2011 IEEE/RSJ International Conference on
Intelligent Robots and Systems
September 25-30, 2011. San Francisco, CA, USA

Practical Bipedal Walking Control on Uneven Terrain Using Surface Learning and Push Recovery

Seung-Joon Yi*†, Byoung-Tak Zhang†, Dennis Hong‡ and Daniel D. Lee*

Abstract— **Bipedal walking in human environments is made difficult by the unevenness of the terrain and by external disturbances. Most approaches to bipedal walking in such environments either rely upon a precise model of the surface or special hardware designed for uneven terrain. In this paper, we present an alternative approach to stabilize the walking of an inexpensive, commercially-available, position-controlled humanoid robot in difficult environments. We use electrically compliant swing foot dynamics and onboard sensors to estimate the inclination of the local surface, and use a online learning algorithm to learn an adaptive surface model. Perturbations due to external disturbances or model errors are rejected by a hierarchical push recovery controller, which modulates three biomechanically motivated push recovery controllers according to the current estimated state. We use a physically realistic simulation with an articulated robot model and reinforcement learning algorithm to train the push recovery controller, and implement the learned controller on a commercial DARwIn-OP small humanoid robot. Experimental results show that this combined approach enables the robot to walk over unknown, uneven surfaces without falling down.**

Keywords: Bipedal Walking, Uneven Terrain, Surface Model Learning, Full Body Push Recovery, Reinforcement Learning

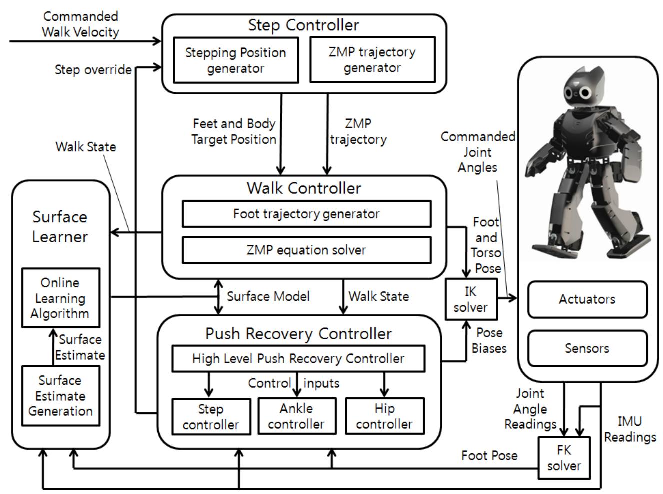


Fig. 1. Overview of the integrated walk controller with surface learning and push recovery controllers

I. INTRODUCTION

Dynamic bipedal walking is not very stable in nature, and open loop walking controllers tend to fail due to two sources of error: surface unevenness and external disturbances. Most walking controller implementations for humanoid robots assume perfectly flat surfaces, and even a minute surface inclination typically present in indoor environment can make the walking unbalanced and fall down eventually. And bipedal walking is also susceptible to external disturbances from various sources, including local surface unevenness, surface compliance, model error or contact with other objects. So it is crucial to handle these two problems to make a walking humanoid robot to operate in realistic environments.

Thus, these two problems have been a topic of major interest in humanoid robot research. The first problem, legged locomotion over uneven terrain, has been studied over a number of researchers [1], [2], [3], [4], [5], [6], [7], [8], [9], [10], [11]. One branch of approaches relies on precise models of surface either given a priori or built by sensors such as laser scanners, which may have some limitations in noncontrolled environments [1], [2], [3], [4], [5]. There have been other approaches which use sensory feedback to adapt the walk controller parameters to handle uneven surfaces [6], [7], [8], [9], [10], [11].

The second problem of push recovery been addressed by force-based approaches which uses force sensors and full body model of the robot to calculate the joint torques required to reject external disturbance force [12], [13] and biomechanically motivated push recovery approaches which focus on simple biomechanically motivated push recovery behavior based on a simplified model of the robot [14], [15], [16], [17], [18], [19].

A common drawback of these approaches is most physical implementations of them require specialized hardware such as triaxis force sensors and torque controlled joints, in addition to very fast feedback control and high processing power, which makes the implementation of them on commercially-available position-controlled humanoid robots like Nao[1] or DARwIn-OP[2] infeasible. In our former works, we have suggested practical methods to handle each problems for generic humanoid robots [20], [21]. However they are not integrated as a single controller to handle both problems at once, as are many former approaches.

In this work, our aim is to design an integrated controller for generic position-controlled humanoid robots without force sensors that is able to walk over unknown, uneven

* GRASP Laboratory, University of Pennsylvania, Philadelphia, PA 19104 {yiseung,ddlee}@seas.upenn.edu
† BI Laboratory, Seoul National University, Seoul, Korea. btzhang@ceas.snu.ac.kr
‡ RoMeLa Laboratory, Virginia Tech, Blacksburg, VA 24061 dhong@vt.edu

[1] http://www.aldebaran-robotics.com/
[2] http://www.robotis.com/xe/darwin_en

terrain, and reject external disturbance by a full body push recovery control. For the first problem, we use an electrically compliant swing leg and extended foot trajectory to make the swing foot adapt to local unevenness, and use inertial and proprioceptive sensors to estimate the height and gradient of the foot landing position. Then we use a online learning algorithm to quickly learn the explicit surface model the robot is walking on. For the second problem, we use a hierarchical push recovery controller which generates a full body push recovery behavior based on biomechanically motivated primitive push recovery behaviors to reject perturbations due to local surface unevenness or modeling error. Situationally aware push recovery behavior is learned in a simulated environment using reinforcement learning algorithm and multi body model of the robot. Our walk controller is trained and validated in simulated environment, and finally implemented on a commercially available humanoid robot platform, the DARwIn-OP robot. Experimental results show that our method can make the robot successfully walk over unknown and uneven terrain.

This paper is structured as follows. Section II describes our former works including a walk controller for uneven terrain and a push recovery controller. Section III explains the integrated walk controller in detail. Section IV addresses the simulation setup we use and shows the simulation results. Section V shows the experimental results from the physical DARwIn-OP robot. Finally, we conclude with some potential issues and future directions arising from this work.

II. Prior Work

In this section, we briefly introduce our prior work which addresses the walk controller for uneven terrain and the full body push recovery controller for a generic humanoid robot with limited sensory, actuational and processing capabilities.

A. Walk Control over Uneven Terrain

When humans walk on an unknown terrain without visual feedback, they feel the height and inclination of current foot landing position at every footfall using proprioception and force feedback, and update their belief on the surface based on that information.

In [20], this process is implemented on a generic humanoid robot with electrical compliance control, joint angle and inertial feedback. During the landing phase of walk cycle, the electrical compliance of swing ankle is lowered so that the landing foot can adapt to the local height and inclination of the surface. Leg movement is stopped after touchdown, and the swing leg stiffens and sensory information from joint angle encoders and inertial sensors are used to calculate the feet poses using the forward kinematic model of the robot.

After getting the point estimates of the local surface model at the landing position, an online learning algorithm is used to learn the surface model from these noisy estimates, and the walk trajectory is modified to take the current surface model into account. More details on actual implementation follow in the next section.

B. Hierarchical Push Recovery Control

Biomechanical studies show that humans display three distinctive motion patterns in response to sudden external perturbation, which are called ankle, hip and step push recovery strategies [15]. Although there have been theoretical analysis of these strategies using simplified models, physical implementation of such analytical controllers on a position controlled, generic humanoid robot is not straightforward, and there have been little research on how to combine three strategies appropriately as humans do.

Instead of relying on an analytical controller, we have suggested in [21] a machine learning approach to learn the appropriate push recovery controller from experiences to maximize a predetermined cost function. To generate a combination of push recovery behaviors, we use a hierarchical approach where three push recovery controller for each strategy are controlled by a high level controller based on current proprioceptive and inertial sensory information.

III. Integrated Walk Controller with Surface Learning and Push Recovery Control

In this section, we describe the details of our integrated walk controller for generic, position-controlled humanoid robots to walk on an uneven, unknown surface with external disturbance. It consists of the step controller, walk controller, push recovery controller and surface learner. Details of each component are addressed in following subsections.

A. Step Controller

The step controller determines the end positions of torso and feet for each step using current feet configuration and commanded walk velocity. This step-based walk control structure is used because it allows changing support foot and foot landing position at every step, which is helpful for push recovery control. We assume that each step starts and ends at the double support stance where the center of mass (COM) lies in the middle of two support positions, and define the zero moment point (ZMP) trajectory $p(\phi)$ as the following piecewise-linear function

$$p(\phi) = \begin{cases} p_0(1-\frac{\phi}{\phi_1}) + p_1\frac{\phi}{\phi_1} & 0 \le \phi < \phi_1 \\ p_1 & \phi_1 \le \phi < \phi_2 \\ p_2(1-\frac{1-\phi}{1-\phi_2}) + p_1\frac{1-\phi}{1-\phi_2} & \phi_2 \le \phi < 1 \end{cases} \quad (1)$$

where ϕ is the walk phase, p_1 is the support position for single support phase, p_0, p_2 are initial and final support positions and ϕ_1, ϕ_2 are timing parameters.

B. Walk Controller

The walk controller generates foot and torso trajectories from the initial and target foot and body position from the step controller and current surface model. To account for unevenness of the terrain, it maintains two disturbance variables $\vec{d_L}$ and $\vec{d_R}$ for each foot to account for local unevenness, and uses the learned surface model for account for global unevenness.

The orientation of the swinging foot starts with the current disturbance variable $\vec{d_L}$ or $\vec{d_R}$ which was measured at last

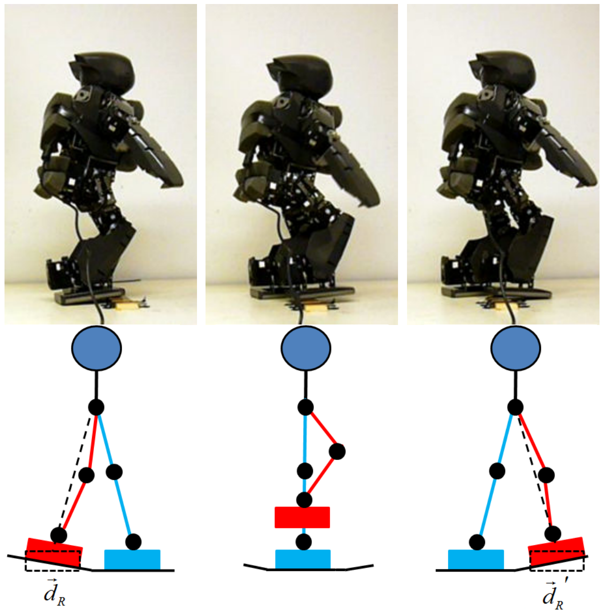


Fig. 2. Walk controller for uneven terrain. (a) The swing foot starts at ground position with distance variable $\vec{d}_R$. (b) The foot pose follows the computed trajectory during the early phase of the step, and the swing leg becomes compliant during the latter part of the step. (c) After the footfall, the swing leg stiffens and new disturbance variable $\vec{d}_R{}'$ is measured to keep the torso upright at the next step.

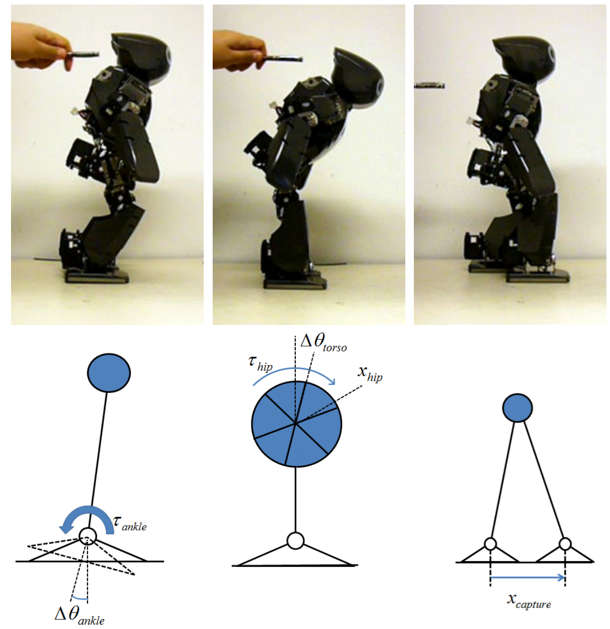


Fig. 3. Three biomechanically motivated push recovery strategies and corresponding controller for position-controlled robot based on simplified model. (a) Ankle strategy that applies control torque on ankle joints. (b) Hip strategy which uses angular acceleration of torso and limbs to apply counteractive ground reaction force (c) Step strategy which moves center of pressure to a new position.

footfall, and goes to level when the walk phase approaches the middle. Then electrical compliance of the swing leg is lowered so that landing foot can adapt to local terrain unevenness there so that the robot can measure a new disturbance variable for that foot. Figure 2 shows the process of stepping on uneven terrain. We do not use foot pressure sensor for touchdown detection as our DARwIn-OP robot is not equipped with one. Instead, we let the push recovery controller reject the perturbation due to landing timing error.

Torso trajectory is calculated according to the ZMP criterion and linear inverted pendulum model. The piecewise linear ZMP trajectory we use yields an analytic solution for torso trajectory with zero ZMP error during the step period. We have found that the discontinuous torso jerk at transitions does not hamper the stability much, as the transition happens in the most stable double support stance.

C. Surface Learner

After each footfall, we get two noisy estimates for surface unevenness using proprioceptive and inertial sensors: the positional difference between two feet $\vec{f}_d$ and the normal vector of the landed foot $\hat{f}_n$. If we use the $\hat{N}$, the global surface normal, as the surface model to learn, we can update the surface model to minimize following cost functions

$$C_{fd} = \quad \|\vec{f}_d \cdot N_{new}^{\hat{f}d}\|^2 + \alpha \|N_{new}^{\hat{f}d} - N^{\hat{f}d}\|^2 \tag{2}$$

$$C_{fn} = \quad \|N_{new}^{\hat{f}n} - \hat{f}_n\|^2 + \alpha \|N_{new}^{\hat{f}n} - N^{\hat{f}n}\|^2 \tag{3}$$

which results in following update rules

$$N_{new}^{\hat{f}d} = \quad N^{\hat{f}d} + \alpha''(\vec{f}_d \cdot N^{\hat{f}d})\vec{f}_d \tag{4}$$

$$N_{new}^{\hat{f}n} = \quad (1-\alpha')N^{\hat{f}n} + \alpha' \hat{f}_n \tag{5}$$

where N^{fd}, N^{fn} are surface models learned using foot displacement and foot normal estimates, and α', α'' are learning rates. As both estimates have relative advantage due to the foot shape and the configuration of the feet, we maintain two models and combine the two into one using weighted sum:

$$\hat{N} = W_1 N^{\hat{f}n} + W_2 N^{\hat{f}d} \tag{6}$$

where W_1 and W_2 can be determined from cross validation.

D. Push Recovery Controller

The push recovery controller is used to reject perturbations due to modeling error or external disturbances. It is even more needed for the walk controller on an uneven surface, as uneven terrain will induce perturbations due to landing pose and timing error at every footfall.

Our hierarchical push recovery controller consists of three low level biomechanically motivated push recovery controllers, which are shown in Figure 3, and a high level controller that controls low level controllers to generate a full body push recovery behavior.

1) Ankle Controller: The ankle controller applies control torque on ankle joints to keep the center of mass within the base of support. For generic robots with position controlled actuators, controlling the ankle torque can be done by either controlling the auxiliary ZMP which accelerates the torso to apply effective torque at ankle [20], [22] or modulating the target angle of the ankle servo. In this work we use the latter method, which is found to be effective for the DARwIn-OP robot we use. Then the controller can be simply implemented as

$$\Delta\theta_{ankle} = \quad x_{ankle} \tag{7}$$

where $\Delta\theta_{ankle}$ is the joint angle bias and x_{ankle} is the input of ankle controller. In addition to ankle angle, we also modulate arm position to apply effective torque at the ankle joint in a similar way, unless overridden by the hip controller.

2) Hip Controller: The hip controller uses angular acceleration of the torso and limbs to generate backward ground reaction force to pull the center of mass back towards the base of support. To perform maximum work, a bang-bang profile of the following form can be used

$$\tau_{hip}(t) = \tau_{max}u(t) - 2\tau_{max}u(t-T_{R1}) + \tau_{max}u(t-T_{R2}) \tag{8}$$

where $u(t)$ is the unit step function, τ_{max} is the maximum torque that the joint can apply, T_{R1} is the time the torso stops accelerating and T_{R2} is the time torso comes to a stop. After T_{R2}, the torso angle should return to initial position. This two-stage control scheme can be approximated for a position controlled actuator as

$$\Delta\theta_{torso} = \begin{cases} x_{hip} & 0 \le t < T_{R2} \\ x_{hip}\frac{T_{R3}-t}{T_{R3}-T_{R2}} & T_{R2} \le t < T_{R3} \end{cases} \tag{9}$$

where $\Delta\theta_{torso}$ is the torso pose bias, T_{R3} the time torso angle completes returning to initial position, and x_{hip} is the input of hip controller. Arm rotation is also used for hip push recovery, which overrides arm movement control by ankle controller.

3) Step Controller: The step controller effectively moves the base of support by taking a step. This is implemented by overriding the step controller to insert a new step with relative target foot position $x_{capture}$. To get rid of the situation where the robot tries to lift the current pivoting foot and lose balance, the support foot is determined according to the current feet configuration and the direction of perturbation.

4) High Level Controller: The high level controller controls three low level push recovery controllers using sensory feedback from onboard sensors, learned surface model and current states of controllers. As there is no analytical controller for individual low level controllers using the raw sensory values, let alone the combination of three controllers, we use a machine-learning approach that trains the controller to optimize a cost function from experience. We formalize the high level controller as a reinforcement learning problem with the following state

$$S = \left\{\theta_{IMU}, \theta_{gyro}, \theta_{foot}, WS, HCS, \hat{N}\right\} \tag{10}$$

where θ_{IMU} and θ_{gyro} are torso pose and gyroscope data from inertial sensor, θ_{foot} is the support foot pose calculated using forward kinematics and onboard sensors, WS is the walk state and HCS is the hip controller state. Then the action is defined as the joint input for three low level controllers

$$A = \left\{x_{ankle}, x_{hip}, x_{capture}\right\} \tag{11}$$

and reward is defined as

$$R = \left|\theta_{gyro}\right|^2 + \frac{g}{z_0}\left|\theta_{IMU}\right|^2 \tag{12}$$

where g is the gravitational constant and z_0 is the COM height.

IV. Simulation Results

We first use a physically realistic computer simulation to train and validate the suggested walk controller. In this section we discuss the details of simulation setup and its results.

A. Simulation Setup

Our open source simulation environment consists of the Open Dynamics Engine (ODE) with Matlab-based controllers and graphics routine. This solution provide us full controllability, observability and repeatability which is helpful for machine-learning tasks which require repeated trials from the same initial state. The multi-body model of the robot is based on actual physical property of DARwIn-OP robot, and each servomotor is modeled as a joint controlled by a high-gain p-controller. The update frequency of walk controller is set to 100Hz to match that of actual robot, and a time step of $0.0001s$ is used for the physics simulation.

B. Learning Setup

We used the stochastic policy gradient reinforcement learning algorithm which randomly generates a number of test policies around the currently best policy to get the gradient at the point and uses stochastic gradient descent to improve the policy. To accelerate the learning process, we use simpler parameterized policy functions than [21], which consists of a linear function over each inputs with dead-band, gain, saturation parameters for ankle push recovery controller, and a step function with magnitude and threshold parameters for the hip and step push recovery controller. Each trial lasts for 2 seconds, and 20 trials are done at each episode.

C. Simulation Result

Figure 4 shows the response of the learned push recovery controller to perturbation on inclined and declined surfaces. We can see the push recovery controller correctly initiates appropriate full body push recovery behaviors to stabilize the robot on uneven surface.

We also let the robot walk over a test surface with changing inclination (8%, 0% and -8%) and local unevenness modeled by 40 randomly placed blocks with 70×70×1 mm size, which create more than 3% of local unevenness in the worst case. Figure 5 shows the comparison of walking trials over the test surface. Figure 5 (a) shows the walk controller without surface learning, where the combined effect of surface inclination and local unevenness is beyond

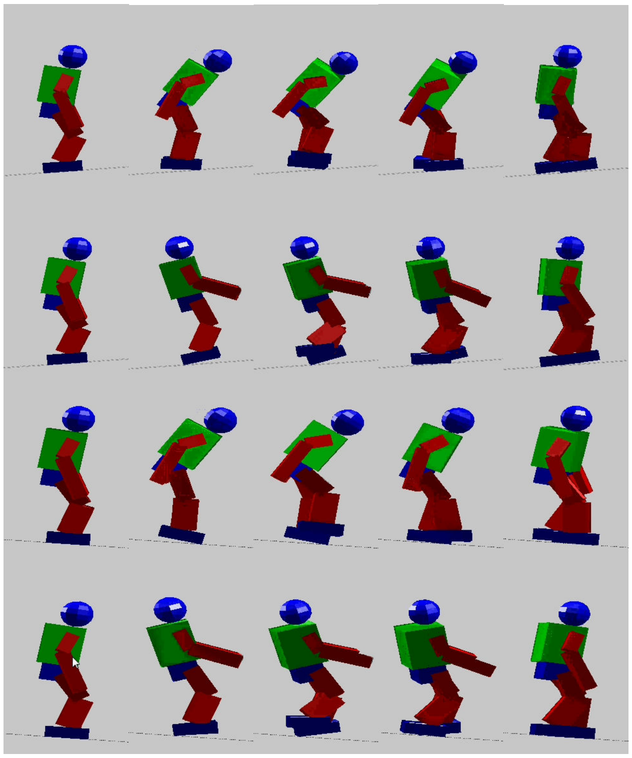
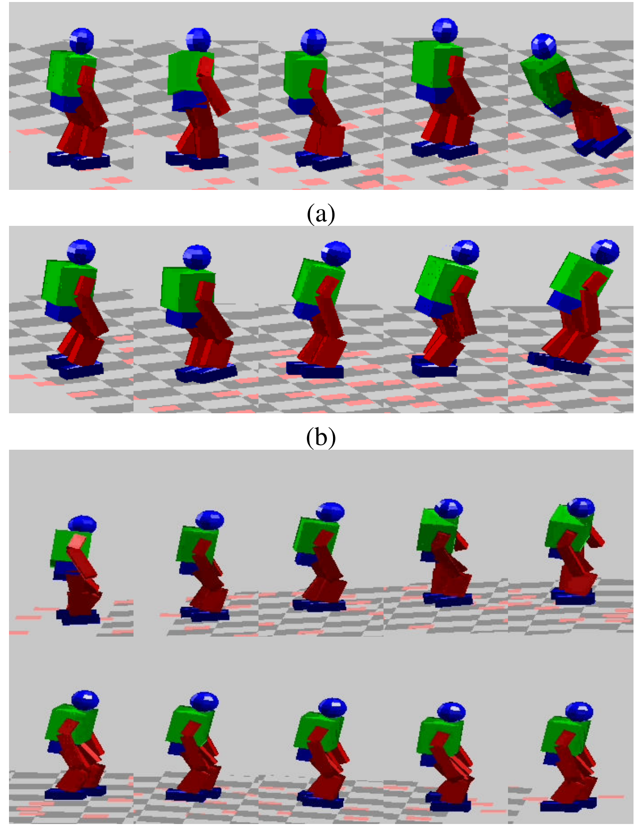


Fig. 4. Responses of learned push recovery controller for external perturbation on inclined and declined surface.



(a)

(b)

(c)

Fig. 5. Comparison of walking trials on uneven terrain. (a) walk controller with push recovery controller (b) walk controller with surface learning (c) walk controller with both surface learning and push recovery control.

the ability of the push recovery controller. Figure 5 (b) shows the walk controller without push recovery controller, where the robot loses balance as it walks over local unevenness and finally falls down. On the other hand, Figure 5 (c) shows the walk controller with both push recovery control and surface learning. The robot rejects disturbance due to local unevenness and sudden inclination change using full body push recovery control, and it successfully walks over the test surface without falling down.

V. Experimental Results

After training the push recovery controller in the simulated environment, we use a commercially available, open-source DARwIn-OP robot to validate the walk controller with learned push recovery controller.

A. Experimental Setup

We use the commercially available DARwIn-OP humanoid robot developed by Robotis Co., Ltd. and the RoMeLa lab. It is 45 cm tall, weighs 2.8kg, and has 20 degrees of freedom. It has a web camera for visual feedback, and 3-axis accelerometer and 3-axis gyroscope for inertial sensing. Position-controlled Dynamixel servos are used for actuators, which are controlled by a custom microcontroller connected by an Intel Atom-based embedded PC at a control frequency of 100hz. The same walk parameter sets are used as the simulated environment, except for longer double support ratio to compensate for non-ideal inertial sensor readings.

B. Results

Figure 6 shows the walk trial over the test terrain which consists of two plates with changing inclination (0%, 8%) and local obstacles with 4mm thickness. We can see that the suggested walk controller with both the surface learning and push recovery control can correctly learn the surface model and reject perturbation, and successfully let the robot walk over unknown, uneven surface[3]. On the other hand, walk controllers without surface learning or push recovery control cannot make the robot walk over the surface without falling down.

VI. Conclusions

We have proposed a simple and practical method to make a generic position-controlled humanoid robot without specialized hardware walk over unknown, uneven terrain. Local terrain height and inclination is estimated by means of electrically compliant swing leg and proprioceptive and inertial sensor readings, and an online learning algorithm is used to explicitly learn a surface model. To reject perturbations due to model error and external disturbances, a hierarchical push recovery controller based on biomechanically motivated push recovery behaviors is used, which takes the current

[3] `http://www.youtube.com/watch?v=pVesPOXGzvc`

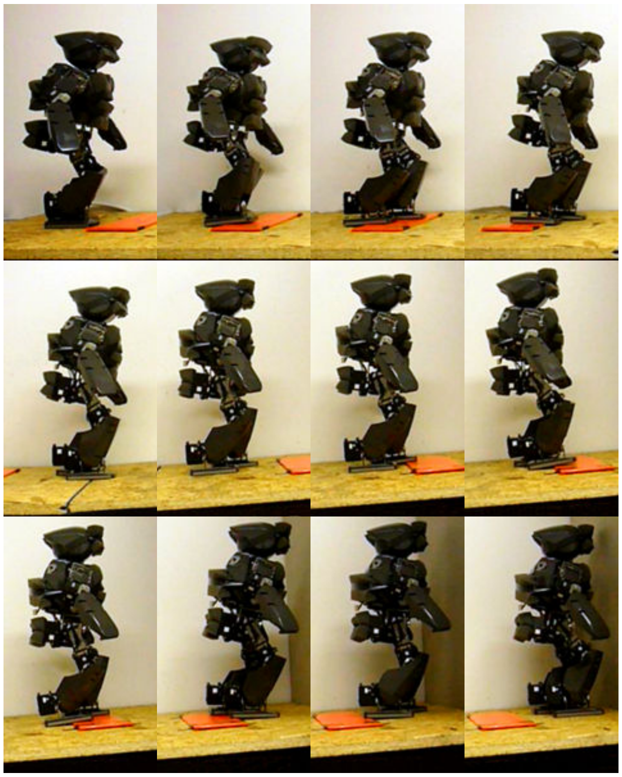


Fig. 6. Result of walk trail over unknown, uneven terrain with DARwIn-OP robot.

surface model, walk state and onboard sensory readings as input. The push recovery controller is trained to minimize a predetermined cost function by repeated trial in simulated environment using a full body physical model of the robot, and the learned push recovery controller is implemented on a commercially available DARwIn-OP humanoid robot without any modifications. Experimental results show that our integrated walk controller can make the robot walk over an uneven terrain which cannot be traversed by using a walk controller that uses either a surface-learning walk controller or a push recovery controller alone.

Our approach is simple enough to be implemented on an inexpensive, commercially-available small humanoid robot, yet it enables the robot to walk over an unknown, uneven terrain without falling down. Our approach can also be used in addition to an a priori surface model or better sensory equipment such as a range-finder or torque sensor, which would allow for more expressive surface model and less perturbation due to modeling error.

Future work includes learning the walk controller using the physical robot and a moving platform that can apply a controlled perturbation to the robot, and implementing the suggested algorithm to the full sized humanoid robots.

Acknowledgments

We acknowledge the support of the NSF PIRE program under contract OISE-0730206. This work was also partially supported by NSF grant (CNS-0958487), IT R&D program of MKE/KEIT (KI002138, MARS), NRF grant of MEST (314-2008-1-D00377, Xtran), and the BK21-IT program of MEST.

References

[1] M. Kalakrishnan, J. Buchli, P. Pastor, and S. Schaal, "learning locomotion over rough terrain using terrain templates," in *IEEE international conference on intelligent robots and systems*, 2009.

[2] J. Kolter, M. Rodgers, and A. Ng, "A control architecture for quadruped locomotion over rough terrain," in *ICRA*, May 2008, pp. 811–818.

[3] N. Ratliff, J. A. D. Bagnell, and S. Srinivasa, "Imitation learning for locomotion and manipulation," in *IEEE-RAS International Conference on Humanoid Robots*, November 2007.

[4] J. Rebula, P. Neuhaus, B. Bonnlander, M. Johnson, and J. Pratt, "A controller for the littledog quadruped walking on rough terrain," in *ICRA*, April 2007, pp. 1467–1473.

[5] D. Pongas, M. Mistry, and S. Schaal, "A robust quadruped walking gait for traversing rough terrain," in *ICRA*, April 2007, pp. 1474–1479.

[6] K. Hashimoto, Y. Sugahara, A. Ohta, H. Sunazuka, C. Tanaka, M. Kawase, H. Lim, and A. Takanishi, "Realization of stable biped walking on public road with new biped foot system adaptable to uneven terrain," in *BioRob*, Feb. 2006, pp. 226–231.

[7] S.-H. Hyon, "Compliant terrain adaptation for biped humanoids without measuring ground surface and contact forces," *IEEE Transactions on Robotics*, vol. 25, no. 1, pp. 171–178, Feb. 2009.

[8] Y.-D. Kim, B.-J. Lee, J.-H. Ryu, and J.-H. Kim, "Landing force control for humanoid robot by time-domain passivity approach," *IEEE Transactions on Robotics*, vol. 23, no. 6, pp. 1294–1301, Dec. 2007.

[9] J.-Y. Kim, I.-W. Park, and J.-H. Oh, "Walking control algorithm of biped humanoid robot on uneven and inclined floor," *J. Intell. Robotics Syst.*, vol. 48, no. 4, pp. 457–484, 2007.

[10] Y. Mikuriya, H. Sunazuka, K. Hashimoto, Y. Sugahara, M. Kawase, H. O. Lim, T. Hosobata, and A. Takanishi, "Walking control method of biped locomotors on inclined plane." in *ICRA*. IEEE, 2005, pp. 1977–1982.

[11] M. Ogino, H. Toyama, and M. Asada, "Stabilizing biped walking on rough terrain based on the compliance control," in *IROS*. IEEE, 2007, pp. 4047–4052.

[12] S.-H. Hyon and G. Cheng, "Disturbance rejection for biped humanoids," *IEEE International Conference on Robotics and Automation*, pp. 2668 –2675, apr. 2007.

[13] J. Park, J. Haan, and F. C. Park, "Convex optimization algorithms for active balancing of humanoid robots," *IEEE Transactions on Robotics*, vol. 23, no. 4, pp. 817–822, 2007.

[14] B. Jalgha and D. C. Asmar, "A simple momentum controller for humanoid push recovery," in *the FIRA RoboWorld Congress 2009 on Advances in Robotics*. Berlin, Heidelberg: Springer-Verlag, 2009, pp. 95–102.

[15] M. Abdallah and A. Goswami, "A biomechanically motivated two-phase strategy for biped upright balance control," in *In International Conference on Robotics and Automation*, 2005, pp. 2008–2013.

[16] J. Pratt, J. Carff, and S. Drakunov, "Capture point: A step toward humanoid push recovery," in *in 6th IEEE-RAS International Conference on Humanoid Robots*, 2006, pp. 200–207.

[17] T. Komura, H. Leung, S. Kudoh, and J. Kuffner, "A feedback controller for biped humanoids that can counteract large perturbations during gait," in *Proc. IEEE Int. Conf. Robotics and Automation*, 2005, pp. 1989–1995.

[18] D. N. Nenchev and A. Nishio, "Ankle and hip strategies for balance recovery of a biped subjected to an impact," *Robotica*, vol. 26, no. 5, pp. 643–653, 2008.

[19] B. Stephens, "Humanoid push recovery," in *Proceedings of the IEEE RAS International Conference on Humanoid Robots*, 2007.

[20] S.-J. Yi, B.-T. Zhang, and D. D. Lee, "Online learning of uneven terrain for humanoid bipedal walking." in *AAAI'10*, 2010, pp. –1–1.

[21] S.-J. Yi, B.-T. Zhang, D. Hong, and D. D. Lee, "Learning full body push recovery control for small humanoid robots." in *IEEE International Conference on Robotics and Automation*, 2011.

[22] S. Kajita, M. Morisawa, K. Harada, K. Kaneko, F. Kanehiro, K. Fujiwara, and H. Hirukawa, "Biped walking pattern generator allowing auxiliary zmp control," in *Intelligent Robots and Systems, 2006 IEEE/RSJ International Conference on*, oct. 2006, pp. 2993 –2999.