Proceedings of the 17th World Congress
The International Federation of Automatic Control
Seoul, Korea, July 6-11, 2008

Walking Pattern Generation for Planar Biped Walking Using Q-learning

Jungho Lee*, Jun Ho Oh**

*Humanoid Robot Research Center
Korea Advanced Institute of Science and Technology
335 GwahangnoYuseong-gu Daejeon 305-701
Korea (Tel: +82-42-869-5223; e-mail: jungho77@gmail.com).
** Korea Advanced Institute of Science and Technology
335 GwahangnoYuseong-gu Daejeon 305-701
Korea (Tel: +82-42-869-8903; e-mail: jhoh@kaist.ac.kr).

Abstract: In this research, a stable biped walking pattern is generated using reinforcement learning. The biped walking pattern is chosen as a simple third order polynomial. To complete the walking pattern, four boundary conditions are needed. Initial position and velocity and final position and velocity of the joint are selected as boundary conditions. In order to find the proper boundary condition value, a reinforcement learning algorithm is used. Also, desired motion or posture can be achieved using the initial and final position. The final velocity of the walking pattern is chosen as a learning parameter. To test the algorithm, a simulator that takes into consideration the reaction between the foot of the robot and the ground was developed. The algorithm is verified through a simulation.

1. INTRODUCTION

A humanoid robot is a robot that totally or partially resembles a human in shape and/or function. This kind of robot is different from industrial robots, which are normally used in factories and typically perform tasks repetitively. The humanoid robot may have a torso, head, arms and so on and may even be able to perform various functions using fingers. It may also have artificial intelligence, and thus can recognize objects or human faces and also can represent himself using gestures. For a humanoid robot, its method of locomotion is critical. Numerous movement methods have been considered, including the use of wheels and caterpillar, quadped, and hexapod walking methods based on motions of animals and insects. While these methods have respective strengths, it is preferable that humanoid robots used in human society be capable of biped, as humans walk on two legs and our environment is geared to biped walking. By employing biped motion, the humanoid robot will be able to navigate stairs easily, doorsill, and walk over stepping stones.

HUBO, the first humanoid robot in Korea, was developed by Oh and colleagues at KAIST in 2004. Its performance was later improved and new biped walking robots, Albert HUBO and HUBO FX-1, were developed. Albert HUBO is an android type robot and HUBO FX-1 is a human riding biped walking robot. These humanoid robots combine several biped walking methods for stable walking. For the walking strategy, first a walking pattern that is suitable for a given environment is designed, and then a ZMP feedback controller and other sub controllers are used to maintain its stability for a dynamically changeable environment. In other words, the robot follows the walking pattern that is designed for a given environment. In order to keep its stability for a slightly changed environment, it uses the ZMP feedback controller and other sub-controllers such as a posture controller and landing orientation controller. Many researchers use only a ZMP feedback controller; however, while stable walking can be maintained, it is difficult to generate a desired motion.

The key challenge in the existing method used by HUBO is finding proper parameters for designing or generating a stable walking pattern. It is difficult to find proper parameters, because they are influenced by many factors such as robot posture, ground conditions, and so on. Also, the walking mechanism is not fully understood, in other words, it is not clear how humans walk or run, and thus it is naturally hard to apply the mechanism of human walking and design the walking pattern. The existing HUBO finds these parameters through many experiments and a walking data analysis using the real system. This process is, however, difficult and time-consuming. Furthermore, because the unconfirmed walking pattern is tested using the real robot, there is an inherent risk of accident. This is the starting point of the present research.

2. RELATED WORK

Chew and A. Pratt simulated their biped walking robot, Spring Flamingo and M2, in the planar plane (two-dimensional simulation). A reinforcement leaning system was used as the main controller. They chose the following states: (a) Velocity of the hip in the forward direction (x-coordinate); (b) x-coordinate of the previous swing ankle measured with reference to the hip; and (c) step length. The next position of the swing foot was used as the action. They used a torque controller in each ankle to control the ankle joint torque. The ankle joint torque was limited to a certain stable value, and thus the robot could walk stably without considering the ZMP. However, because their goal was to

realize walking with constant speed, the posture of the robot was not considered.

Benbrahim and A. Franklin used a reinforcement learning system as both the main and sub controllers. To achieve dynamic walking of their planar robot, central and other peripheral controllers were used. The central controller used the experience of the peripheral controllers to learn an average control policy. Using several peripheral controllers, it was possible to generate various stable walking patterns. The main controller activated specific peripheral controllers, an approach that is suitable for specific situations. However, the architecture of the controller is very complex and this approach required many trials for the leaning and long convergence time.

Morimoto, Cheng, G. Atkeson, and Zeglin used a simple five link planar biped robot to test their reinforcement learning algorithm. The foot of each leg had a 'U' shape and there were no joints in the ankle and thus it moved in the manner of a passive walker. The goal of the learning system was to walk with constant speed and the states were: (a) velocity of the hip in the forward direction; and (b) forward direction distance between the hip and ankle. The reward was simply falling down or not and the action was the angle of the knee joint. This work concentrated on stable walking only and thus the posture of the robot was not considered.

3. WALKING PATTERN

Methods of designing the stable walking pattern can be categorized as follows. The first approach is the inverted pendulum model control method. In this method, a simple inverted pendulum, i.e., inverted pendulum model, is used as a biped walking model. Based on this model, the proper ZMP reference is generated and the ZMP feedback controller follows this reference. Because this method uses a simple inverted pendulum model, its structure is very simple. Furthermore, since it follows the ZMP reference for stable walking, stability is always guaranteed. However, it requires a proper ZMP reference and it is difficult to clearly and accurately define the relationship between the ZMP reference and the posture of the biped walking robot. Therefore, it is difficult to select the proper ZMP reference if the posture of the biped walking robot is considered in addition to stable walking. A pattern generator, which translates the ZMP reference to the walking pattern, is also needed. Fig. 3-1 shows a block diagram of the inverted pendulum model control method.

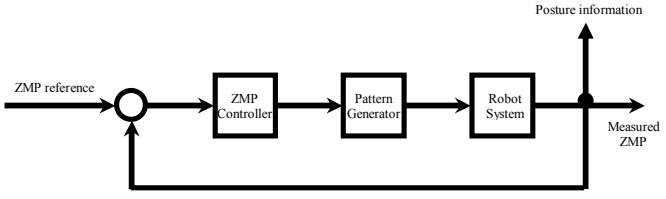

Fig. 3-1 Inverted pendulum model control method

The other method is called the accuracy model method. This model requires accuracy models of the biped walking robot and its environment. In this method, a stable walking pattern is generated in advance based on the accuracy model and the biped walking robot follows this walking pattern without a ZMP feedback controller. The strengths of this method are that it is possible to control the biped walking robot with the desired posture and it does not need a ZMP controller. However, the generated walking pattern is not a general walking pattern. For example, the walking pattern that is generated for flat ground is not suitable to inclined ground. Therefore, if the given environment is changed (e.g., ground condition, step length, step period, etc.) then the walking pattern should be regenerated for the changed environment.

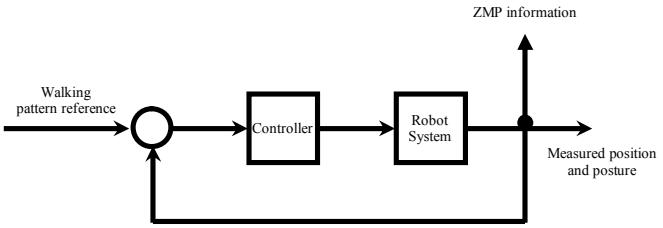

Fig. 3-2 Accuracy model method

An additional problem of this method is the difficulty in obtaining an accurate model of the robot and its environment, including such factors as the influence of the posture of the robot and the reaction force from the ground. Consequently, the generated walking pattern should be tuned by experiments. The generated walking pattern for the specific environment is sensitive to external forces, because this method does not include a ZMP controller. However, when the precise posture of the biped walking robot is required, for example, when moving upstairs or doorsill, this "accuracy model method" is very powerful. In order to resolve these problems, in the present work the walking pattern generating algorithm is developed using reinforcement learning. Fig. 3-2 shows the process of the accuracy model method.

To generate the walking pattern, first, the structure of the walking pattern should be selected carefully. Selection of the kind of structure is made based on polynomial equations, sine curves, etc. according to the requirements. In this research, a third order polynomial ankle and hip joint pattern for the support leg is designed as the walking pattern. To simplify the problem and also to make the body upright from the ground, the sum of the hip, knee, and ankle angles is set to be zero. The knee angle of the support leg is constant while walking, and thus the hip angle is not independent of the pattern of the ankle with respect to making the body upright. Thus, only the ankle joint or the hip joint pattern is required for the walking pattern of the support leg.

To create or complete the third order walking pattern, four boundary conditions are needed. These boundary conditions are chosen with a number of factors taken into account. First, to avoid jerking motions and make a smooth walking pattern, the walking pattern must be continuous. For this reason, the angle and angular velocity of the ankle joint at the moment of the beginning of the walking pattern of the support leg were chosen as the boundary conditions. Additionally, when the foot must be placed in a specific location, such as when stepping stones, the final position of the walking pattern is

important. This final position is related to the desired posture or step length, and this value is defined by the user. Hence, the final angle of the ankle can be another boundary condition. Finally, the final velocity of the walking pattern is utilized as the boundary condition. Using this final velocity, it is possible to modify the walking pattern shape without changing the final position and also stabilize the walking pattern. From these four boundary conditions, a third order polynomial walking pattern can be generated

Table 3-1 Boundary conditions for the walking pattern

Boundary condition	Reason
Initial velocity	To avoid jerk motion
Initial position	To avoid jerk motion and continuous motion
Final velocity	To make the walking pattern stable
Final position	To make wanted posture

However, it is difficult to choose the correct final velocity of the pattern. Because exact models include the biped robot, ground and other environmental factors are unknown. The existing HUBO robot uses a trial-and-error method to find the proper final velocity parameter, but numerous trials and experiments are required to tune the final velocity. Thus, in order to find a proper value for this parameter, the reinforcement leaning algorithm is used.

4. REINFORCEMENT LEARNING

The reinforcement learning agent uses the Q-learning algorithm, which uses the Q-value. To store the various Q-values, which represent actual experience or trained data, generalization methods are needed. Various generalization methods can be used; in the present work, the CMAC (Cerebellar Model Articulation Controller) is employed. This algorithm converges quickly and is readily applicable to real systems.

For the selection of proper states, the linear inverted pendulum model, which is normally used to model a biped walking robot, is considered. If the third order polynomial is used as the walking pattern, as mentioned previously, the ZMP equation can be written as shown in Fig. 4-1. As shown in Fig. 4-1, the body position and body acceleration are related to the ZMP position. If the ZMP position is located in the support region of the robot, the robot will then be dynamically stable. Therefore, choosing the body position and body acceleration as states is acceptable. In terms of energy efficiency, conserving the angular and linear momentum is important. The body velocity shows the direction of the movement of the body. Therefore, the body velocity can be another state. Table 4-1 shows the selected states and the related reasons for the selection of each state. All states are normalized to -1.0~1.0. However, the reinforcement learning agent has no data for the maximum values of the states; the reinforcement learning agent receives this data during the training and updates it automatically. First, these maximum values are set to be sufficiently small, in this case 0.1. The reinforcement learning agent then updates the maximum value at every step if the current values are larger than the maximum values.

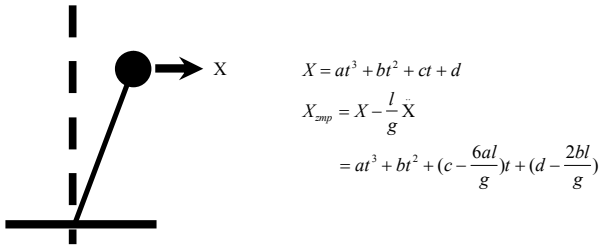

Fig. 4-1 The ZMP of the inverted pendulum

To create a third order polynomial walking pattern, the final velocity is needed, as discussed in Section 3. Hence, the final velocity is used as an action and other conditions are determined by the user. Table 4-2 shows the action and the reason for it. The maximum value of the action is limited to 0.3m/s. This maximum value is based on the physical motor and reduction gear specifications.

The reward function should be the correct criterion of the current action and also represents the goal of the reinforcement learning agent. The reinforcement learning agent should learn to determine a viable parameter value for the walking pattern generation; the goal is to have the robot walk stably. The reward is thus divided as 'fall down or not' and 'how good is it' in this research. Table 4-3 shows the reward and reasons. If the robot falls down, the reinforcement learning agent then gives a high negative value as the reward; in other cases, the robot receives positive values according to the body rotation angle. The body rotation angle represents the feasibility of the posture of the robot.

Table 4-1 States

State	Reason
Body position respect to the support foot	Relationship between the C.G. position and the ZMP and the body posture
Body velocity	Angular and linear momentum
Body velocity	Relationship between the C.G. position and the ZMP

Table 4-2 Action

Action	Reason
Final velocity of the walking pattern	Only the final velocity is the unknown parameter. It is related to stable walking.

Table 4-3 Reward function

Reward	Reason
Fall down or remain upright	This denotes the stability of the robot(or absence of stability)
Body rotation angle	It represents how good it is for stable dynamic walking

5. SIMULATOR

Because reinforcement learning is basically based on a trial-and-error method, it is both dangerous and difficult to apply it in actual systems before sufficient training is performed. In particular, when the system is inherently unstable, such as in the case of a biped walking robot, more attention is needed. Therefore, a learning agent should be fully trained in the

biped walking robot simulator and then applied to the actual robot.

The HUBO simulator is used to train a reinforcement learning agent, and hence its model is very important. The model used for the simulator should take into account the robot dynamics and the interaction between the robot and its environment model. Many researchers confuse robot dynamics and the interaction between the robot and the environment. Interactions are more important than the robot dynamics, because stability problems occur when the robot interacts with its environment, such as in the case of reaction force.

In this research, ODE (Open Dynamics Engine) was used to develop the robot model and the environment model such that it will accurately represent the real world. ODE is a physics engine initially developed by Smith. Its source code has been opened and is governed by an open source community. ODE provides libraries for dynamics analyses, including collision analyses. The performance of ODE has been validated by various research groups and many engineering programs use ODE as a physics engine.

The HUBO simulator, which was developed for this research, is composed of a learning system that is in charge of all leaning processes, a physics engine that models the biped robot and its environment, and utility functions to validate the simulation results. Fig. 5-1 shows these modules and the relationships between them.

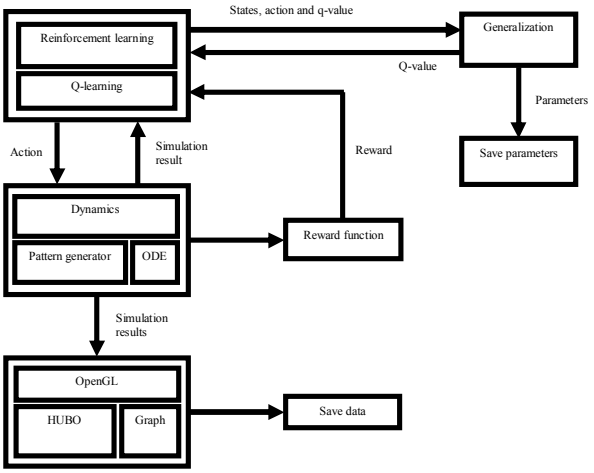

Figure 5-1. Structure of the HUBO simulator

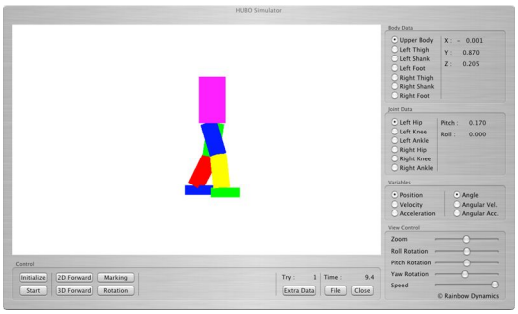

Figure 5-2 Layout of the HUBO simulator

6. EXPERIMENT

To test the performance of the walking pattern generation algorithm, specific motions are used, as shown in Table 6-1. The overall walking period is 0.9 sec. The support leg moves within 0.9 sec and the swing leg moves within 0.7 sec. The support leg and swing leg moving periods are different because it is necessary to reduce the reaction force from the ground when the swing leg touches the ground. The target step length is 0.358m and the target joint angle is shown in Table 6-1. Knee joints are fixed at 0.2rad and the overall walking speed is 1.432 km/h. HUBO is used as a simulation model in this experiment.

From Fig. 6-1, it is seen that the reinforcement learning agent converges after the 19th trial. After the 19th trial, the robot walks more than 400 steps and 120m. In the 10th trial, the robot succeeds in walking 38 steps but falls down (and therefore received punishment) in the next step. This can be considered as the local minimum. After the 19th trial, the states and action converge to specific vales and fall into a limit cycle.

Table 6-1 Simulation conditions

Step period	0.9 sec	
Step length	0.179+0.179=0.358 m	
Target motion of the front leg (0.9 sec)	Hip	-0.4 rad
	Knee	0.2 rad
	Ankle	0.2 rad
Target motion of the rear leg (0.7 sec)	Hip	0.2 rad
	Knee	0.2 rad

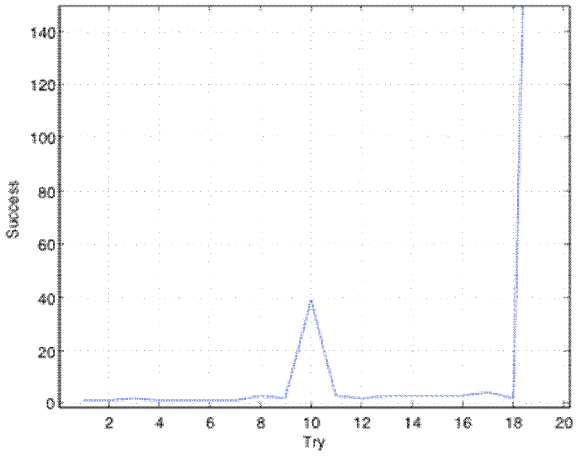

Figure 6-1 Iterations and number of successes

Fig. 6-2 and Fig. 6-3 show the body movements of the robot in the 19th trial. From these figures, it is seen that the robot walks stably and the walking sequence is repeated (limit cycle). The body moves up and down, because the knee angle of the support leg is fixed during walking. This motion is similar to passive walking. The center of the upper body is located 0.886m from the ground initially and at 0.86m during walking.

Fig. 6-4 shows the body rotation angle (pitch). The maximum value of the body rotation angle is 1.28 degrees. This value is reached when the support leg is changed from right to left or

from left to right. At this moment, the dynamic model is changed. This also shows that stability problems mostly occur at this moment.

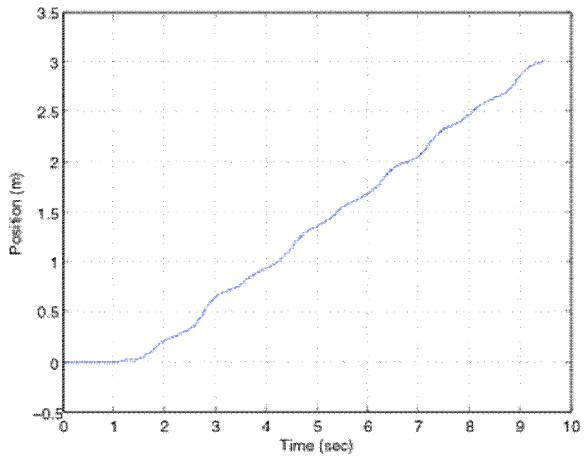

Fig. 6-2 Body movement (x-direction)

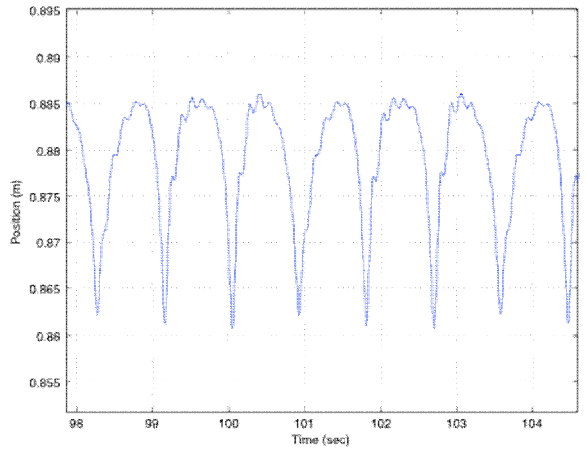

Fig. 6-3 Body movement (z-direction)

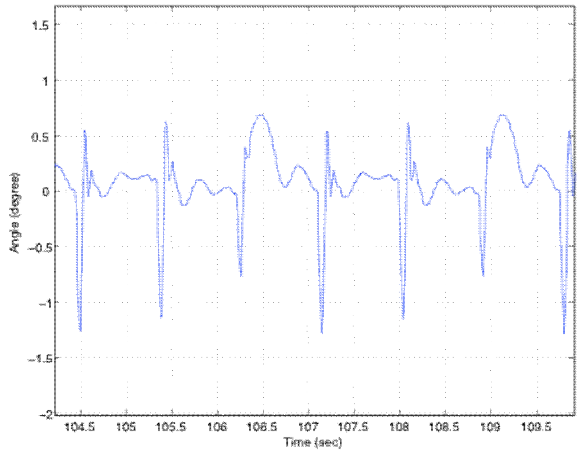

Fig. 6-4 Body rotation angle (pitch angle)

Fig. 6-5 and Fig. 6-6 show the position of the foot during the stable walking process. From Fig. 6-5, it is seen that the robot follows the given conditions, as outlined in Table 6-1. Its step length is 0.382m and the step period is 0.9 sec. This implies that the robot can walk stably and will place its foot in the desired position. Fig. 6-6 shows there is no DSP(double support phase) during walking.

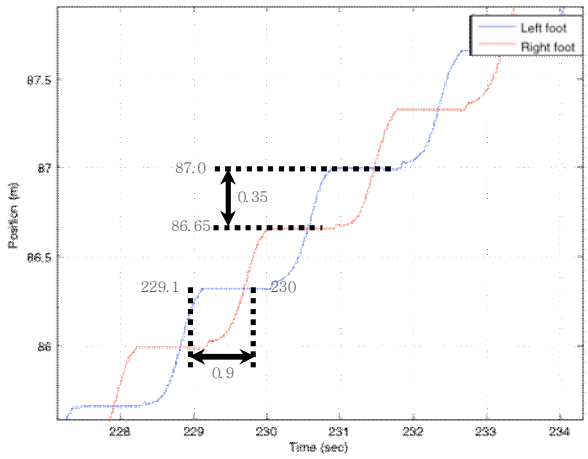

Fig. 6-5 Foot position (x-direction)

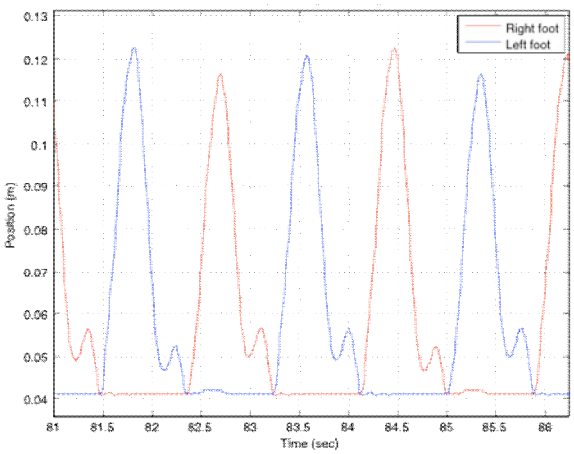

Fig. 6-6 Foot position (z-direction)

7. CONCLUSION

In this research, a learning system for a stable biped walking pattern generator was developed. The structure of the walking pattern is fixed as a third order polynomial expression and, on the basis of four boundary conditions, the initial position, initial velocity, final position, and final velocity of the walking pattern, it is possible to complete the walking pattern. Among the four boundary conditions, the initial position and initial velocity are selected to facilitate continuous and smooth walking without jerky motion. By using the final position as the boundary condition, the biped walking robot can step on a specific position. The other boundary conditions the final velocity of the walking pattern, is related to the stability of the robot. By changing this parameter, it is possible to realize a stable walking pattern. However, it is difficult to determine this parameter manually. Thus, by using the reinforcement learning system, the proper final velocity that allows for stable walking in a given environment is found. Using these boundary conditions, a stable walking pattern is generated whereby the robot can place its feet in specific positions.

All states, rewards, and actions for the learning system are chosen by physical insight and experience. Position, velocity, and acceleration of the upper body are related to ZMP. Hence, these states reflect the stability. Pitch angle of the upper body in relation to the ground is considered as the reward. There are many ways to describe the goal, which in this case is related to stability of the robot, such as the position of the upper body, the ZMP, the angle of the support foot or simply falling down or not. However, the pitch angle of the upper body is chosen by experience. Using this reward, the learning system had converged within only 19 trials. The action is selected to complete the stable walking pattern.

To train and verify the learning process, the HUBO simulator was developed given that it is difficult and dangerous to implement the learning system with a real robot without full learning. To make the simulation more realistic, a well known physics engine, the ODE, is used. Using the ODE, it is possible to simulate the robot and its environment. All features of the simulator are modulated and it is easy to add new components and algorithms. Specific motions were tested and verified using the HUBO simulator. The learning system learns the final velocity of the walking pattern and this walking pattern is verified using the simulator.

REFERENCES

A. Takanishi, M. Ishida, M. Yamazaki and I. Kato, *'The Realization of Dynamic Walking by the Biped Walking Robot WL-10RD'*, ICRA 1985, 1985.

C. Angulo, R. Tellez and D. Pardo, *'Emergent Walking Behaviour in an Aibo Robot'*, ERCIM News 64, p38-p39, 2006.

Chew-Meng Chew and Gill A. Pratt, *'Dynamic Bipedal Walking Assisted by Learning'*, Robotica, Volume 20, p477-p491, 2002

Dusko Katic and Miomir Vukobratovic, *'Control Algorithm for Biped Walking Using Reinforcement Learning'*, 2nd Serbian-Hungarian Joint Symposium on Intelligent Systems, 2004.

Hamid Benbrahim and Judy A. Franklin, *'Biped Dynamic Walking Using Reinforcement Learning'*, Robotics And Autonomous Systems, Volume 22, p283-p302, 1997.

Ill-Woo Park, Jung-Yup Kim, Jungho Lee, and Jun-Ho Oh, *'Mechanical Design of the Humanoid Robot Platform, HUBO'*, Journal of Advanced Robotics, Vol. 21, No. 11, 2007.

Ill-Woo Park, Jung-Yup Kim and Jun-Ho Oh, *'Online Walking Pattern Generation and Its Application to a Biped Humanoid Robot-KHR-3(HUBO)'*, Journal of Advanced Robotics, 2007.

Jun-Ho Oh, David Hanson, Won-Sup Kim, Il Young Han, Jung-Yup Kim, and Ill-Woo Park, *'Design of Android type Humanoid Robot Albert HUBO'*, in Proc. IEEE/RSJ Int. Conf. on Intelligent Robots and Systems, Beijin, China, 2006.

Jun Morimoto, Gordon Cheng, Christopher Atkeson and Garth Zeglin, *'A Simple Reinforcement Learning Algorithm for Biped Walking'*, Proc. of the 2004 International Conference on Robotics & Automation, p3030-p3035, 2004.

Jung-Yup Kim, Ill-Woo Park, and Jun-Ho Oh, *'Walking Control Algorithm of Biped Humanoid Robot on Uneven and Inclined Floor'*, Journal of Intelligent and Robotic Systems, Accepted, 2006.

Jung-Yup Kim, *'On the Stable Dynamic Walking of Biped Humanoid Robots'*, Ph. D Thesis, Korea Advanced Institute of Science and Technology, 2006.

Jung-Yup Kim, Jungho Lee and Jun Ho Oh, *'Experimental Realization of Dynamic Walking for the Human-Riding Biped Robot, HUBO FX-1'*, Advanced Robotics, Volume 21, No. 3-4, p461-p484, 2007.

Jungho Lee, Jung-Yup Kim, Ill-Woo Park, Baek-Kyu Cho, Min-Su Kim, Inhyeok Kim and Jun Ho Oh, *'Development of a Human-Riding Humanoid Robot HUBO FX-1'*, SICE-ICCAS 2006, 2006.

K. Hirai, *'Current and Future Perspective of Honda Humanoid Robot'*, Proc. IEEE/RSJ International Conference on Intelligent Robots and Systems, p500-p508, 1997

K. Kaneko, S. Kajita, F. Kanehiro, K. Yokoi, K. Fujiwara, H. Hirukawa, T. Kawasaki, M. Hirata and T. Isozumi, *'Design of Advanced Leg Module for Humanoid Robot Project of METI'*, Proc. IEEE International Conference on Robotics and Automation, p38-p45, 2002.

L.Holh, R. Tellez, O. Michel and A. Ijspeert, *'Aibo and Webots: simulation, wireless remote control and controller transfer'*, Robotics and Autonomous Systems, Volume 54, Issue 6, p472-p485, 2006.

Naoto Shiraga, Seiichi Ozawa and Shigeo Abe, *A Reinforcement Leaning Algorithm for Neural Networks with Incremental Learning Ability'*, Proceedings of International Conference Neural Information Processing, 2002.

Russel Smith, *'www.ode.org/ode.html'*, 2007.

Shuuji Kajita, Fumio Kanehiro, Kenji Kaneko, Kiyoshi Fujiwara, Kensuke Harada, Kazuhito Yokoi and Hirohisa Hiukawa, *'Biped Walking Pattern Generation by Using Preview Control of Zero-Moment Point'*, Proceedings of the 2003 IEEE International Conference on Robotics & Automation, p1620-p1626, 2003.

William Donard Smart, *'Making Reinforcement Learning Work on Real robots'*, Ph. D. Thesis, Brown University, 2002.

Y. Sakagami, R. Watanabe, C. Aoyama, S. Matsunaga, N. Higaki and K. Fujimura, *'The Intelligent ASIMO: System Overview and Integration'*, Proc. IEEE/RSJ International Conference on Intelligent Robots and Systems, p2478-p2483, 2002.