Proceedings of the 2007 IEEE/RSJ International
Conference on Intelligent Robots and Systems
San Diego, CA, USA, Oct 29 - Nov 2, 2007

Adjustment of Home Posture of a Biped Humanoid Robot Using an Inertial Sensor and Force Torque Sensors

Jung-Hoon Kim, Jung-Yup Kim and Jun-Ho Oh

Abstract— **In the walking control of a biped humanoid robot, the walking performance is seriously affected by the initial home posture. An important thing is that the initial home posture of real robot is slightly different at every setting because a zero position of joint is not exactly the same. Accurate and consistent initial home posture is essential when we compare and analyze walking control algorithms. In general, the conventional method for electric motor uses an incremental encoder with a limit switch or an absolute encoder such as potentiometer to make a zero position. However, this method for the multi-axis humanoid robot is difficult, time-consuming, and inaccurate. Furthermore, it has the disadvantage that additional limit switch or absolute encoder can interfere with the design objective of compactness. This paper describes a novel adjustment method of home posture for a humanoid robot utilizing incremental encoders, an inertial sensor and force torque sensors. Four kinds of controllers are proposed and adjusted offsets are measured when outputs of the controllers are converged. The experimental results from KHR-2 show the effectiveness of the proposed algorithm.**

I. INTRODUCTION

Research on humanoid robots has been increasing since the successful debut of the Honda humanoid robot[1]. Humanoid robotics research can be classified into mobility, manipulation, navigation, intelligence, and interaction. Among these, dynamic biped walking control is related to the mobility, and it is fundamental research for realizing a humanoid robot that can move in a human environment.

The Humanoid Research Center in KAIST has developed humanoid robots since 2001[2]–[7]. KHR-1(KAIST Humanoid Robot Platform-1) was a first platform for testing biped walking capability [2]–[4]. After achieving successful dynamic walking, we designed the second platform KHR-2, which has an enhanced control architecture based on a distributed control system using a CAN (Controller Area Network)[5]. It can walk, climb stairs, and perform visual tracking using a single CCD camera. HUBO is our third platform. It has a beautiful appearance and a strong mechanical structure. It can change step length and walking frequency while walking and can perform upper body motions based on Fast Fourier Transform and human motion capture [6].

As essential sensors for walking, many humanoid robots are equipped with an inertial sensor on the torso and force torque sensors at the bottom of each feet. In order to measure the joint angle, an incremental encoder or an absolute encoder like potentiometer can be used. For incremental encoder, a limit switch is necessary to set a zero position of the joint. As for an absolute encoder, it does not need a limit switch, but generally its resolution is lower than that of an incremental encoder and a potentiometer signal can drift according to the temperature. Ideally, if the robot is moved into the home posture under position control after setting the zero position of each joint, the upper body angle from an inertial sensor should be zero and the weight on the each foot should be same. However, in reality, it is hard for a multi-axis robot to achieve an ideal home posture and force condition on the foot because of the accumulated errors of all joints. These accumulated errors are composed of two things. One is an error due to an inaccurate zero position of the joint. The other is a position control error of the joint. In this paper, the authors only address an error from an inaccurate zero position of the joint because it cannot be eliminated by using excellent joint position controller. In order to adjust the home posture, we can modify the initial angles of all joints to realize zero upper body angle and same weight on the each foot through trial and error method, but this is not only hard but also time-consuming since there are so many axes on a humanoid robot. In addition, compact design of a humanoid robot cannot be achieved by attaching limit switches on all the robot joints.

If the initial ZMP(Zero Moment Point)[7] is not located at the ideal point, the actual robot's pose does not perfectly correspond with the desired pose derived by the inverse kinematics solution. Under this condition, the robot becomes unstable when it walks or will have limited walking performance. Improving the accuracy and repeatability of the home posture is very important not only for walking performance but also for developing the walking control algorithm since constant experimental conditions help consistent experimental results

Therefore, this paper proposes an effective adjustment method of home posture for humanoid robots using an inertial

This work was mainly supported by MOCIE (Ministry of Commerce, Industry and Energy) and partly supported by HWRS (Human Welfare Robotic System) and BK-21 (Brain Korea - 21) project.

Jung-Hoon Kim is with the Mechanical Engineering Department, Myoungji University, San 38-2, Namdong, Cheoin-Gu, Yongin, Gyeonggi-Do, 449-728, South Korea (corresponding author, Tel. +82-31-330-6421; e-mail: jhkim2006@mju.ac.kr).

Jung-Yup Kim is with Carnegie Mellon University, 5000 Forbes Avenue Pittsburgh, PA 15213, on leave from the Mechanical Engineering Department, KAIST (e-mail: jungyup@cs.cmu.edu).

Jun-Ho Oh is with the humanoid Research Center of the Mechanical Engineering Department, KAIST, 373-1 Kusong-dong, Yusong-gu, Taejon, 305-701, South Korea (e-mail: jhoh@kaist.ac.kr).

sensor and force torque sensors. We introduce four kinds of controllers to find offsets and adjust the initial home posture. As a result, the proposed adjustment method enables the consistent walking control performance without additional limit switches or absolute encoders.

This paper is organized as follows: In section 2, we briefly introduce the robot platform of KHR-2. Section 3 describes a adjustment algorithm. Section 4 presents the experimental results of the adjustment algorithm. Finally, in section 5, we conclude the paper with discussion and future work.

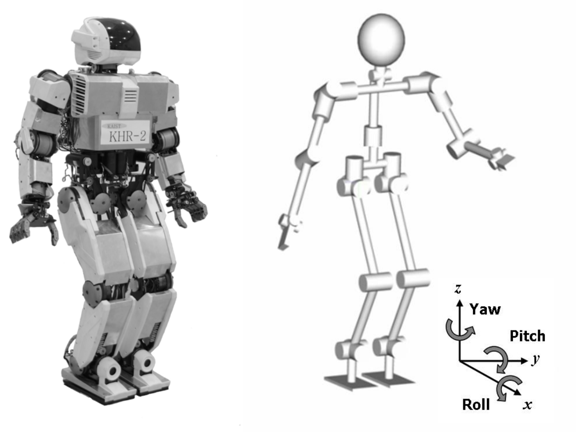

Fig. 1 Photograph of the biped humanoid robot KHR-2 and its joint structure

II. PLATFORM DESCRIPTION

A. Mechanical Hardware

In 2003, we developed KHR-2, the second version of our biped humanoid robot (Fig. 1). For a human-friendly intelligent robot, we designed it to be the size of a child and gave it sufficient joints to imitate human motions. The weight, height and D.O.F (degrees of freedom) of KHR-2 are 56 kg, 120 cm and 41 (6 for each leg, 4 for each arm, 7 for each hand, 1 for the torso and 6 for the head) respectively [5]. All joint actuators are brushed DC motors with harmonic reduction gears or planetary gears. We achieved a self-contained system by putting all mechanical and electronic parts into the robot body. Hence, KHR-2 is tele-operated via wireless LAN (Local Area Network).

B. Control System Architecture

The control system architecture of KHR-2 is a distributed control system. The main computer is installed in the torso, and sub-controllers such as joint motor controllers and sensory devices are distributed throughout the body. Communication between the main computer and the sub-controllers is achieved by using a CAN (Controller Area Network) protocol. The specifications and descriptions of the sub-controllers including sensors are presented in [5].

III. ADJUSTMENT ALGORITHM

Adjustment of our humanoid robot is performed by four kinds of controller using the inertial sensor and force torque sensors. They are ZMP Regulation Controller, Orientation Correction Controller, Compliance Controller on the roll joint of the ankle and Ankle Torque Difference Controller. It is important to note that these controllers are activated simultaneously not one by one. As a result of adjustment, offset values of the four controllers are found. If we add the offsets to the reference position in the Cartesian space or joint space, we can minimize the difference between real and experimental conditions and obtain good walking performance. For ground, which is perfectly flat and perpendicular to the gravity, we proposed to use the Balance Plate as shown in Fig. 2. It has three supporters and a level meter so that an ideal ground can be easily obtained. During the adjustment process, the robot stands on the Balance Plate in double support. After the adjustment, the robot comes down on the real ground.

Fig. 2. Photograph of the Balance Plate

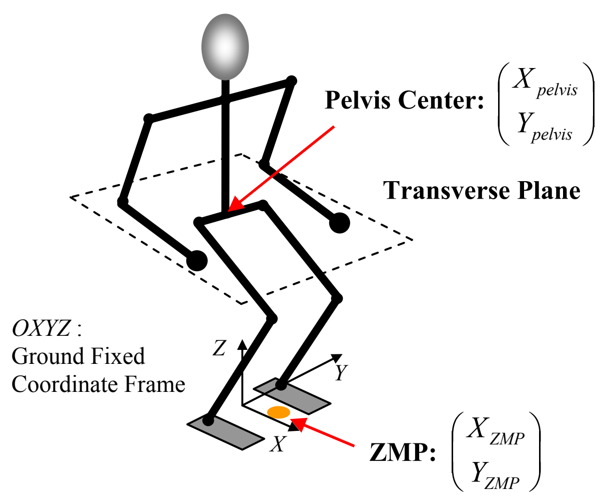

Fig. 3. Schematics of home posture and ZMP regulation controller

A. ZMP Regulation Controller (ZRC)

As a ready posture for walking, the robot usually holds the posture as shown in figure 3, where the robot is bending its knees a little and keeping a pelvis at constant height. The actual ZMP with respect to *OXYZ* which is fixed at the center position between the two feet on the ground is calculated from the force/torque sensors integrated with the sole of the foot [7]. We assume that the robot is holding the home posture of figure 3 on the perfect horizontal floor of figure 2. Here, the measured ZMP should be located at a specific point which is calculated from the ideal model, but, in reality, the measured ZMP has a deviation from the specific point because of the accumulated absolute angle error of several joints. In other words, there exist offsets of the pelvis center point along the *X* and *Y* axes in the real environment with respect to the *OXYZ* frame. In this situation, ZMP regulation controller(ZRC) is useful in finding offsets in the transverse plane. By feedback from a measured ZMP using the F/T sensors, ZRC modifies the pelvis center position in the Cartesian space of *OXYZ*.

In order to compensate the deviation between the ideal and actual ZMP, we designed a ZMP regulation controller. The control objective is to regulate the measured ZMP at the ideal ZMP zero by changing the pelvis offset position while keeping its height constant. In order to satisfy this objective, many different kinds of online ZMP controller can be applied. For example, the controller using COG Jacobian [8], and torso position compliance control method [9] which tracks a given ZMP trajectory can be used for this objective. Here, we applied the following compensator.

$$X_{pelvis}^{offset\ ZRC} = -\frac{b_0 s + b_1}{s(a_0 s + a_1)} X_{ZMP}^{error} \tag{1}$$

$$Y_{pelvis}^{offset\ ZRC} = -\frac{b_0 s + b_1}{s(a_0 s + a_1)} Y_{ZMP}^{error} \tag{2}$$

$$X_{pelvis}^{adjusted} = X_{pelvis}^{original} + X_{pelvis}^{offset\ ZRC} \tag{3}$$

$$Y_{pelvis}^{adjusted} = Y_{pelvis}^{original} + Y_{pelvis}^{offset\ ZRC} \tag{4}$$

This controller has the characteristics of proportional-integral controller which guarantees the measured X_{ZMP} and Y_{ZMP} to converge into the reference X_{ZMP} and Y_{ZMP}. The authors derived a simple inverted pendulum model with a flexible joint as a double supported biped robot model [4].

$$\frac{Y\ or\ X_{ZMP}}{Y\ or\ X_{pelvis}} = \frac{K}{ml^2} \cdot \frac{1-(l/g)s^2}{s^2 + (K/ml^2 - g/l)} \tag{5}$$

where, *m* is the total point mass, *l* is the link length, *K* is the spring constant of the joint mainly due to the compliant force torque sensor, and *g* is the gravity. The coefficients of controller, a_0, a_1, b_0 and b_1 were determined through pole placement technique and experimental hand tuning. During adjustment, external forces and disturbances should not be applied to the robot. After finding the offsets $X_{pelvis}^{offset\ ZRC}$, $Y_{pelvis}^{offset\ ZRC}$ by means of the ZRC, and adding these offsets to the position command for the pelvis, the robot does not have an error between the measured ZMP and the theoretical one at the home posture.

B. Orientation Correction Controller (OCC)

When a robot holds a pose as shown in Fig. 1 on a flat floor perpendicular to gravity, the roll and pitch orientation of the torso may deviate from the ideal zero because of an inaccuracy of zero positions of joints. Therefore an orientation correction controller is introduced to correct the pitch and roll angle of the torso in both sagittal and frontal planes. In the sagittal plane, the controller corrects the pose by measuring the pitch angle with the inertial sensor measuring the direction of gravity and changing the pitch angle of the ankle during adjustment. We can use the pitch angle of the hip, but eventually the result will be same because other controllers are also activating. In the frontal plane, the orientation correction controller corrects the pose by measuring the roll angle of the torso and changing the relative leg length of the left and right legs. During this controller, the ZRC regulates ZMP as well. Here, the leg length means the length between hip joint and ankle joint along Z-axis. Fig. 4 shows the action of the orientation correction controller. At the pitch angle of the ankle, the orientation correction controller superimposes the control input of the PI controller using the torso pitch error $\theta_{pitch\ error}^{torso}$ on the original ankle position as follows:

$$\theta_{ankle\ pitch\ all}^{offset\ OCC} = -\left(K_P + \frac{K_I}{s}\right)\theta_{pitch\ error}^{torso} \tag{6}$$

$$\theta_{ankle\ pitch\ L}^{adjusted} = \theta_{ankle\ pitch\ L}^{original} + \theta_{ankle\ pitch\ all}^{offset\ OCC} \tag{7}$$

$$\theta_{ankle\ pitch\ R}^{adjusted} = \theta_{ankle\ pitch\ R}^{original} + \theta_{ankle\ pitch\ all}^{offset\ OCC} \tag{8}$$

where, K_P and K_I are the proportional and integral gains of PI controller and this control is performed on both left and right ankle joint during adjustment process.

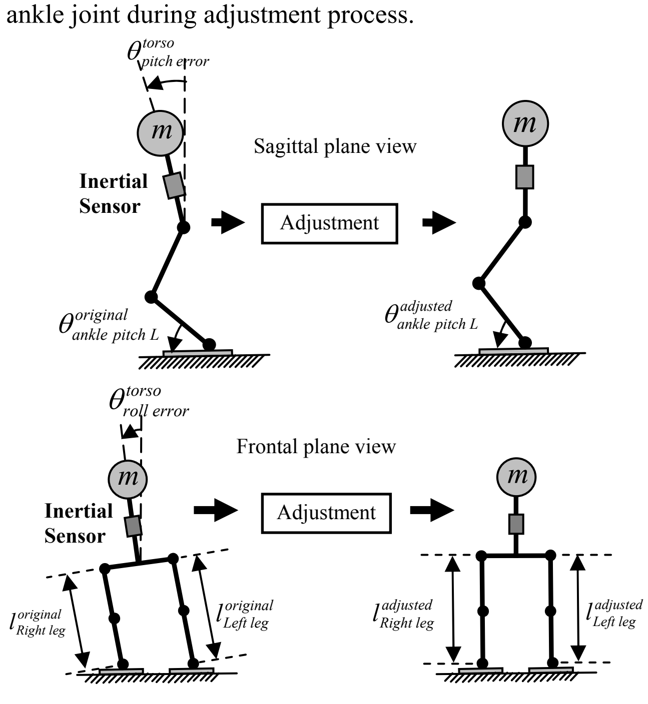

Fig. 4. Action of orientation correction controller

The roll angle of the robot is adjusted by changing the relative leg length. The orientation correction controller finds the offsets of the leg length by a PI feedback control rule with the torso roll error $\theta_{roll\ error}^{torso}$ as input. Then, the adjusted right leg length is calculated by adding this offset to the original leg length and the adjusted left leg length is calculated by subtracting the offset from the original leg length. The following equations represent the control law.

$$l_{Leg}^{offset\ OCC} = -\left(K_P + \frac{K_I}{s}\right)\theta_{roll\ error}^{torso} \tag{9}$$

$$l_{Right\ Leg}^{adjusted} = l_{Right\ Leg}^{original} + l_{Leg}^{offset\ OCC} \tag{10}$$

$$l_{Left\ Leg}^{adjusted} = l_{Left\ Leg}^{original} - l_{Leg}^{offset\ OCC} \tag{11}$$

where, K_P and K_I are the proportional and integral gains of PI controller. In this manner, we can make the offsets that enable the robot to keep its torso upright against the global inclination of the floor before operating walking.

C. Compliance Controller on the roll joint of the ankle (CC)

During adjustment of the orientation correction controller, position control of the roll joint of the two ankles is disabled in order to move freely against the external torque. In case of KHR (KAIST Humanoid Robot) series, the backdrivability of ankle joint is enough. This fact means that the torque of the roll joint on the ankle is nearly zero during adjustment and its backdrivability does not affect the convergence of the adjustment process. However, when the robot has a non-backdrivable roll joint on the ankle, roll joint cannot be moved by an external torque even though the position control is off. As a result of non-backdrivability, undesirable torque occurs between foot and ground along roll axis. In this case, it is necessary to actively control the ankle to make its torque to zero. This compliance control at the roll ankle joint is expressed as the following equation.

$$\begin{cases} \textit{for non-backdrivable joint}: \\ \qquad \theta_{ankle\ roll\ L}^{offset\ CC} = \dfrac{T_{roll\ L}}{Cs + K} \\ \textit{for backdrivable joint}: \text{in this case, position control is} \end{cases} \tag{12}$$

turned off and the measured encoder error is set to the offset.

$$\theta_{ankle\ roll\ L}^{offset\ CC} = \theta_{ankle\ roll\ L}^{Encoder\ Error\ from\ initial\ angle} \tag{13}$$

$$\theta_{ankle\ roll\ L}^{adjusted} = \theta_{ankle\ roll\ L}^{original} + \theta_{ankle\ roll\ L}^{offset\ CC} \tag{14}$$

The compliance control on the right ankle joint can be expressed in the same manner with an independent controller, where, $T_{roll\ L}$ is the measured torque at roll joint of left ankle, C and K are tuning parameters that create the satisfactory backdrivability around the roll joint of the ankle. For the adjustment process of Hubo FX-1[10], the above compliance control algorithm was indispensable since overall convergence was affected by the backdrivability.

D. Ankle Torque Difference Controller (ATDC)

During adjustment, there is a possibility that the feet of the robot do not fully contact the ground even though ZMP and pose converge to some reference value from the initial posture by means of ZMP regulation controller and orientation correction controller. This imperfect contact with the ground can be confirmed by observing the roll and pitch torques of the F/T sensors. As the roll ankle joints are free, we can assume that only pitch angle of the ankle occurs imperfect contact. Torques of the pitch ankle axes could be different on the two feet. The ankle torque difference controller makes the two torque values on the pitch ankles equal. The authors applied the PI controller which uses the torque difference between the two pitch ankle joints as an input. The equation of the ankle torque difference controller is expressed as follows.

$$\theta_{ankle\ pitch\ L}^{offset\ ATDC} = -\left(K_p + \frac{K_I}{s}\right)\cdot(T_{pitch\ R} - T_{pitch\ L}) \tag{15}$$

where, $T_{pitch\ R}$ and $T_{pitch\ L}$ is the measured torques at right and left pitch joints of ankles. Finally, adding the above offset to (7), (8) becomes following equations.

$$\theta_{ankle\ pitch\ L}^{adjusted} = \theta_{ankle\ pitch\ L}^{original} + \theta_{ankle\ pitch\ all}^{offset\ OCC} + \theta_{ankle\ pitch\ L}^{offset\ ATDC} \tag{16}$$

$$\theta_{ankle\ pitch\ R}^{adjusted} = \theta_{ankle\ pitch\ R}^{original} + \theta_{ankle\ pitch\ all}^{offset\ OCC} \tag{17}$$

It is important to note that the authors do not make both pitch torques zero. We only make the torque difference into zero so that the two ankle torques automatically converge to some value which corresponds to X_{ZMP}, the reference of ZMP regulation controller.

Summarizing the algorithms of section III, the proposed four controllers for the adjustment process are shown in figure 5. The robot system is a MIMO system and the measured input vector and output vector are as follows.

Input vector of adjustment :

$$\theta_{pitch\ error}^{torso}, \theta_{roll\ error}^{torso}, T_{roll\ L/R}, T_{pitch\ L/R}, X_{ZMP}^{error}, Y_{ZMP}^{error}$$

Output vector of adjestment :

$$X_{ZMP}^{offset\ ZRC}, Y_{ZMP}^{offset\ ZRC}, \theta_{ankle\ pitch\ L}^{offset\ ATDC},$$

$$\theta_{ankle\ pitch\ all}^{offset\ OCC}, l_{Leg}^{offset\ OCC}, \theta_{ankle\ roll\ L/R}^{offset\ CC}$$

When all the outputs have converged, the offsets for position, orientation and relative leg length are determined and robot joints are adjusted.

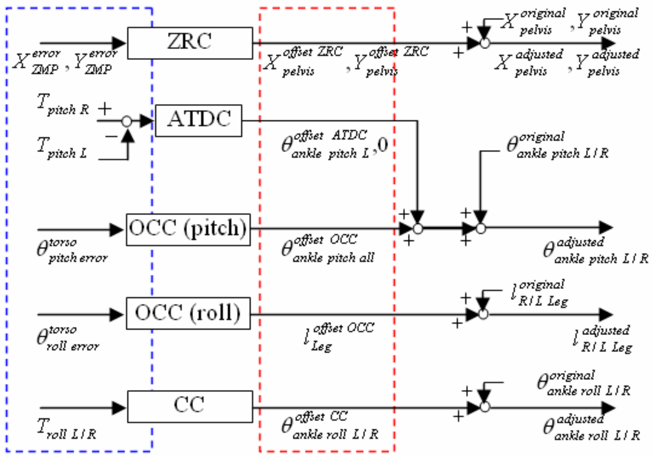

Fig. 5 Schematic diagram of adjustment algorithms

IV. Experiment

To demonstrate the effectiveness of our approach, we conducted the adjustment process through the use of KHR-2. Before the experiment, zero positions of each joints of the lower body were set by hand. By position control using inverse kinematics, the robot attained the home posture as shown in Fig. 1. First, we tested the adjustment algorithm by analyzing sensory data and control inputs. Second, we also tested the effectiveness of the adjustment through walking experiments using KHR-2.

A. Adjustment Algorithm Experiment

After the robot stands on the Balance Plate with the home posture, we activated the adjustment algorithm at 5.0 seconds. It is important to note that all controllers of the adjustment algorithm were activated simultaneously. As for the ZMP regulation controller, desired X_{ZMP} and Y_{ZMP} are +10 mm and 0 mm respectively. Fig. 6 shows the experimental results of the ZMP regulation controller. Since there always exists offset between actual pelvis center and ideal pelvis center, initial ZMP values in the graph are arbitrary. When we applied control, it was observed that the actual ZMP values measured by the force/torque sensor are regulated well. X and Y components of the pelvis center with respect to the ground fixed coordinate frame are changing continuously even though ZMP converged to desired value. This is because other controllers are working together.

Fig 7 shows experimental results of the orientation correction controller. Since, at the beginning, the robot's upper body was not adjusted, upper body is inclined a little. This small inclination can also have bad effects on the robot's stable walking. In general, if the robot walks stably, its inclination is within one degree. Therefore, even a small inclination of less than one degree may generate instability. In the graph, it is observed that the roll and pitch angle of the upper body gradually converges to zero when the controller is applied. Leg lengths and ankle pitch angles of both legs are changing slowly.

Fig. 8 shows the experimental results of the ankle torque difference controller. Since the robot stood on the ideal ground made by the Balance Plate, an initial difference of torque between both ankle pitch joints was produced due to the incorrect initial positions of ankle pitch joints. When the controller was applied, the torque difference became zero quickly. By using this controller, we can make soles parallel to the ideal ground with ease. If the pitch angles of both feet are different, the walking motion along the pitch axis is not balanced when it walks.

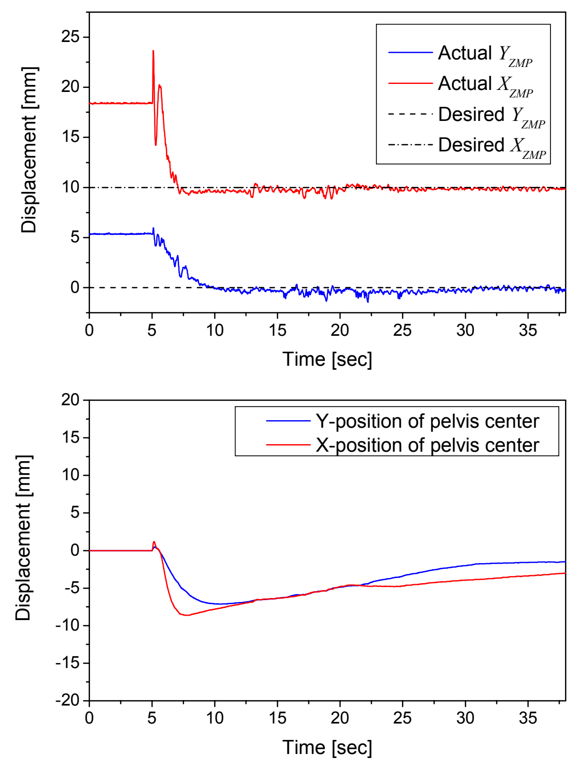

Fig. 6 Experimental results of ZMP regulation controller

B. Walking Experiments

We also conducted walking experiments by using KHR-2. As the biped walking control strategy, we used the algorithms that have been developed by the authors[4][5]. The objective of the experiment is to evaluate the walking performance before and after adjustment when the robot walks in place. Before adjustment, we set initial angles of the joints by hand. When the robot walks, we measured upper body inclination and ZMP by using the inertial sensor and force torque sensors. Fig. 9 shows the experimental results of walking in place before adjustment. Basically, upper body inclination is changing a lot, where the range of roll angle is from -2.5° to +2.3° and that of pitch angle is from 2° to 4.8°. In particular, neither initial pitch angle nor initial X_{ZMP} is near zero. These errors make the robot unstable during walking. Even though initial roll angle and initial Y_{ZMP} are well-balanced and near zero, they begin to be excited by the unstable pitch motion and other factors. Fig. 10 shows the experimental data when the KHR-2 is walking after the adjustment. As a result of the adjustment, the initial pitch and roll angle are almost zero, and initial X_{ZMP} and Y_{ZMP} are 10 mm and 0 mm respectively. While the robot walks in place, the roll angle of the torso changes from approximately -1° to 1°. The pitch angle is near zero, where the range is within ±0.3°. X_{ZMP} and Y_{ZMP} are also well-balanced within the stable range. The robot is walking very stably after adjustment.

In these experiments, we could confirm the proper working of the proposed adjustment algorithm and prove its effectiveness. The proposed algorithm has been successfully used in our KHR (KAIST Humanoid Robot Platform) series.

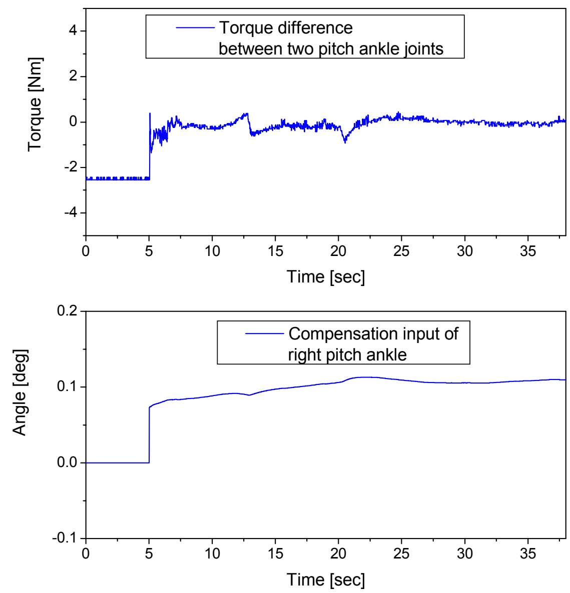

Fig. 8 Experimental results of ankle torque difference controller

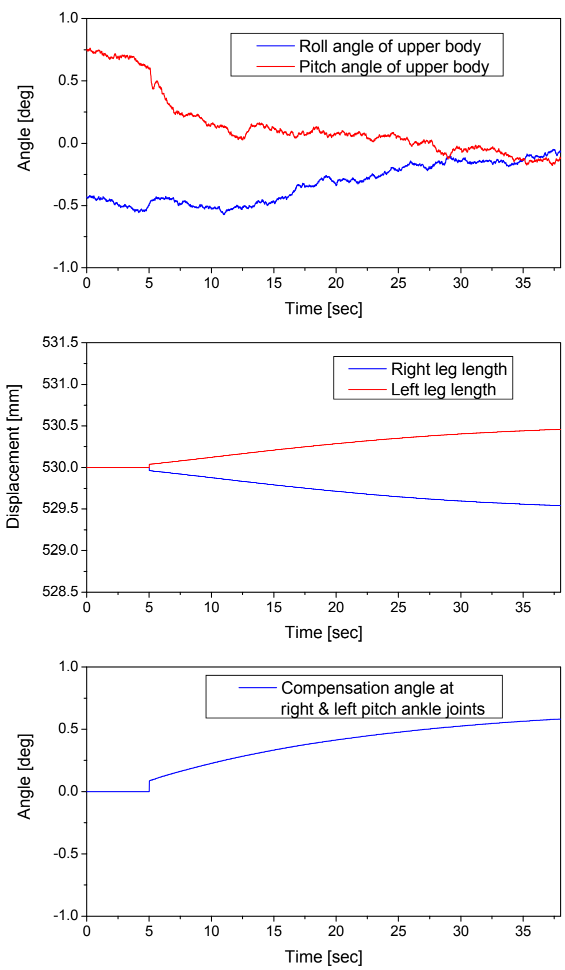

Fig. 7 Experimental results of orientation correction controller

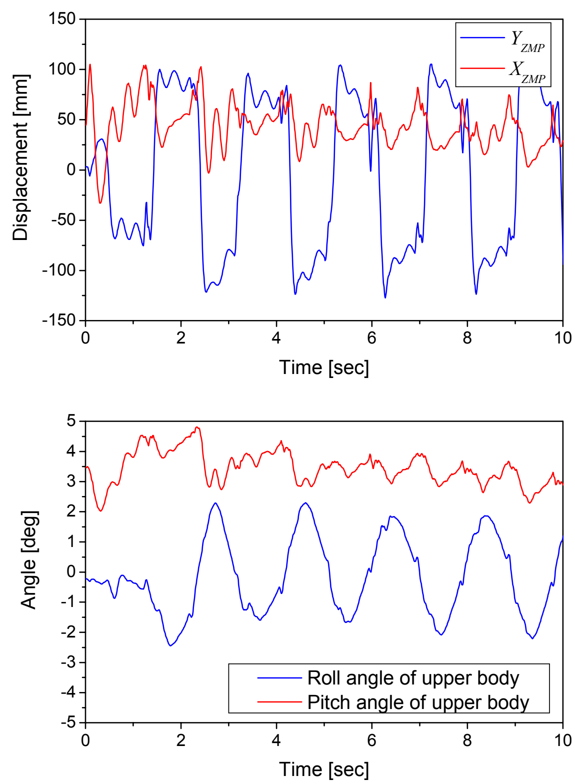

Fig. 9 Experimental results of walking in place before the adjustment

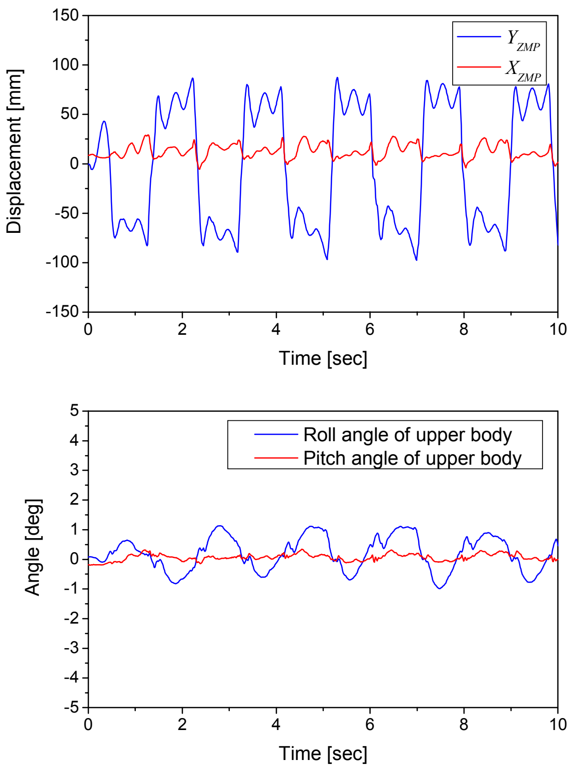

Fig. 10 Experimental results of walking in place after the adjustment

V. CONCLUSION

In this paper, we proposed an adjustment algorithm for the home posture of a biped humanoid robot. The algorithm is composed of the four controllers; ZMP regulation controller, orientation correction controller, compliance controller and ankle torque difference controller. Originally, this method was used for our first humanoid robot, KHR-1 in 2002, but it has not been made public until now. This algorithm is not complicated and very effective when we adjust the initial position, orientation and C.G point of robot based on the sensors before starting the operation. Though the adjustment process needs about one minute for convergence, it can guarantee a consistent initial posture before walking. Accordingly, the authors can always analyze and design biped walking controllers effectively with a constant initial posture. It can be thought that a consistent initial posture can be realized through the use of potentiometers or limit switches. However, there are many disadvantages: they complicate mechanical design, their resolutions are not very good, the voltage of a potentiometer drifts according to the body temperature or room temperature, and they must be fixed on the body strongly. Furthermore, there may be assembly and manufacturing tolerance in the robot. It is very difficult to resolve the accumulated tolerances into offsets of each joint. These explain why the authors developed the adjustment algorithm. Therefore, it is necessary to adjust the home posture of the robot by using sensory feedback so that the robot can correct its home posture automatically and precisely. As we can see the experimental results in this paper, our adjustment algorithm has been demonstrated on KHR-2.

At this moment, only the joints of lower body of the biped robot are adjusted. This is because the adjustment of the lower body is more important than the upper body in terms of accurate inverse kinematics during walking. In fact, the upper body pose affects the adjustment of the lower body. So, the development of an efficient adjustment algorithm for the upper body of a humanoid robot is future work.

REFERENCES

[1] K. Hirai, M. Hirose, Y. Haikawa, and T. Takenaka, “The development of honda humanoid robot,” *in Proc. of Int. Conf. on Robotics and Automations*, Leuven, 1998, pp. 1321-1326.

[2] J.-H. Kim, S.-W. Park, I.-W. Park, and J.-H. Oh, “Development of a Humanoid Biped Walking Robot Platform KHR-1 - Initial Design and Its Performance Evaluation,” *in Proc. of 3rd IARP Int. Workshop on Humanoid and Human Friendly Robotics*, Tsukuba, 2002, pp. 14-21.

[3] J.-H. Kim and Jun-Ho Oh, “Torque Feedback Control of the Humanoid Platform KHR-1,” *in Proc. of 3rd IEEE Int. Conf. on Humanoid Robots*, Karlsruhe and Munich, Germany, 2003.

[4] J.-H. Kim and J.-H. Oh, “Realization of Dynamic Walking for the Humanoid Platform KHR-1,” *Advanced Robotics*, vol. 18, no. 7, 2004, pp.749-768.

[5] J.-Y. Kim, I.-W. Park, J. Lee, M.-S. Kim, B.-K. Cho and J.-H. Oh, “System Design And Dynamic Walking Of Humanoid Robot KHR-2,” *in Proc. IEEE Int. Conf. on Robotics & Automation*, pp.1431-1436, 2005.

[6] I.-W. Park, J.-Y. Kim, J. Lee and J.-H. Oh, “Mechanical design of humanoid robot platform KHR-3 (KAIST Humanoid Robot-3: HUBO),” *in Proc. IEEE/RAS Int. Conf. on Humanoid Robots*, Tsukuba, 2005, pp. 321–326.

[7] M. Vukobratovic, B. Borovac, D. Surla, and D. Stokic, *Biped locomotion : dynamics, stability, control and application*, Springer-Verlag, 1990.

[8] T. Sugihara and Y. Nakamura, “ Whole-body cooperative balancing of humanoid robot using COG Jacobian,” *in Proc. IEEE Int. Conf. on Intelligent Robots and Systems*, 2002, pp. 2575–2580.

[9] K. Nagasaka, M. Inaba, and H. Inoue, “Stabilization of dynamic walk on a humanoid using torso position compliance control.” *In Proc. 17th Annual Conference on Robotics Society of Japan*, 1999, pp. 1193–1194 .

[10] J.-Y. Kim, J. Lee and J.-H. Oh, Experimental realization of dynamic walking for a human-riding biped robot, HUBO FX-1, *Advanced Robotics*, vol. 21, no. 3, 2007, pp. 461–484.