

DETC2011-48786

**DEVELOPMENT OF AN OMNIDIRECTIONAL WALKING ENGINE
FOR FULL-SIZED LIGHTWEIGHT HUMANOID ROBOTS**

Seungmoon Song

Robotics Institute
Carnegie Mellon University
Pittsburgh, Pennsylvania 15213
smsong@cmu.edu

Young-Jae Ryoo

Dept. of Control and Robot
Engineering
Mokpo National University
Muan-goon, Korea
yjryoo@gmail.com

Dennis W. Hong

Robotics and Mechanisms Laboratory
Dept. of Mechanical Engineering
Virginia Tech
Blacksburg, Virginia 24061
dhong@vt.edu

ABSTRACT

In this paper, we propose and demonstrate an omnidirectional walking engine that achieves stable walking using feedback from an inertial measurement unit (IMU). The 3D linear inverted pendulum model (3D-LIPM) is used as a simplified model of the robot, the zero moment point (ZMP) criterion is used as the stability criterion, and only the feedback from the IMU is utilized for stabilization. The proposed walking engine consists of two parts; the omnidirectional gait generator, and the stability controller. ZMP equations, derived based on the 3D-LIPM, are used in the omnidirectional gait generator. The solutions of the differential equations are directly used which reduces the computation cost compare to other existing methods. Two kinds of feedback controllers are implemented for the stability controller; one is the indirect reference ZMP controller, and the other is the indirect joint controller. The walking engine is tested on a lightweight, full-sized, 21-degree-of-freedom (DOF) humanoid robot CHARLI-L (Cognitive Humanoid Autonomous Robot with Learning Intelligence, version Lightweight) which stands 141 cm tall and weighs only 12.7 kg. The design goals of CHARLI-L are low development cost, lightweight, and simple design, which all match well with the proposed walking engine. The results of the experiments present the efficacy of our approach.

INTRODUCTION

Bipedal locomotion for humanoid robots is a challenging task, especially for tall, adult sized robots. Though there are many approaches to achieve stable walking, currently the most successful and practical implementations utilize the zero moment point (ZMP) criterion ([1-6]). Some of these robots are

capable of climbing stairs or even running ([7-9]). Most of the successful humanoid robots in this size class are very expensive, heavy, and complicated in design, making them difficult to be used as an affordable and safe robotic platform for research and education.

CHARLI-L (Cognitive Humanoid Autonomous Robot with Learning Intelligence, version Lightweight) is a low-cost humanoid robot developed in the Robotics and Mechanisms Laboratory (RoMeLa) at Virginia Tech ([10]). To achieve the design objectives of a lightweight and low cost system, only an inertial measurement unit (IMU) is used (vs. additional force-torque (F/T) sensors) as the main feedback sensor for bipedal locomotion. Additionally, since the on-board, single-board computer utilizes most of its computational power for vision processing, motion planning, and autonomous behaviors, a walking engine with low computational cost that utilizes only an IMU for feedback was needed. For miniature humanoid robots with a height of less than 60 cm, such as the DARwIn (Dynamic Anthropomorphic Robot with Intelligence) series humanoid robots [11], omnidirectional walking can be achieved based on an open loop motion planner and thus requires much less computational power than full-sized humanoid robots. For CHARLI-L, the challenge is to develop a walking engine that utilizes low computational power for a full-sized humanoid robot. This is possible partially due to the fact that CHARLI-L is lightweight and has no F/T sensors which can simplify the algorithms making it less computational intensive than that for a heavier full-sized humanoid robot.

To achieve this goal, a simplified model of biped robots such as the linear inverted pendulum model (LIPM) ([12]) is used for the walking engine. LIMP models the robot as a point mass at a constant height while connected to the ground by a

zero-mass rod. A 3D version of this simplified model is adopted for our gait generator ([13]). The ZMP criterion is also a popular and practical method for achieving stable bipedal walking ([1,2]). Various methods have been proposed for calculating the trajectory of the center of mass (COM) of the robot that induces the ZMP to follow the desired ZMP trajectory. This is referred to as the “reference ZMP” approach ([14-16]). A preview controller was proposed in [16] that enables on-line calculation of generating the COM trajectory. We propose to directly solve the ZMP equation with the boundary conditions to calculate the COM trajectory, which does not guarantee an on-line solution, but through our experiments, proven to be effective within certain boundaries.

In this paper, we present a computationally efficient on-line omnidirectional walking engine that achieves stable walking using only IMU feedback. This walking engine is proposed for adult-size, lightweight, low cost humanoid robots, and is tested on our humanoid robot platform CHARLI-L.

OVERVIEW OF THE WALKING ENGINE

The bipedal walking engine we present in this paper uses the 3D-LIPM model with the ZMP criterion, and is an omnidirectional walking engine with IMU feedback for stabilization control. The walking engine is omnidirectional in the sense that the robot can take a step toward any direction and the foot in any orientation, allowing it to move forward, backwards, side step, and turn to change directions.

Fig. 1 shows the block diagram of the walking engine and the servo motor actuators used at each of its joints. For low-level control, distributed control is used as each actuator has

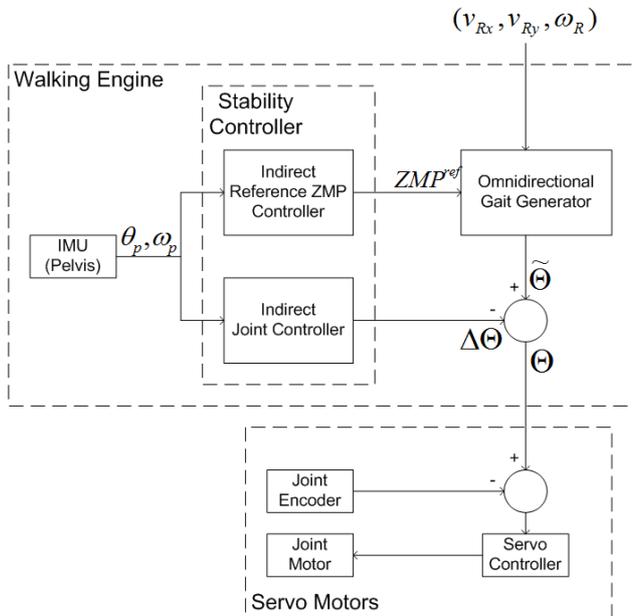


FIGURE 1. BLOCK DIAGRAM OF THE WALKING ENGINE

their own servo controller and position encoder. The actuators communicate through a RS-485 serial bus, but the walking engine does not read the encoder positions from each joint due to the limit of the communication speed.

The walking engine is divided into two processes, the omnidirectional gait generator and the stability controller. The stability controller is composed with two controllers; the indirect reference ZMP controller and the indirect joint controller. The omnidirectional gait generator and the indirect reference ZMP controller run every 20 ms, and the indirect joint controller runs every 5 ms.

The walking engine is only concerned about controlling the legs of humanoid robot for locomotion. While arms swing in a sinusoidal motion, which helps the stability by compensating the yaw moment ([17]), we have not yet investigated its impact on our robot.

OMNIDIRECTIONAL GAIT GENERATOR

A block diagram of the omnidirectional gait generator is shown in Fig. 2. The input into the omnidirectional gait generator is the desired velocity of the COM along the x-axis and y-axis, v_{Rx} and v_{Ry} , and the angular rate of the orientation of the body around the z-axis, ω_R , in the robot-frame. The origin of the robot-frame is on the COM of the robot, with the x-axis pointing forward, y-axis pointing to the left, and z-axis pointing up. The outputs from the omnidirectional gait generator are the joint positions for the leg actuators, $\Theta = \{\theta_{L1}, \theta_{L2}, \dots, \theta_{R1}, \theta_{R2}, \dots\}$.

Next Step Pose Generator

We first define a pose of a moment called Next-Step-Pose as the pose at the middle of the next double stance phase. The omnidirectional gait generator first calculates the pelvis and footstep positions of the Next-Step-Pose from the input $(v_{Rx}, v_{Ry}, \omega_R)$, which is a continuous value in a given boundary.

We claim our walking to be omnidirectional not only because it is able to move its foot in any direction to place the next footstep, but also because we can change the direction of the desired next footstep at any time. In other words, changing the input $(v_{Rx}, v_{Ry}, \omega_R)$ is effective even when the robot is

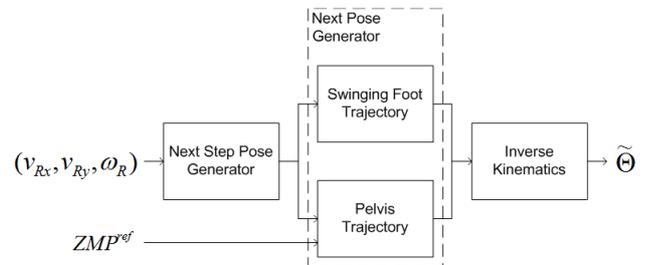


FIGURE 2. BLOCK DIAGRAM OF THE OMNIDIRECTIONAL GAIT GENERATOR

swinging its leg. To ensure stability, we impose an empirically chosen limit on the change of the input to prevent radical changes of the Next-Step-Pose.

Next Pose Generator

Where the pelvis and the swinging foot should be at the very next 20 ms are calculated based on the Next-Step-Pose.

Swinging Foot Trajectory: We adapt the idea that the swinging foot should move smoothly from the current position to the Next-Step-Pose with near zero velocity when the foot lifts and lands ([18]). Cubic spline interpolation is used to generate this smooth trajectory. While generating these trajectories is quite intuitive our gait generator defines the motion along the z-axis to help the hip joint lifting the swinging foot.

Pelvis Trajectory: The 3D-LIPM and the ZMP criterion are used in our walking engine to generate the pelvis trajectory, which we assume is identical to the COM trajectory. There are studies on proposing a trajectory of the reference ZMP for human-like walk ([19,20]). However, we choose the reference ZMP to be at the center of the supporting foot during the single stance phase and move immediately to the center of the other foot at the middle of double stance. Adapting the 3D-LIPM and assuming the COM remains at a constant height while walking, the relationship between the position of the ZMP and the COM can be obtained by

$$p_x = x_{CoM} - \frac{z_{CoM}}{g} \ddot{x}_{CoM} \quad (1)$$

$$p_y = y_{CoM} - \frac{z_{CoM}}{g} \ddot{y}_{CoM} \quad (2)$$

where p_x, p_y are the positions of the ZMP along the x and y-axis, x_{CoM}, y_{CoM} and z_{CoM} are the positions of the COM, along the x,y and z-axis where z_{CoM} is a constant, and g is gravity.

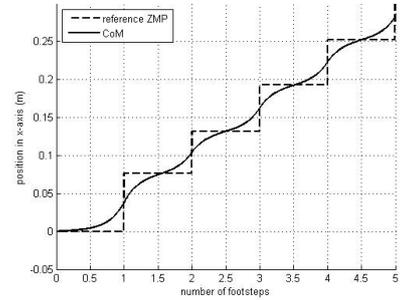
The issue is how to generate the trajectory of the COM or the pelvis to induce the ZMP to match the reference ZMP, which is the inverse problem of (1) and (2). Generating the COM trajectory by the preview controller was proposed in [16]. However, we directly solve the ordinary differential equations with boundary conditions of the current position and the position at the Next-Step-Pose of the pelvis. If we solve this in the supporting-foot-frame of which orientation is same as the robot-frame and the origin is at the center of the supporting foot in where $p_x = p_y = 0$, the equations are solved as

$$x(t) = x_0 \cdot \cosh\left(\sqrt{\frac{g}{z_{CoM}}} \cdot t\right) + \left\{ x_T - x_0 \cdot \cosh\left(\sqrt{\frac{g}{z_{CoM}}} \cdot T\right) \right\} \cdot \frac{\sinh\left(\sqrt{\frac{g}{z_{CoM}}} \cdot t\right)}{\sinh\left(\sqrt{\frac{g}{z_{CoM}}} \cdot T\right)} \quad (3)$$

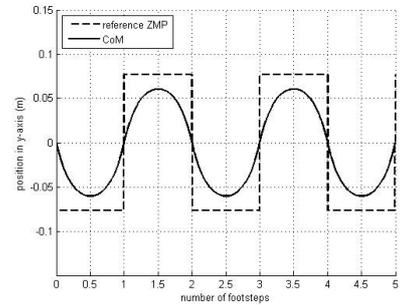
$$y(t) = y_0 \cdot \cosh\left(\sqrt{\frac{g}{z_{CoM}}} \cdot t\right) + \left\{ y_T - y_0 \cdot \cosh\left(\sqrt{\frac{g}{z_{CoM}}} \cdot T\right) \right\} \cdot \frac{\sinh\left(\sqrt{\frac{g}{z_{CoM}}} \cdot t\right)}{\sinh\left(\sqrt{\frac{g}{z_{CoM}}} \cdot T\right)} \quad (4)$$

where current time is $t=0$, x_0, y_0 are the current pelvis position, x_T, y_T are the goal position of the pelvis at the end of the current step, equal to the center of both feet at next double stance, and T is the remaining time until the end of the current step. Whenever the equation is solved, current time is $t=0$ and calculates for next t , which is 20 ms.

Equations (3) and (4) give the trajectory of the COM that induces the ZMP to be at the center of the supporting foot as shown in Fig. 3 for forward walking. This is only true when the



(a) x-axis in the robot-frame



(b) y-axis in the robot-frame

FIGURE 3. TRAJECTORIES OF THE REFERENCE ZMP AND THE CORRESPONDING COM DURING A SEQUENCE OF FIVE STEPS OF FORWARD WALKING

Next-Step-Pose does not change during the step. If the Next-Step-Pose changes during the step, stable walking is not guaranteed by trajectories generated from (3) and (4); however, experiment results show that we can still use these equations within certain boundaries.

Inverse Kinematics

Once the positions of the pelvis and feet for the very next 20 ms are calculated, the output Θ can be by solving the inverse kinematics of the legs. The inverse kinematics of CHARLI-L are presented in ([10]).

STABILITY CONTROLLER

Sensor feedback control in bipedal walking is critical. This is especially true for adult-size humanoid robots to handle disturbances and to compensate for the approximate underlying models. For example, our gait generator used the 3D-LIPM to enable real-time computation but the results are not exact due to the approximations made for model used. In addition, there are uncertainties arising from the hardware, which come from the backlash from the gear train and deflections of the links. As such, the feet do not always reach the intended goal positions precisely. For CHARLI-L, servomotors that are not commonly used for high torque purposes use utilized and thus the position control of the joints were not exact.. The stability controller is designed to overcome these imperfections to enable stable walking.

Before designing the stability controller, we first needed to define the criterion of stability. One of the most widely used criteria is keeping the ZMP in the supporting polygon. If the ZMP can be measured in real-time, this definition is effective because it guarantees walking without falling. However, to measure the ZMP directly, the robot needs other sensors such as force/torque sensors ([21]). Therefore, we define stable walking as walking with a pelvis parallel to the ground, which can easily be measured by an IMU attached to the pelvis. This makes sense because the pelvis of the poses generated from our proposed gait generator is supposed to be parallel with the feet all the time. Therefore, unless the robot collapses, it will not fall down with a horizontal pelvis.

With this criterion of stability, we propose two indirect feedback controls: indirect reference ZMP control and indirect joint control. Both controls use the data from the IMU.

Indirect Reference ZMP Control

In the gait generator, we use the reference ZMP to calculate the trajectory of the COM. The reference ZMP was assigned to be at the center of the supporting foot during single stance and immediately shift to the other foot at the middle of double stance. However, because of the error in the 3D-LIPM and the mechanical system, the actual ZMP may not follow the reference ZMP.

While the ZMP is in the supporting polygon the robot does not fall down, even if the ZMP is not at the center of the polygon. However, if the ZMP reaches an edge of the supporting polygon, the foot of the robot starts to roll over. If we could measure the ZMP, this can be prevented by controlling the reference ZMP to keep the ZMP at the desired location ([22]).

A similar control strategy can be used without measuring the ZMP. Once the ZMP gets to an edge of the supporting polygon the pelvis will start to tilt towards the direction of the edge, and this can be detected by the IMU at the pelvis. We can use this data instead of the measured ZMP data. The controller applies the proportional-derivative (PD) control of the reference ZMP with the IMU data. This is different from the indirect ZMP controller introduced in [23], where the reference ZMP is controlled with the COM error. We propose to control the reference ZMP with the pelvis angle error to keep the pelvis parallel with the ground. The controller does not control the ZMP to match the reference ZMP but to keep it in the supporting polygon. The equations for the PD control of the reference ZMP with the IMU data is

$$p_x^{ref}(t) = p_x^{ref_open}(t) + Z_{Px} \theta_{pitch}(t) + Z_{Dx} \omega_{pitch}(t) \quad (5)$$

$$p_y^{ref}(t) = p_y^{ref_open}(t) + Z_{Py} \theta_{roll}(t) + Z_{Dy} \omega_{roll}(t) \quad (6)$$

where p^{ref_open} is the position of the reference ZMP before feedback, θ_{pitch} and ω_{pitch} are the current pitch angle and angular rate, θ_{roll} and ω_{roll} are the current roll data from the IMU, Z are the corresponding PD gains, and p^{ref} is the resulting position of the reference ZMP.

Because the input and output dimensions of our PD controller are different, calculating the gain cannot be done directly. We obtain gains in an empirical fashion.

Indirect Joint Control

While the reference ZMP controller is the coarse compensator, the joint controller is the fine compensator. The gait is revised by the reference ZMP controller every 20 ms, and the joint controller tunes the joints individually every 5 ms. This process is different with the controls in the individual servo motors. The indirect joint controller controls each joint based on the IMU data, similar to [24]. In [24], only angular rate was implemented for a small humanoid robot. We use both angle and angular rate information about the roll and pitch. There is no feedback control on the yaw angle assuming that the yaw motion has little influence on the stability comparing to the roll and pitch.

We control corresponding joints based on the information about the roll and pitch. This depends on the structure of the legs of the robot. Most common humanoid legs contains 6-DOF – which consist of a 3-DOF hip, a 1-DOF knee and a 2-DOF ankle. The hip, knee, and ankle joints generate roll motion, and

the hip and ankle joints generate pitch motion ([3-5]). For CHARLI's 5-DOF legs with four-bar linkages, the hip and knee joints generate pitch motion, and the two knee joints generate the pitch motion.

PD control via IMU data for our indirect joint controller is given by

$$j_5(t) = j_5^{open}(t) - J_{P5}\theta_{roll}(t) + J_{D5}\omega_{roll}(t) \quad (7)$$

$$j_2(t) = j_2^{open}(t) - J_{P2}\theta_{roll}(t) + J_{D2}\omega_{roll}(t) \quad (8)$$

for roll compensation and

$$j_4(t) = j_4^{open}(t) - J_{P4}\theta_{pitch}(t) + J_{D4}\omega_{pitch}(t) \quad (9)$$

$$j_3(t) = j_3^{open}(t) - J_{P3}\theta_{pitch}(t) + J_{D3}\omega_{pitch}(t) \quad (10)$$

for the pitch compensation. J are the corresponding PD gains of the joints, j^{open} is angle value before feedback, and j_5, j_2, j_4 and j_3 represents the angle values of the ankle, lower hip, lower knee and upper knee joints, respectively, where the joints are named by the positions of those actuators.

Our control gains are obtained through testings. We first test the reference ZMP controller to choose the control gains with the joint controller deactivated. We then obtain gains for the joint controller. Among the joint gains, those for (7) and (9) are first decided followed by those for (8) and (10). This is done to find controller gains which have a greater affect on the stability first. For example, the position of the COM relative to the supporting foot changes more through the ankle joints, j_5 , than the hip joints, j_2 .

EXPERIMENTS

CHARLI-L, the Experimental Platform

Our test platform CHARLI-L is a humanoid robot that stands 141 cm tall weighs only 12.7 kg. CHARLI-L was designed to be a light weight, low cost, untethered, autonomous robot which walks only on flat, even surfaces, and this well matches with the proposed walking engine. Fig. 4 shows the overall dimensions of CHARLI-L.

To reduce the weight of the robot, a double parallel four-bar linkage is used for the leg structure to reduce a DoF. As a result, while 6-DOF legs are most common in humanoid robots, CHARLI-L has only 5-DOF per leg. Because of this, the pitch motion of the feet is sacrificed and thus the foot is always parallel to the ground. This limits CHARLI-L's locomotion to flat, even terrains.

Experiment Set Up and Procedure

Table 1 shows the empirically determined gait parameters from our gait generator for CHARLI-L.

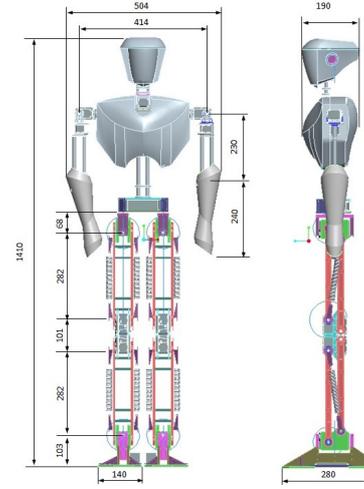


FIGURE 4. DIMENSIONS OF CHARLI-L (mm)

To obtain the velocity region in which stable walking is assured, we first tested each maximum speed in forward, backward, sideways walk and turning motion separately. These become the range of each component of the input, $(v_{Rx}, v_{Ry}, \omega_R)$. Then, the input magnitude was limited by $\sqrt{v_{Rx}^2 + v_{Ry}^2 + \omega_R'^2} \leq v_{max}$, where $\omega_R' = 0.15\omega_R$ is the scaled angular velocity and v_{max} is the maximum forward speed. The scaling factor 0.15 is to weight rotational velocity which is found empirically. Walking velocity in this region was then tested for stability.

Experiment Results

Omnidirectional Walking: With our proposed walking engine, CHARLI-L walks successfully in omni-directions within the range given in Table 2. The acceleration is limited to 7 cm/sec², which achieves the maximum forward speed in one

TABLE 1. GAIT PARAMETERS

Parameter	Value
Foot-step period (sec)	1
% of single stance	0.4
Height of pelvis while walking (cm)	85 (=87-2)
Stance distance (cm)	15.4
Step height (cm)	8

TABLE 2. MAXIMUM WALKING VELOCITIES

Walk type	Velocity
Walking forwards	7 cm/sec
Walking backwards	5 cm/sec
Side stepping	3 cm/sec
Turning in place	0.17 rad/sec

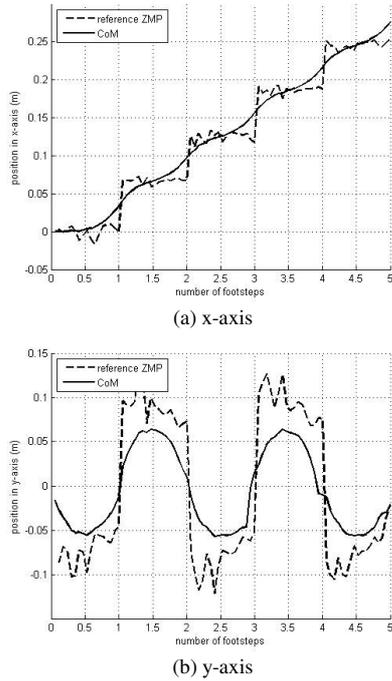


FIGURE 5. TRAJECTORIES PLANNED FOR THE REFERENCE ZMP AND CoM DURING A SEQUENCE OF FIVE STEPS OF FORWARD WALKING WITH THE FEEDBACK CONTROLLERS

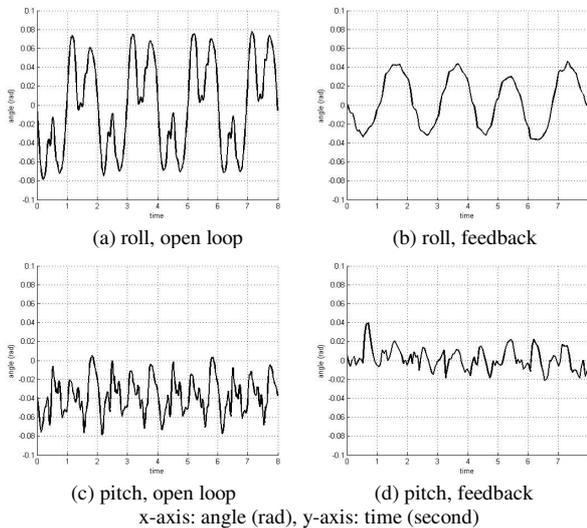


FIGURE 6. DATA FROM THE IMU DURING EIGHT FORWARD WALKING STEPS

step.

Stability Control: To test the controller's effectiveness, we compared the walking engine with the IMU to the version without the stability controller. All the data presented in Fig. 5

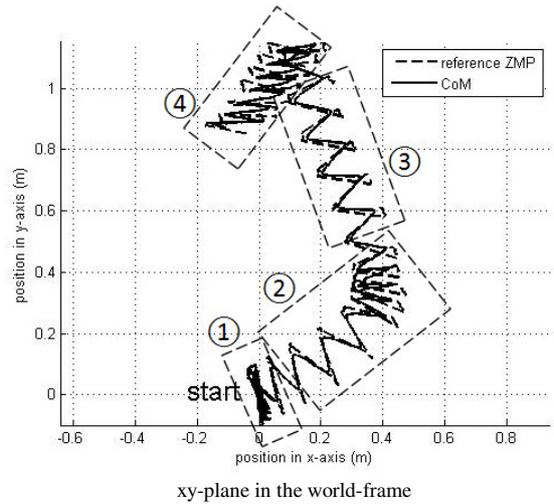


FIGURE 7. TRAJECTORIES PLANNED FOR THE REFERENCE ZMP AND CoM DURING VARIOUS TYPES OF WALKING WITH THE FEEDBACK CONTROLLERS

and 6 is based on a forward walking at 6 cm/sec. The trajectories planned on-line for the reference ZMP and CoM with the indirect reference ZMP controller are shown in Fig. 5, which we can compare with Fig. 3.

Fig. 6 shows the IMU data of the walking engines without and with the stability controller. Comparing the data in (a) to (b), the improvement for the roll angle in the walking engine is revealed. The trajectory became smoother, and the amplitude of the trajectory became smaller through application of feedback control. This indicates the upper body tilts less during walking. Large fluctuations in open-loop walking seem to stem from the mismatch between the natural frequency and the walking frequency. However, feedback control compensates for this phenomenon.

(c) and (d) in Fig. 6 show that a similar conclusion can be drawn for the pitch angle. In addition all data is shifted to negative angle in (c), which indicates a tendency of leaning backwards during open loop walking. (d) shows that this is fixed by our proposed stabilization controls.

Fig. 7 shows the trajectories planned on-line for the reference ZMP and CoM during various types of walking with IMU feedback. These types of walking are generated by changing the input $(v_{Rx}, v_{Ry}, \omega_R)$ of the gait generator as explained before. The figure shows the sequence of ① left turn, ② left turn while forward, ③ forward-leftward diagonal, and ④ backward-leftward diagonal walking.

CONCLUSION

In this paper, a computationally efficient walking engine for full-sized lightweight humanoid robots consisting of an

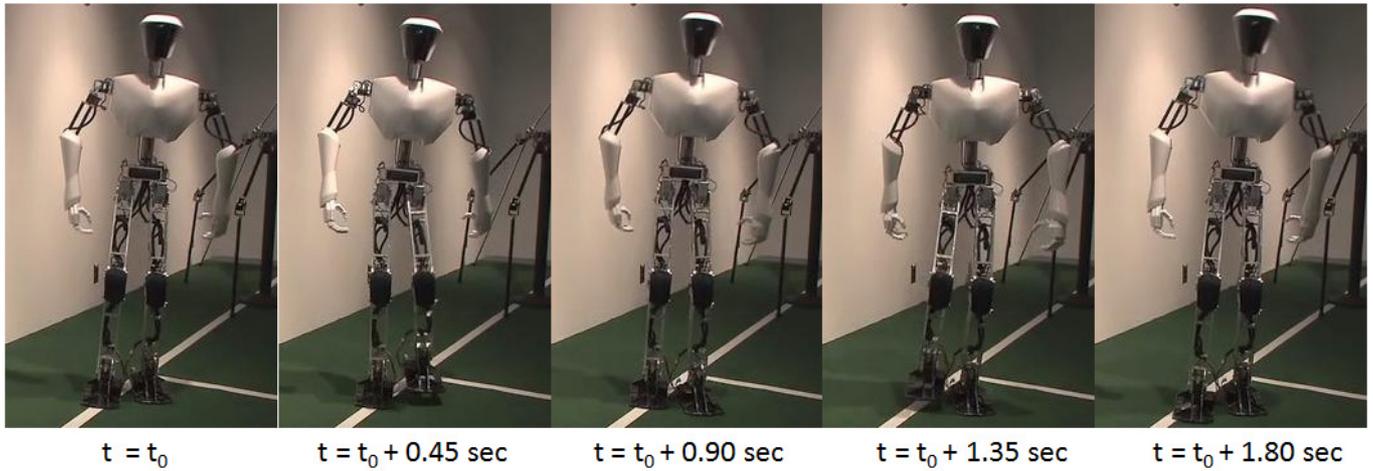


FIGURE 8. CHARLI-L WALKING FORWARD

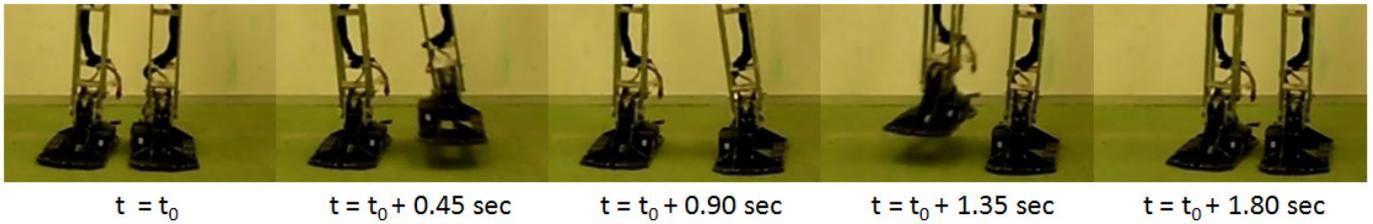


FIGURE 9. CHARLI-L SIDE STEPPING

omnidirectional gait generator and a stability controller is presented. The role of the omnidirectional gait generator is to allow the robot to take a step towards any direction for high mobility. The stability controller stabilizes the walking motion to prevent falling and to enable smoother walking motions, using only IMU feedback data. To reduce the computation cost ZMP equations are derived based on the 3D-LIPM for the omnidirectional gait generator, and the solutions of the resulting differential equations are directly used.

Difference between the calculated gait and the actual gait exist because of the approximations made in the robot model and the miscellaneous uncertainties in the actual physical system. Stability controllers using IMU feedback are used to compensate for these errors, which consist of two indirect feedback loops; one for the reference ZMP trajectory and another for the joint trajectories. PD controls are used in both loops with the IMU data from the sensor located at the pelvis. A simple constraint of keeping the pelvis parallel to the ground for the controller was imposed. The proposed walking engine matches well with the design concepts of CHARLI-L which is light-weight, low-cost, and simple design.

Through the experiments conducted with CHARLI-L, our proposed walking engine was proven to work successfully on flat even ground. In a given range of walking speeds, CHARLI-L is capable of stable omnidirectional walking. We also demonstrated the effectiveness of the stability controller by

comparing the IMU data with and without the indirect feedback controllers.

As the walking engine for the CHARLI-L humanoid robot platform, we have successfully developed a computationally efficient method that can achieve stable omnidirectional gaits. However, the performance of the walking of this system is still less than many of the other successful heavy and expensive humanoid robots, and even more remote to that of a human. Though we can improve the performance by adding more sensors and controllers, it is always a design trade off between improved performance and cost, complexity, and weight. However, as a unique class of robot CHARLI-L is, the proposed walking engine is well suited for this particular system.

Future work includes the development of a newer version, CHARLI-L2, which utilizes new joint designs that can produce higher torques, which will also improve the performance. Humanoid robot development is an art of balance and harmony between hardware and software design, and thus both needs to be considered concurrently for a successful system.

NOMENCLATURE

CHARLI-L Cognitive Humanoid Autonomous Robot with Learning Intelligence, version Lightweight

COM Center of Mass

DARwIn Dynamic Anthropomorphic Robot with Intelligence

DOF Degree Of Freedom

IMU Inertial Measurement Unit
PD Proportional-Derivative
ZMP Zero Moment Point

ACKNOWLEDGMENTS

The authors would like to recognize the National Science Foundation and the Office of Naval Research for partially supporting this work through grants OISE 0730206, CNS 0960061, CNS 0958406, and ONR 450066.

REFERENCES

- [1] M. Vukobratovic, A. A. Frank, and D. Juricic, 1970, "On the Stability of Biped Locomotion," *IEEE Transactions on Bio-medical Engineering*, BME-17(1).
- [2] M. Vukobratovic, 2004, "Zero-Moment Point—Thirty Five Years of Its Life," *Int. Journal of Humanoid Robotics*, 1(1).
- [3] K. Hirai, M. Hirose, Y. Haikawa, and T. Takenaka, 1998, "The Develop of Honda Humanoid Robot," *Proc. IEEE Int. Conf. on Robotics & Automation*.
- [4] K. Kaneko et al., 2004, "Humanoid Robot HRP-2," *Proc. IEEE Int. Conf. on Robotics & Automation*.
- [5] J. Kim et al., 2005, "System Design and Dynamic Walking of Humanoid Robot KHR-2," *Proc. IEEE Int. Conf. Robotics and Automation*.
- [6] Y. Ogura et al., 2006, "Development of a New Humanoid Robot WABIAN-2," *Proc. IEEE Int. Conf. Robotics and Automation*.
- [7] M. Hirose and K. Ogawa, 2007, "Honda Humanoid Robots Development," *Phil. Trans. R. Soc. A*, 365, pp. 11-19.
- [8] S. Kajita, T. Nagasaki, K. Kaneko, K. Yokoi, and K. Tanie, 2005, "A Running Controller of Humanoid Biped HRP-2LR," *Proc. IEEE Int. Conf. Robotics and Automation*.
- [9] B. Cho, S. Park, and J. Oh, 2009, "Controllers for Running in the Humanoid Robot, HUBO," 9th IEEE-RAS Int. Conf. on Humanoid Robots.
- [10] S. Song, 2010, "Development of an Onmi-Directional Gait Generator and a Stabilization Feedback Controller for Humanoid Robots," M.S. thesis, Dept. Elect. Eng., Virginia Polytech Institute and State Univ.
- [11] K. Muecke, D. W. Hong, and S. Lim, 2007, "DARwIn's Evolution: Development of a Humanoid Robot," *IEEE International Conference on Intelligent Robotics and Systems*.
- [12] S. Kajita et al., 2002, "A Real Time Pattern Generator for Bipedal Walking," *Proc. IEEE Int. Conf. Robotics and Automation*, 1, pp. 31-37.
- [13] S. Kajita, F. Kanehiro, K. Kaneko, K. Yokoi, and H. Hirukawa, 2001 "The 3D Linear Inverted Mode: A Simple Modeling for a Biped Walking Pattern Generation," *Proc. IEEE/RSJ Int. Conf. Intelligent Robots and Systems*.
- [14] A. Takanishi, H. Lim, M. Tsuda, and I. Kato, 1990, "Realization of Dynamic Biped Walking Stabilized by Trunk Motion on a Sagittally Uneven Surface," *Proc. IEEE Int. Workshop on Intelligent Robots and Systems*, pp. 323-330.
- [15] S. Kagami, K. Nishiwaki, T. Kitagawa, T. Sugihara, M. Inaba, and H. Inoue, 2000, "A Fast Generation Method of a Dynamically Stable Humanoid Robot Trajectory with Enhanced ZMP Constraint," *Proc. IEEE Int. Conf. Humanoid Robotics*.
- [16] S. Kajita et al., 2003, "Biped Walking Pattern Generation by Using Preview Control of Zero-Moment Point," *Proc. IEEE Int. Conf. Robotics and Automation*.
- [17] J. Yamaguchi, A. Takanishi, I. Kato, 1993, "Development of a Biped Walking Robot Compensating for Three-Axis Moment by Trunk Motion," *Proc. IEEE/RSJ Int. Conf. Robots and Systems*.
- [18] J. Strom, G. Slavov, and E. Chown, 2009, "Omnidirectional Walking Using ZMP and Preview Control for the Nao Humanoid Robot," In *RoboCup 2009: Robot Soccer WorldCup XIII*.
- [19] K. Erbatur, O. Koca, E. Taskiran, M. Yilmaz, and U. Seven, 2009, "ZMP Based Reference Generation for Biped Walking Robots," *ICICRA'09, Int. Conf. Intelligent Control, Robotics and Automation*.
- [20] E. Taskiran, M. Yilmaz, Ozer Koca, U. Seven, and K. Erbatur, 2010, "Trajectory Generation with Natural ZMP References for Biped Walking Robot SURALP," *IEEE Int. Conf. Robotics and Automation*.
- [21] K. Erbatur, A. Okazaki, T. Takahashi, and A. Kawamura, 2002, "A Study on the Zero Moment Point Measurement for Biped Walking Robots," *Proc. IEEE Int. Workshop on Advanced Motion Control*, page 431.436.
- [22] K. Nishiwaki and S. Kagami, 2010, "Strategies for Adjusting the ZMP Reference Trajectory for Maintaining Balance in Humanoid Walking," *IEEE Int. Conf. on Robotics and Automation*.
- [23] Y. Choi, B. J. You, and S. R. Oh, 2004, "On the Stability of Indirect ZMP Controller for Biped Robot Systems," *Proc. of Int. Conf. Intelligent Robots and Systems*, 2, pp: 1966-1971.
- [24] J. Baltes, S. McGrath, and J. Anderson, 2004, "Active Balancing Using Gyroscopes for a Small Humanoid Robot," 2nd Int. Conf. Autonomous Robots and Agents.