improve current humanoid robots.

To be able to improve legged robot technology, Raibert and his colleagues have revealed fundamental control laws of running motion[4]. Their impressive robots were powered by hydraulic actuators and they demonstrated various stable running motions. Using similar approach in control laws and considering energy efficiency, Ahmadi and Buehler achieved monopod running with a spring supported mechanism[7].

Although running and jumping are required functions for current humanoids, they are mainly excepted to walk with a normal speed and interact with human environment. For this purpose, robots with hydraulic/pneumatic actuators and spring supported mechanisms are not the best the match. Hence, in order to realize both fast motions and usual movements, many researchers developed various jumping pattern generation methods for robots which can also perform usual humanoid activities[1][2][11].

Kajita et al. suggested an offline pattern generation technique based on the total momentum resolution, called RMC (Resolved Momentum Control)[3]. Authors group also simulated RMC based jumping pattern generation[13]. Despite the fact that RMC covers dominant robot dynamics; it is relatively difficult to manipulate robot's control strategy in real-time. Therefore, our studies are mainly aimed at real-time jumping pattern generation in which neither special hardware structures are required nor restriction on mass balance is necessary.

Considering the facts mentioned above, we composed a method to generate online jumping patterns which may ensure the overall stability through a complete jumping sequence. In our proposed method, ZMP equation is discretized in polar coordinates, so that, angular momentum information is smoothly included without referring any predetermined function or set of values. In the present paper, our one legged jumping robot KEN-2 and its 3-D simulation model are introduced in Section II. Our Proposed real-time jumping pattern generation method is explained in Section III. Simulation and experimental results are discussed in Section IV. Finally, the paper is concluded in Section V.

II. One Legged Jumping Robot KEN-2

In order to explore jumping dynamics, we developed a one legged experimental jumping robot, called KEN-2. Its mechanical and electronic hardware structure is very similar to the actual humanoid biped robot MARI-3[10], therefore, acquired knowledge and experience can directly be adapted to humanoid running research.

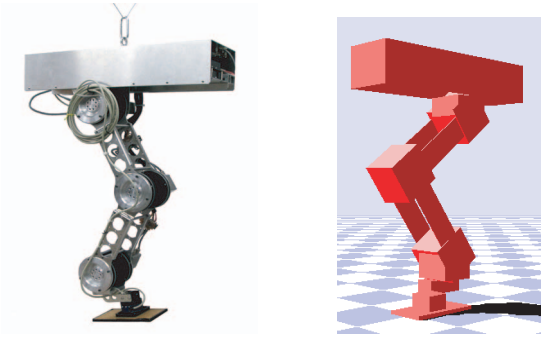
Table I summarizes the mechanical specifications of KEN-2. It has 3 DOF (degrees of freedom) at ankle, knee and hip about pitch axis. Hence its motion is constrained on sagittal plane. It might be noted that, torques about pitch axes are generating forward running motion in biped robots as well. Moreover, KEN-2 is equipped with a 6 axis Force/Torque sensor at the wrist and a gyro sensor at the upper body. Main controller is a 32 bit built-in Renesas SH-4 microcontroller. Data communication is performed via RON (RObot Network)

978-1-4244-1766-7/08/$25.00 ©2008 IEEE

TABLE I

MECHANICAL SPECIFICATIONS OF KEN-2

Size	Upper Leg lenght: 300 [mm] Lower Leg lenght: 300 [mm] Ankle-sole lenght: 170 [mm] Toe-heel lenght: 200 [mm]
Weight	Leg: 7.4523 [kg] Body: 6.1306 [kg] Total : 13.5829 [kg]



(a) Actual Robot (b) ROCOS Model

Fig. 1. One Legged Jumping Robot KEN-2

bus which is a serial communication bus system, developed in our laboratory. Fig. 1 shows the actual robot and its ROCOS[9] (RObot COntrol Simulator) simulation model. ROCOS is a 3-D dynamic simulator, also developed in our laboratory.

III. REAL-TIME JUMPING PATTERN GENERATION

Here, the main idea of generating jumping pattern is to obtain real time joint motions that ensure desired ZMP profile through the support phase and ensure predetermined foot trajectory through the flight phase, consecutively. In Fig. 2, significant moments from a jumping sequence are illustrated.

In the support phase, the robot is considered as a point mass and a massless telescopic leg, contacting to the floor with a foot as shown in Fig. 3a. In this model, main parameters are CoM lenght, r, and angle about vertical z-axis, θ. Deriving CoM lenght by a 5^{th} order polynomial and using ZMP reference, θ angle's trajectory is calculated by solving ZMP equation in discrete domain.

In the flight phase, our robot's upper body is considered as a pseudo base as shown in Fig. 6. Wherat foot trajectory is calculated based on a desired flight angle, ϕ, using 5^{th} order polynomials. ϕ is the angle about vertical z-axis when the robot is on air.

A. CoM Lenght Planning

Determining a proper timing profile, CoM lenght is derived from a two segmental 5^{th} order polynomials by setting inital and final values of position, velocity and acceleration of CoM. Notwithstanding, this method is highly depending on determination of initial values. In Fig. 4, a possible problem

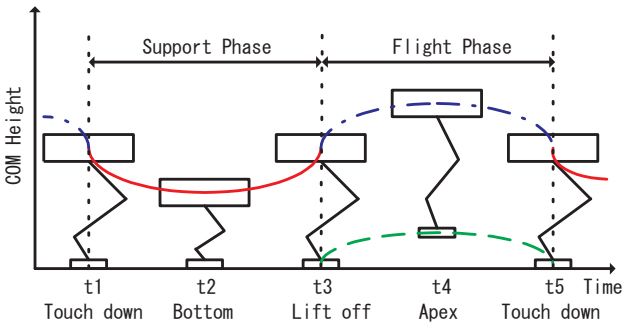


Fig. 2. A Complete Jumping Sequence

is illustrated which may occur just after a landing state. In this case, if the initial values of CoM cannot be clarified, polynomial approach might not derive a successful trajectory. Equating the total energy between the states of lift off and touch down within the exact measurement of flight time(T_f); it is possible to calculate landing position and velocity as below,

$$\begin{aligned} \dot{x}_{td} &= \dot{x}_{lo} \\ \dot{r}_{td} &= \sqrt{(gT_f - \dot{z}_{lo})^2 + \dot{x}_{td}^2} \\ x_{td} &= tan(\phi_{td}) \left(z_{lo} - \left(\dot{z}_{lo} T_f - \tfrac{1}{2} g T_f^2 \right) \right) \\ r_{td} &= \sqrt{\left(z_{lo} - \left(\dot{z}_{lo} T_f - \tfrac{1}{2} g T_f^2 \right) \right)^2 + x_{td}^2} \end{aligned} \quad (1)$$

where subscripts lo refers lift off, td refers touch down states. g symbolizes gravitational acceleration. ϕ_{td} is the angle about vertical z-axis at the moment of touch down which is planned in the flight phase.

B. Angular Trajectory Generation

Here, the main goal is to calculate θ angle's trajectory by using ZMP stability condition. In Fig. 3b, cart-curved table model is illutstrated. Since z-axis CoM position is varied, our table is curved. In this section, we are going to explain how to derive θ angular trajectory by manipulating ZMP while CoM lenght is derived by using polynomials.

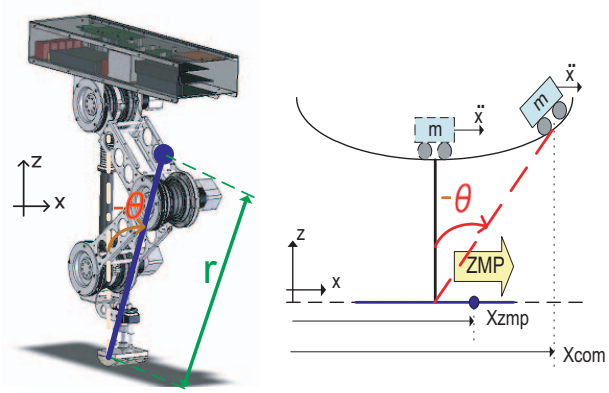


Fig. 3. a)One Mass Model[left], b)Cart-Curved Table Model[right]

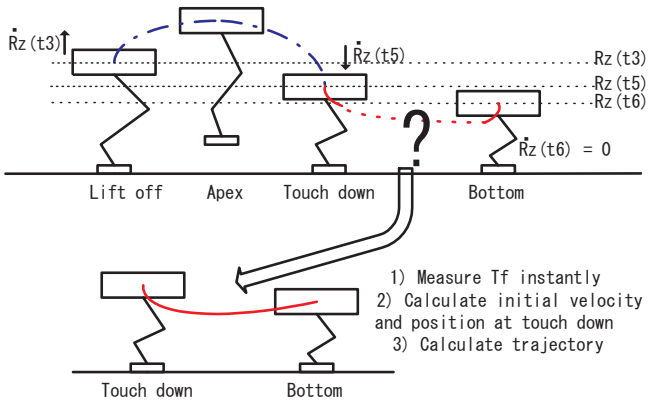


Fig. 4. Uncertainty at a touch down state. R_z shows the vertical position.

1) Motion Constraint on KEN-2: As mentioned previously, KEN-2 has 3 actuators which are able to generate torques only about pitch axis because of the robot's mechanical structure. Positively, it makes its reference pitch axis angular momentum expression independent from other planes.

If the friction between the robot's foot sole and the floor is sufficient, we can consider the support phase motion as a rotation of a rigid body about a fixed point. For this case, Euler's equations of motion might give us insights,

$$\begin{aligned} I_{xx}\dot{\omega}_x - (I_{yy} - I_{zz})\omega_y\omega_z &= \tau_x \\ I_{yy}\dot{\omega}_y - (I_{zz} - I_{xx})\omega_z\omega_x &= \tau_y \\ I_{zz}\dot{\omega}_z - (I_{xx} - I_{yy})\omega_x\omega_y &= \tau_z \end{aligned} \tag{2}$$

in which I_{xx}, I_{yy} and I_{zz} are moments of inertia about principle axes, ω_x, ω_y and ω_z are angular velocities about principle axes and τ_x, τ_y and τ_z are rate change of angular momentums(torques) about roll pitch and yaw axes. Because the motion is constrained on sagittal plane, τ_x, τ_z, ω_x and ω_z are set to zero. In this case, first derivative of pitch axis angular momentum, $\dot{L}_y$, can be expressed as below[5]:

$$\begin{aligned} \omega_x = \omega_z &= 0 \\ \tau_x = \tau_z &= 0 \\ \tau_y = \dot{L}_y = I_{yy}\dot{\omega}_y &= I_{yy}\ddot{\theta}_y \end{aligned} \tag{3}$$

2) Discretization of ZMP Equation: In [1], ZMP equation of x-axis is expressed as below,

$$\dot{L}_y = mx(\ddot{z} + g) - mX_{zmp}(\ddot{z} + g) - mz\ddot{x} \tag{4}$$

where m is the total mass and two dots indicates accelerations. As shown in Fig. 3a, θ angle is clockwise. Hence, CoM position and acceleration can be expressed as follows.

$$\begin{aligned} x &= -rsin(\theta) \\ z &= rcos(\theta) \\ \ddot{x} &= -\ddot{r}sin(\theta) - 2\dot{r}\dot{\theta}cos(\theta) - r\ddot{\theta}cos(\theta) + r\dot{\theta}^2sin(\theta) \\ \ddot{z} &= \ddot{r}cos(\theta) - 2\dot{r}\dot{\theta}sin(\theta) - r\ddot{\theta}sin(\theta) - r\dot{\theta}^2cos(\theta) \end{aligned} \tag{5}$$

Placing (3) and (5) in (4) and simplifying the resultant equation we obtain;

$$\tau_{ZMP} = (mr^2 - I_{yy})\ddot{\theta} + 2mr\dot{r}\dot{\theta} - mgrsin(\theta) \tag{6}$$

in which $\tau_{ZMP} = mX_{zmp}(\ddot{z} + g)$, indicating the torque generated by the foot when ZMP is not coincided with the foot sole center. Please note that, angular momentum is not referenced as it is directly included in (6). In order to obtain θ angular trajectory, (6) is discretized with the sampling time Δt step by step in following equations.

$$\dot{\theta}[k] = \frac{\theta[k+1] - \theta[k-1]}{2\Delta t} \tag{7}$$

$$\ddot{\theta}[k] = \frac{\theta[k+1] - 2\theta[k] + \theta[k-1]}{\Delta t^2} \tag{8}$$

$$P[k] = \frac{mr[k]^2 - I_{yy}[k]}{\Delta t^2} \tag{9}$$

$$R[k] = \frac{mr\dot{r}}{\Delta t} \tag{10}$$

Atop, (7) and (8) are displaying Euler's discretization technique. (9) and (10) are used to ease our calculations. Combining (7), (8), (9) and (10) in (6) and adding one delay, θ angular trajectory can be expressed as below.

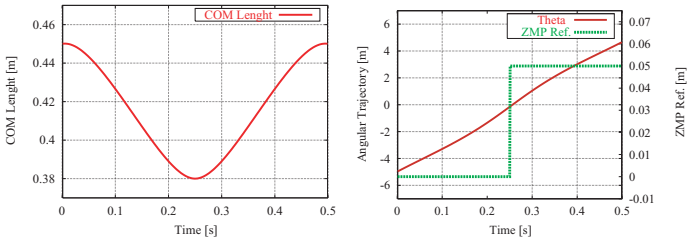
$$\begin{aligned} \theta[k] &= \frac{\tau_{ZMP}[k-1] + 2\theta[k-1]P[k-1]}{P[k-1] + R[k-1]} \\ &+ \frac{\theta[k-2](P[k-1] - R[k-1])}{P[k-1] + R[k-1]} \\ &- \frac{mgr[k-1]sin(\theta[k-1])}{P[k-1] + R[k-1]} \end{aligned} \tag{11}$$

As it may be understood in (11), θ angular trajectory can be calculated by ZMP command input, predetermined CoM lenght and previous values. Fig. 3b illustrates this situation. A sample support phase trajectory is shown in Fig. 5 for a given set of mentioned parameters. Having calculated this angular trajectory, one may easily convert θ and r into z and x components of CoM trajectory using first two expressions in (5).

C. Foot Trajectory

In the flight phase, a well designed foot trajectory ensures a safe landing and stable initial conditions for the next support phase. Moreover, adjusting z-axis of foot trajectory may extend flight time as well. Fig. 6 illustrates a flight phase. Here, z-axis foot trajectory and ϕ angle is derived by 5^{th} order polynomials. Especially, adjusting ϕ angle at the touch down state can accelerate or decelerate the robot. Using these parameters, x-axis foot trajectory is simply calculated as below,

$$x_{foot} = x_{body} - (z_{body} - z_{foot})\,tan(\phi) \tag{12}$$



(a) CoM Lenght (b) Angular Traj. and ZMP Ref.

Fig. 5. A Sample Angular Traj. for given CoM Lenght and ZMP References

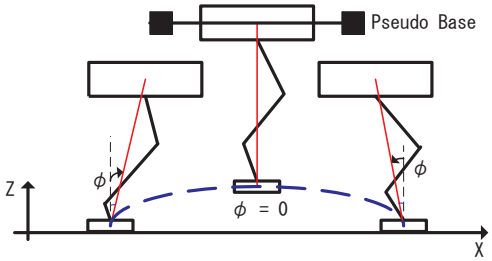


Fig. 6. Foot Trajectory Through a Flight Phase

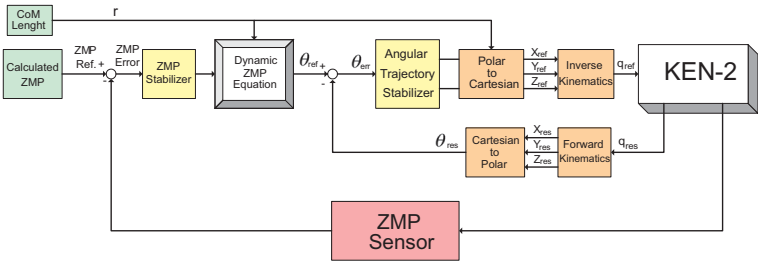


Fig. 7. General Control Frame in the Support Phase

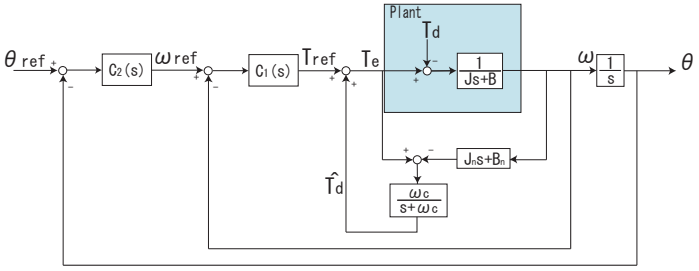


Fig. 8. Disturbance Observer Based Servo Control

where subscripts $foot$ and $body$ refer foot sole and base link respectively. Thanks to the well designed body altitude control, base link is considered as a pseudo base point and the robot becomes a 3 DOF manipulator through the flight phase.

D. Control Strategy

General control strategy is based on determining each joint motion by employing aforementioned methods. In the support phase, ZMP reference is chosen as the command input in order to ensure stability. Fig. 7 displays the general control frame through the support phase. In this frame subscripts ref, res and err symbolizes reference, response and error values respectively. Firstly, ZMP reference and previously determined CoM lenght are inserted to dynamic ZMP equation and θ angular trajectory is obtained. Converting the polar coordinates (r, θ) into cartesian coordinates (x, z) CoM trajectory is extracted. Finally, inverse kinematics block gives us the each joint angle's references using CoM trajectory.

However, based on external effects and/or internal disturbances, ZMP response may not follow the ZMP reference. In order to prevent this problem, ZMP reference is updated in ZMP stabilizer block, based on the following equation on each time step.

$$X_{ZMP}^{vref} = X_{ZMP}^{ref} - K_1 X_{ZMP}^{err} - K_2 \ddot{x} \qquad (13)$$

Here, subscript $vref$ stands for the virtual reference, updated version of the predetermined ZMP value. Nonetheless, inertial force is one of the dominant characteristics of ZMP response and it is caused by horizontal acceleration. Therefore, ZMP reference is updated using ZMP error and measured horizontal acceleration with scaling factors, K_1 and K_2. ZMP is measured using the method explained in [8]. Moreover, angular trajectory stabilizer also compensates θ angular trajectory error with a simple PI controller.

What is more, our proposed method much relies on specifying joint motions. Thus, servo control of actuators are quite important. Each actuator is positonally controlled with torque disturbance observer[12] shown in Fig. 8. This servo control block includes a position loop and a velocity loop. The merit of this servo block is that it has a disturbance observer, estimating the disturbance torque in order to neutralize its effect.

Body altitude controller is employed to keep the undesired body angle zero through the flight phase. It uses gyro feedback in a simple PD loop and upper body does not rotate, especially on air.

IV. Simulation and Experimental Results

A. Simulations

Implementing the proposed method, vertical and forward jumping simulations are performed using ROCOS model of the robot. In both simulations, flight time is measured around 0.1 [s] while the support time is around 0.6 [s].

Vertical jumping simulation results are illustrated in Fig 9 and Fig. 10. Ground reaction force (GRF) is displayed in Fig 9. Zero GRF indicates a successful flight phase cycle even within the presence of touch down impact which may also be noticed. In Fig. 10, vertical foot position is shown, so that, we may say average jumping height is around 1.3 [cm]. These results are also observed in forward jumping case.

Forward jumping simulation results may be seen from Fig.11 to Fig.14. In Fig.11 and Fig. 12, x-axis and z-axis CoM-foot trajectories are displayed respectively. From Fig. 11, we

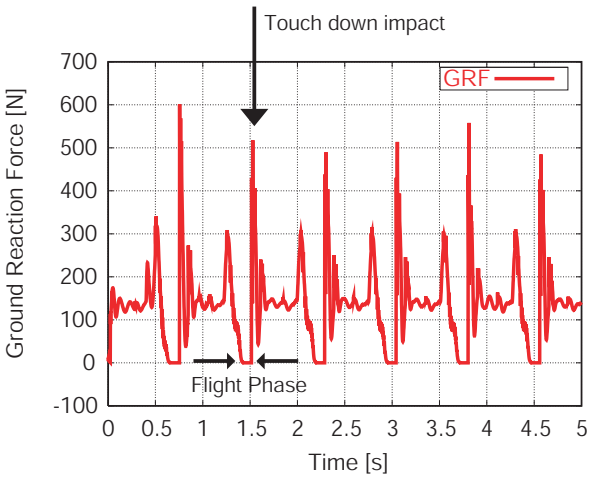


Fig. 9. Measured Ground Reaction Force at the Foot

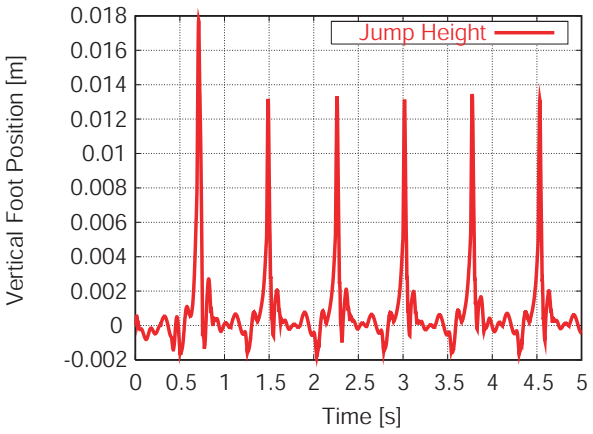


Fig. 10. Foot Position, Vertical Axis

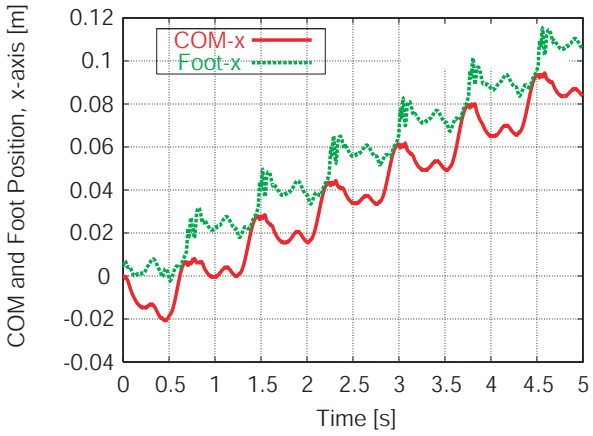


Fig. 11. Horizontal CoM and Foot Positions

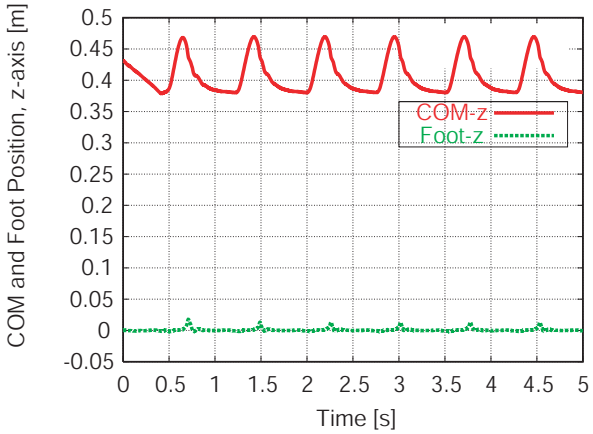


Fig. 12. Vertical CoM and Foot Positions

can see that the robot moves horizontally within the forward velocity of 2.22 [cm/s] and keeps its pace. On the other hand, its jumping height remains the same at each jumping sequence as Fig. 12 indicates.

Pitch axis angular momentum response is plotted in Fig. 13. While the robot is on support phase, it is kept constant around zero. However it never becomes zero as robot moves both horizontally and vertically. As we included angular momentum information in the dynamic ZMP equation, modeling error is expected to be lower. On the other hand, after each landing, angular momentum fluctuates due to the touch down impact.

In Fig. 14, actual ZMP is plotted with foot's toe and hill positions with respect to the moving coordinate frame in which the foot sole center is chosen as the origin. It is seen that actual ZMP is inside the support polygon while the robot is not in the flight phase. When the robot is on air, ZMP is not measured and substituted as zero since there is no contact with floor.

B. Experiments

Having obtained stable and repetitive jumping cycles in simulations, vertical jumping experiment is performed. Since vertical jumping includes only vertical motion, z-axis CoM trajectory is derived by using polynomials as explained in section III-A. In contrast to simulations, jumping frequency is kept low as the robot jumped once in every five seconds. In this experiment, five successive jumping sequences are realized as results indicate.

Fig. 15 shows CoM height during the experiment. It might be observed that CoM height is same at every jumping sequence. Measured GRF is illustrated in Fig. 16. Zero GRF shows a successful flight phase, as flight time is measured as 0.08 seconds.

V. Conclusions

Having implemented the proposed method, vertical and jumping simulations are performed. Resultant simulations point out that aforementioned method provides stable, continous and repetitive jumping cycles both for vertical and forward jumping cases. In the forward jumping case, horizontal velocity is measured as 2.22 [cm/s].

Consequently, vertical jumping experiments are also realized. As a result, we obtained five successive jumping

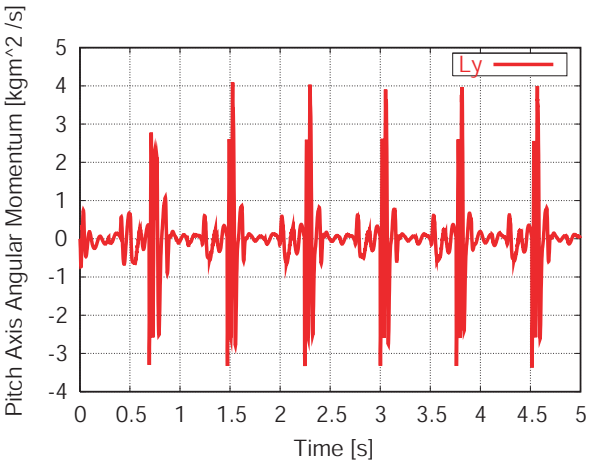


Fig. 13. Pitch Axis Angular Momentum Response

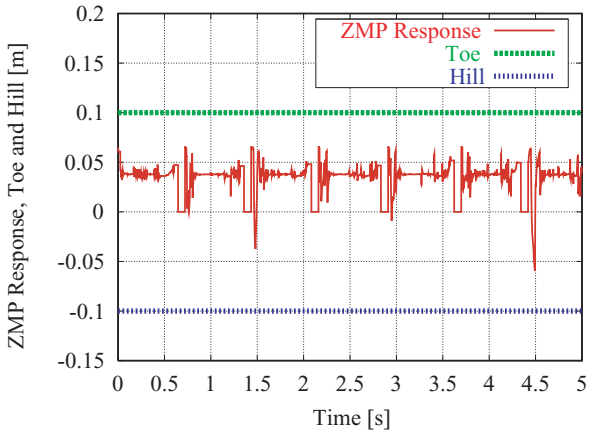


Fig. 14. Measured x-axis ZMP, Toe and Hill Positions

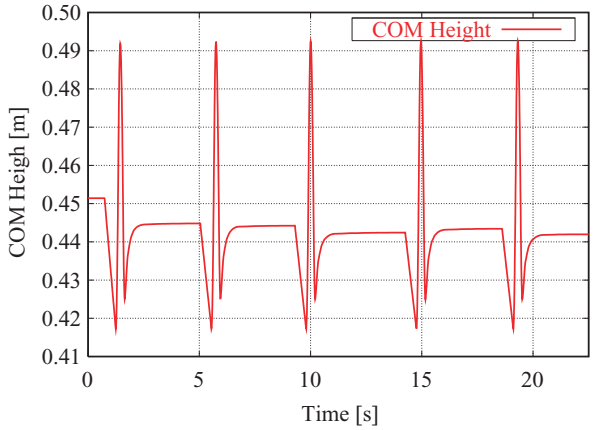


Fig. 15. Measured CoM Height, Experiment

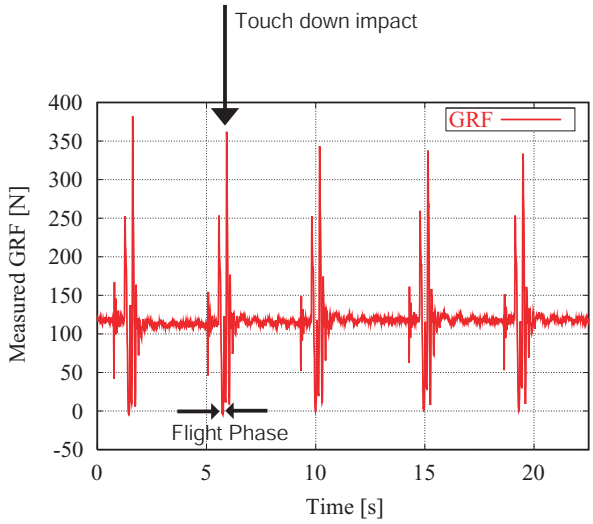


Fig. 16. Measured Ground Reaction Force, Experiment

cycles with flight periods measured as 0.08 seconds. Moreover, jumping height was kept constant in every sequence.

As the future work, proposed method will be extended and applied to biped robots in order to realize jogging and eventually running. In this case, angular momentum expressions will be more complex as sagittal and lateral planes' rotational motions effect each other. A possible trajectory generation for biped robots will be our next assignment.

REFERENCES

[1] R. Tajima, and K. Suga, "Motion having a flight phase: Experiments involving a one-legged robot," in *Proc. IEEE Conf. Intelligent Systems and Robots*, Beijing, China, 2006, pp. 1726-1731.

[2] S. Kajita, T. Nagasaki, K. Kaneko, K. Yokoi, and K. Tanie, "A hop towards running humanoid biped," in *Proc. IEEE Conf. Robotics and Automation*, New Orleans, USA, 2004 pp. 629-635.

[3] S. Kajita, F. Kanehiro, K. Kaneko, K. Yokoi and K. Tanie "Resolved momentum control: Humanoid motion planning based on the linear and angular momenta," in *Proc. IEEE Conf. Intelligent Robots and Systems*, Las Vegas, NV, USA, 2003, pp. 1644-1650.

[4] M. H. Raibert, *Legged robot that balance*, Cambridge, MA: MIT Press, 1986.

[5] S. Timoshenko and D. H. Young *Advanced Dynamics*, McGraw-Hill Book Company, Inc., 1948.

[6] K. Nishiwaki, S. Kagami, Y. Kuniyoshi, M. Inaba and H. Inoue "Online generation of humanoid walking motion based on a fast generation method of motion pattern that follows desired ZMP," in *Proc. IEEE Conf. Intelligent Systems and Robots*, Switzerland, 2002, pp. 2684-2689.

[7] P. Gregorio, M. Ahmadi, and M. Buehler, "Design, control and energetics of an electrically actuated legged robot," in *IEEE Trans. Systems, Man and Cybernetics*, vol. 27, no. 2, pp. 626-634, Aug 1997.

[8] Q. Li, A. Takanishi and I. Kato, "A biped walking robot having a ZMP measurement system using universal force-moment sensors," in *Proc. IEEE Conf. Intelligent Systems and Robots*, Osaka, Japan, 1991, pp. 1568-1573.

[9] Y. Fujimoto, and A. Kawamura "Simulation of an autonomous biped walking robot including environmental force interaction," in *IEEE Robotics and Automation Magazine*, vol. 5, no. 2, pp. 33-42, 1998.

[10] A. Kawamura, and C. Zhu "The development of biped robot MARI-3 for fast walking and running," in *Proc. IEEE Conf. Intelligent Systems and Robots*, Beijing, China, 2006, pp. 599-604.

[11] T. Maeda, B. Ugurlu, and A. Kawamura, "Two legged jumping simulation and experiment on biped robot MARI-3," in *Advanced Motion Control*, Trento, Italy, 2008.

[12] H. Yamamoto, A. Kawamura, and C. Zhu "Experiments on one leg planar jumping robot," in 24^{th} *Japan Robotics Society*, 2F17, 2006. (in Japanese)

[13] B. Ugurlu, and A. Kawamura, "Research on maximum speed of one legged jumping robot," in 25^{th} *Japan Robotics Society*, 1G35, 2007.