# Exploring the dynamic walking range of the biped robot “*RunBot*” with an active upper-body component


Poramate Manoonpong
Bernstein Center for Computational Neuroscience, University of Göttingen
Göttingen, Germany
Email: poramate@chaos.gwdg.de

Tao Geng
Department of Psychology
University of Stirling
Stirling, UK
Email: runbot05@googlemail.com

Florentin Wörgötter
Bernstein Center for Computational Neuroscience, University of Göttingen
Göttingen, Germany
Email: worgott@chaos.gwdg.de



***Abstract*— In this paper, we explore the dynamic walking capability of the planar biped robot RunBot with a now added active upper-body component. Originally, the robot was designed and built to perform fast walking especially on a flat floor. Its locomotion is driven by so-called neural reflexive control. This controller does not employ any kind of position or trajectory-tracking control algorithm. Instead, it enables RunBot to exploit its own natural dynamics during critical stages of its gait cycles. The actual gait pattern is determined by the set of neural control parameters, like gain and activation thresholds. Thus, different gait patterns can be induced by changing these parameters. These walking patterns, cooperating with an added active-upper body component, allow RunBot to walk on different terrains, e.g. flat floor, up and down slopes between 0 and 7.5 degrees. The transition phase of each gait was experimentally tuned. As a result, RunBot can *continuously* walk on the three different terrains. During walking experiments, gait switching was manually triggered while the active body was controlled to lean either forward or backward according to the slope.**



***Index Terms*— Biped robot, Reflexive neural control, Dynamic walking, Active body component**


## I. Introduction

Research in the domain of dynamic walking of biped robots has been ongoing for over 10 years [1], [2], [3] because, by this, one hopes to reach human-like performance and energy efficiency while walking. Additionally, such robotic systems can serve as platforms for study human locomotion [4], [5]. However, the difficult problem faced by these systems is not only their design but also how to control them, especially when wanting a dynamic change of the walking pattern allowing the robot to continuously walk on different terrains [6], [7], [8]. Thus, the rational behind this paper is to explore the dynamic walking capability of the planar biped robot RunBot [9] with a now added active upper-body component (see Fig. 1).

The following section describes the technical specifications of RunBot with its active-upper body component. Section 3 explains the neural model of the reflexive network for locomotion control as well as the neural model for active body control. The experiments and results are discussed in section 4. Conclusions and an outlook on future research are given in the last section.

## II. The biped robot RunBot

RunBot is a planar biped robot [9] which is 23 cm tall, foot to hip joint axis (see Fig. 1). Each leg consists of two degrees of freedom: hip and knee joints. Each hip joint is driven by a modified RC servo motor producing a torque up to 5.5 kg.cm while the motor of each knee joint produces a smaller torque (3 kg.cm) but has fast rotating speed with 21 rad/s. The built-in servo control circuits of the motors are disconnected while the built-in potentiometer is used to measure the joint angles. A mechanical stopper is implemented on each knee joint to prevent it from going into hyperextension similar to human kneecaps. RunBot has no actuated ankle joints resulting in very light feet and being efficient for fast walking. Its feet were designed having a circular form 4.5 cm long [9] similar to passive biped robots [1], [3]. Each foot is equipped with a switch sensor to detect ground contact. This mechanical design of RunBot has some special features, e.g. small curved feet and a properly positioned center of mass that allow the robot to perform natural dynamic walking during some stage of its gait cycles [9]. Hip and knee joints are driven by output signals of the reflexive neural controller through a DA/AD (USB-DUX[1]) board. Sensory signals are also digitized using this board for the purpose of feeding them into the controller.

To extend its walking capabilities for walking up and down slopes one servo motor with a fixed mass, called the *active upper body component*, is implemented on the top of its hip joints for balance. The active body has the total weight of 50 g. It will be controlled to lean forward (see Fig. 1, right) during walking up to a slope of 7.5 degrees while it will be controlled to lean backward (see Fig. 1, left) during walking on a flat floor and down a shallow slope of 3.5 degrees. The active component is manually controlled by a switch sensor via a servo controller board[2]. In addition, keyboard signals are also

[1] http://www.linux-usb-daq.co.uk/.
[2] http://www.ais.fraunhofer.de/BE/volksbot/mboard-content.html.

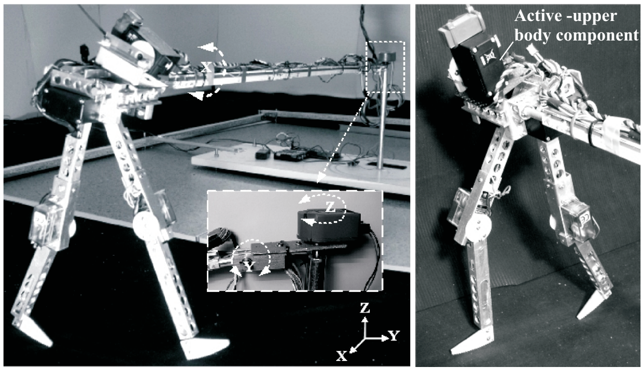


Fig. 1. The planar robot $RunBot$ with its active-upper body component. Left: The three orthogonal axes of the boom indicated by curved arrows rotate freely. Right: The active upper-body component.

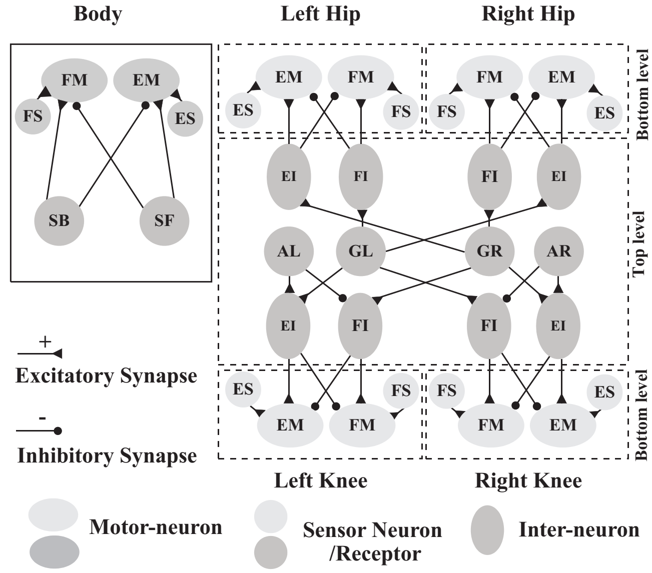


Fig. 2. The reflexive neural controller for generating locomotion of RunBot. AL (AR) refers to stretch receptor for anterior extreme angle of left (right) hip. GL (GR) refers to ground contact sensor neuron of left (right) foot. EI (FI) refers to extensor (flexor) reflex inter-neuron. EM (FM) refers to extensor (flexor) motor neuron and ES (FS) is extensor (flexor) sensor neuron. In addition, switch sensor neurons (SB and SF) are used to control the movement of the active body for leaning forward or backward.

used to switch between different walking patterns according to the terrain conditions. However, these signals are going to be replaced by sensor signals, e.g. accelerometer signals in the future. RunBot is constrained sagitally by a boom of one meter length. It is attached to the boom via a freely-rotating joint in the x axis while the boom is attached to the central column with freely-rotating joints in the y and z axes (compare Fig. 1, left). Thus, the motions of RunBot are only constrained on a circular path. This set-up has no influence on dynamics of RunBot in the sagittal plane.

## III. Reflexive neural controller

The reflexive neural controller [10] of RunBot for generating locomotion follows a hierarchical structure. It consists of two levels: top and bottom. The bottom level (see Fig. 2, dashed frames) is the reflex neural circuit local to the joints, including motor neurons and angle sensor neurons involved in joint reflexes. The top level (see Fig. 2, dashed frames) is a distributed neural network consisting of hip stretch receptors, ground contact sensor neurons and inter-neurons which command the motor neurons.

In addition, two extra motor neurons together with angle sensor neurons are added to control movements of the active body which will improve the walking capability, e.g. walking up and down slopes. Their activations are controlled by switch sensor neurons (see Fig. 2, solid frame). In case of walking up the slope, the active body will be controlled to lean forward (extensor movement) while in a normal walking condition (on a flat floor) and walking down the slope it will be controlled to lean backward (flexor movement) (compare Fig. 3).

All neurons are modelled as non-spiking neurons simulated on a Linux PC, and communicated to the robot via the USB-DUX board at the update frequency of 250 Hz, except the neurons involved in active body control which are independently simulated on the servo controller board. The directions of the extensor (flexor) movements and the thresholds of the sensor neurons are illustrated in Fig. 3. At the bottom level, the functions of the thresholds of the sensor-neurons ($\Theta_{ES,h}$, $\Theta_{FS,h}$, $\Theta_{ES,k}$, $\Theta_{FS,k}$ see Figs. 2 and 3) in each neuron module roughly limit the extensor and flexor movement of the joint. Likewise, the thresholds of the sensor-neurons of the body control $\Theta_{ES,b}$, $\Theta_{FS,b}$ are presented in Figs. 2 and 3, while, at the top level, the functions of the stretch receptor (Anterior Extreme Angle (AEA) signal) and the ground contact signal are different (see Fig. 4).

### A. Model neuron circuit of the top level

The joint coordination mechanism in the top level is implemented with the neuron circuit illustrated in Fig. 2. Each of the ground contact sensor neurons has excitatory connections to the inter-neurons of the ipsi-lateral hip flexor and knee extensor as well as to the contra-lateral hip extensor and knee flexor. The stretch receptor of each hip has excitatory connections to its ipsi-lateral inter-neuron of the knee extensor, and inhibitory connection to its ipsi-lateral inter-neuron of the knee flexor. Detailed models of the interneuron, stretch receptor, and ground contact sensor neuron are described in the following subsections.

*1) Inter-neuron model:* The inter-neuron model is adapted from one used in the neural controller of a hexapod simulating insect locomotion [11]. The state of each model neuron [12] is described as:

$$\tau_i \frac{dy_i}{dt} = -y_i + \sum w_{i,j} u_j \tag{1}$$

$$u_j = \frac{1}{1 + e^{\Theta_j - y_j}} \tag{2}$$

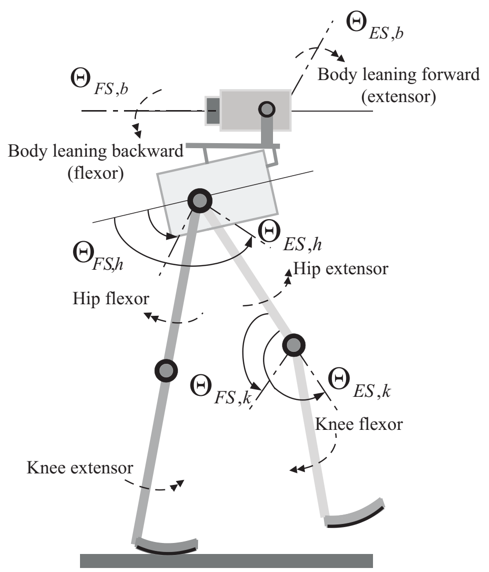


Fig. 3. Control parameters for the joint angles. $\Theta_{ES}$ ($\Theta_{FS}$) indicates to the threshold of the sensor neuron for extensor (flexor) where the subscripts ($b$, $h$ and $k$) are for body, hip and knee, respectively.

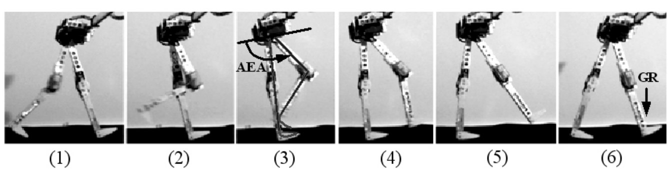


Fig. 4. Series of frames of one walking step. At the time of frame (3), The stretch receptor (AEA signal) of the swing leg is activated, which triggers the extensor of the knee joint in this leg. At the time of frame (6), the swing leg begins to touch the ground. This ground contact signal triggers the hip extensor and knee flexor of the stance leg, as well as the the hip flexor and knee extensor of the swing leg. Thus the swing leg and the stance leg swap their roles thereafter.

where $y_i$ represents the mean membrane potential of the neuron. Equation 2 is a sigmoidal function that can be interpreted as the neuron's short term average firing frequency, $\Theta_j$ is a bias constant that controls the firing threshold. $\tau_i$ is a time constant associated with the passive properties of the cell membrane [12], $w_{i,j}$ represents the connection strength from the $j$ th neuron to the $i$ th neuron.

*2) Stretch receptors:* Stretch receptors play a crucial role in animal locomotion control. When the limb of an animal reaches an extreme position, its stretch receptor sends a signal to the controller, resetting the phase of the limbs. There is also evidence that phasic feedback from stretch receptors is essential for maintaining the frequency and duration of normal locomotive movements in some insects [13]. While other biologically inspired locomotive models and robots use two stretch receptors on each leg to signal the attaining of the leg's AEP (Anterior Extreme Position) and PEP (Posterior Extreme Position) respectively, RunBot has only one stretch receptor on each leg to signal the AEA (Anterior Extreme Angle) of its hip joint. Furthermore, the function of the stretch receptor on RunBot is only to trigger the extensor reflex on the knee joint of the same leg, rather than to implicitly reset the phase relations between different legs.

As a hip joint approaches the AEA, the output of the stretch receptors for the left (AL) and the right hip (AR) are increased as:

$$\rho_{AL} = \frac{1}{1 + e^{\alpha_{AL}(\Theta_{AL} - \phi)}} \tag{3}$$

$$\rho_{AR} = \frac{1}{1 + e^{\alpha_{AR}(\Theta_{AR} - \phi)}} \tag{4}$$

where $\phi$ is the actual angular position of the hip joint, $\Theta_{AL}$ and $\Theta_{AR}$ are the hip anterior extreme angles whose value are manually tuned in an experiment, $\alpha_{AL}$ and $\alpha_{AR}$ are positive constants. This model is inspired by a sensor neuron model presented in [14] that is thought capable of emulating the response characteristics of populations of sensor neurons in animals.

*3) Ground contact sensor neurons:* Another kind of sensor neuron incorporated in the top level is the ground contact sensor neuron, which is active when the foot is in contact with the ground. Its output, similar to that of the stretch receptors, changes according to:

$$\rho_{GL} = \frac{1}{1 + e^{\alpha_{GL}(\Theta_{GL} - V_L + V_R)}} \tag{5}$$

$$\rho_{GR} = \frac{1}{1 + e^{\alpha_{GR}(\Theta_{GR} - V_R + V_L)}} \tag{6}$$

where $V_L$ and $V_R$ are the output voltage signals from switch sensors of the left foot and right foot respectively, $\Theta_{GL}$ and $\Theta_{GR}$ work as thresholds, $\alpha_{GL}$ and $\alpha_{GL}$ are positive constants.

While AEP and PEP signals account for switching between stance phase and swing phase in other walking control structures, ground contact signals play a crucial role in phase transition control of this reflexive controller. In PEP/AEP-models, the movement pattern of a leg will break down as soon as the AEP or PEP can not be reached, which may happen as a consequence of an unexpected disturbance from the environment or due to intrinsic failure. This can be catastrophic for a biped robot, though tolerable for a hexapod due to its high degree of redundancy.

### *B. Model neuron circuit of the bottom level*

In animals, a reflex is a local motor response to a local sensation. It is triggered in response to a suprathreshold stimulus. The quickest reflex in animals is the $monosynaptic\ reflex$, in which the sensor neuron directly contacts the motor-neuron. Similarly, the bottom-level reflex system of RunBot consists of reflexes local to each joint (see Fig. 2). The neuron module for one reflex is composed of one angle sensor neuron and the motor-neuron it contacts (see Fig. 2). Each joint is equipped with two reflexes, extensor reflex and flexor reflex, both are modelled as a monosynaptic reflex, i.e. whenever its threshold is exceeded, the angle sensor neuron directly excites the corresponding motor-neuron. In addition, the motor-neurons of the local reflexes also receive an excitatory synapse and an inhibitory synapse from the inter-neurons of the top level, by which the top level can modulate the bottom level reflexes.

*1) Angle sensor neurons:* Each joint has two angle sensor neurons, one for the extensor reflex, and the other for the flexor reflex (see Fig. 2). Their models are similar to that of the stretch receptors described above. The extensor angle sensor neuron changes its output according to:

$$\rho_{ES} = \frac{1}{1 + e^{\alpha_{ES}(\phi - \Theta_{ES})}} \tag{7}$$

where $\phi$ is the actual angular position obtained from the potentiometer of the joint (see Fig. 3). $\Theta_{ES}$ is the threshold of the extensor reflex (see Fig. 3) and $\alpha_{ES}$ a positive constant.

Likewise, the output of the flexor sensor neuron is modelled as:

$$\rho_{FS} = \frac{1}{1 + e^{\alpha_{FS}(\Theta_{FS} - \phi)}} \tag{8}$$

where $\Theta_{FS}$ and $\alpha_{FS}$ are similar to above. It should be particularly noted that the thresholds of the sensor neurons in the reflex modules do not work as desired positions for joint control, because the reflexive controller does not involve any exact position control algorithms that would ensure that the joint positions converge to a desired value. In fact, as will be presented in walking experiments, the hip joints often pass these thresholds in swing- and stance phase, and move continuously until the friction of the joint gears stop it.

*2) Motor neurons:* The model of the motor-neuron is the same as that of the inter-neurons described above. Note that, in RunBot, the final output value of the motor neurons, after multiplication by a gain coefficient, is sent to the servo amplifier circuit to directly drive the joint motors. The voltage of the motor at each joint is described as:

$$MotorVoltage = M_{AMP} G_M (s_{EM} u_{EM} + s_{FM} u_{FM}), \tag{9}$$

where $M_{AMP}$ represents the magnitude of the servo amplifier, e.g. 3 on RunBot, $G_M$ stands for output gain of the motor-neurons in the joint. $s_{EM}$ and $s_{FM}$ are signs for the motor voltage of flexor and extensor in the joint, being +1 or -1, depending on the polarity of the motors. $u_{EM}$ and $u_{FM}$ are the outputs of the motor-neurons (see Fig 2).

### *C. Model neuron circuit of the active body control*

This neural circuit (see Fig. 2, solid frame) consists of two switch sensor neurons and two neuron modules for activating body movement: one module is for extensor motion (EM, leaning forward) and the other for flexor motion (FM, leaning backward). Each module is composed of one angle sensor neuron and one motor-neuron similar to the reflex neuron module.

The switch sensor neurons are used to control the motions of the active body according to the terrain conditions and they are manually activated. Their outputs, similar to that of the angle sensor neurons, change according to:

$$\rho_{SF} = \frac{1}{1 + e^{\alpha_{SF}(V_s - \Theta_{SF})}} \tag{10}$$

$$\rho_{SB} = \frac{1}{1 + e^{\alpha_{SB}(\Theta_{SB} - V_s)}} \tag{11}$$

where $V_s$ is the output voltage signal from a switch sensor, $\Theta_{SF}$ and $\Theta_{SB}$ work as thresholds, $\alpha_{SF}$ and $\alpha_{SB}$ are positive constants.

In the neuron modules, the angle sensor neurons are modelled in the similar way to the one in the bottom level except for the threshold values of the flexor $\Theta_{FS,b}$ and extensor $\Theta_{ES,b}$ which are different. Once the threshold value is reached, the activation of the motor-neuron of the corresponding side will be held until the signal of the activated switch sensor neuron becomes deactivated. The motor-neuron, which directly modulates the motions of the active body, has the same characteristics as the motor-neuron of the bottom level except $M_{AMP}$ and $G_M$ which are here set to 1.

Most of the values for the neural parameters chosen by trial and error method are listed in Table I, II and III while activation thresholds of the sensor neurons and the output gain of the leg motor-neurons, being a part of neural modules at the bottom level (compare Fig. 2, light gray neurons), are changed according to three different walking patterns described in the next section. The time constant $\tau_i$ of all neurons takes the same value of 5 ms. The weights of all inhibitory connections are set to -10. The weight of all excitatory connections are 10, expect those between inter-neurons and motor-neurons, which are 0.1.

TABLE I

PARAMETERS OF NEURONS FOR HIP AND KNEE JOINTS.FOR MEANING OF THE SUBSCRIPT, SEE FIG. 2

| Joints | $\Theta_{EI}$ | $\Theta_{FI}$ | $\Theta_{EM}$ | $\Theta_{FM}$ | $\alpha_{ES}$ | $\alpha_{FS}$ |
|---|---|---|---|---|---|---|
| Hip | 5 | 5 | 5 | 5 | 4 | 4 |
| Knee | 5 | 5 | 5 | 5 | 2 | 2 |

TABLE II

PARAMETERS OF STRETCH RECEPTORS AND GROUND CONTACT SENSOR NEURONS.

| $\Theta_{GL}$ | $\Theta_{GR}$ | $\Theta_{AL}$ | $\Theta_{AR}$ | $\alpha_{GL}$ | $\alpha_{GR}$ | $\alpha_{AL}$ | $\alpha_{AR}$ |
|---|---|---|---|---|---|---|---|
| 2 | 2 | = $\Theta_{ES}$ | = $\Theta_{ES}$ | 4 | 4 | 4 | 4 |

TABLE III

PARAMETERS OF SWITCH SENSOR NEURONS AND ANGLE SENSORS OF THE ACTIVE BODY.

| $\Theta_{SF}$ | $\Theta_{SB}$ | $\Theta_{ES,b}$ | $\Theta_{FS,b}$ | $\alpha_{SF}$ | $\alpha_{SB}$ |
|---|---|---|---|---|---|
| 2 | 2 | 115 | 5 | 4 | 4 |

*D. Neural parameters for generating different walking patterns*

Three different walking patterns for walking on a flat floor ($FlatFloor$), up a slope of 7.5 degrees ($SlopeUp$) and down a slope of 3.5 degrees ($SlopeDown$) were determined and the active body has to be also controlled to guarantee stable walking. In normal walking condition, a neural parameter set was determined from Fig. 5 of [9] presented in our previous study where the stable gaits appear in addition the upper-body is controlled to lean backward (see Fig. 3). In special situations, e.g. walking up and down the slopes, the parameter sets were experimentally adjusted with respect to the characteristics of human walking [15], [16], i.e. humans usually attempt to keep the center of mass more to the front by leaning forward together with changing gait during walking up a slope and vice versa for walking down. Similarly, the parameter set of RunBot for walking up was adjusted to allow RunBot keeping its center of mass more to the front, and in this situation the upper-body component is activated to lean forward (compare Fig. 3) and vice versa for walking down. Appropriate parameter sets for the different gaits are presented in Table IV where the first column describes the terrain conditions and other columns show the parameters with respect to the given terrains.

TABLE IV
THE PARAMETER SETS OF DIFFERENT WALKING PATTERNS FOR THE SPECIFIC TERRAIN CONDITIONS.

| Terrain | $\Theta_{ES,h}$ | $\Theta_{FS,h}$ | $\Theta_{ES,k}$ | $\Theta_{FS,k}$ | $G_{M,h}$ | $G_{M,k}$ |
|---|---|---|---|---|---|---|
| FlatFloor | 105 | 78 | 175 | 115 | 2.2 | 1.8 |
| SlopeUp | 125 | 106 | 171 | 111 | 1.8 | 1.8 |
| SlopeDown | 85 | 53 | 170 | 100 | 2.0 | 2.2 |

Once the parameter sets of the different walking patterns were found, the next step was to find smooth transition phases between the patterns. Choosing unappropriate transition phases, e.g. changing too fast or too slow from one gait to another gait, will cause RunBot to easily fall down. Thus, on the circular walking path of RunBot, the timing of four transition phases were experimentally explored, where: the first phase is the transition from walking on a flat floor to walking up the slope, the second phase is the transition from walking up to again walking on a flat floor, the third phase is the transition from walking on a flat floor to walking down the slope and the last phase is the transition from walking down the slope to again walking on a flat floor. All transition phases from one gait to another gait are divided into 5 time steps and the transition time of each change is identical to 0.2 s except the transition from walking on a flat floor to walking up a slope, which is set to 0.6 s.

The final stage is to trigger the walking pattern according to the terrain condition. Gait switching is currently still done by a human operator; i.e. during experiments the operator switches the walking pattern and controls the active body via a keyboard, when RunBot starts to approach or leave the slope.

## IV. EXPERIMENTS AND RESULTS

In this section, several experiments have been carried out to assess the walking capability of RunBot. The three different terrain conditions were set-up and RunBot was manually controlled to switch its walking pattern. First, we show the walking performance of RunBot. To do so, the three walking patterns together with the position of the active body were tested with the three terrain conditions: flat floor (FlatFloor), a slope up of 7.5 degrees (SlopeUp) and a slope down of 3.5 degrees (SlopeDown). Here, walking performance is measured by the number of walking steps successfully made before falling. Note that, one step is counted by two different phases: swing and stance phases. Each walking experiment was run for 5 times. The results are shown in Table V where $NormalGait$, $UpSlopeGait$ and $DownSlopeGait$ are the patterns of walking on a flat floor, up the slope and down the slope, respectively. $LF$ ($LB$) is the forward (backward) leaning position of the active body.

TABLE V
COMPARISON OF DIFFERENT WALKING CONDITIONS ON THREE DIFFERENT TERRAINS

| Walking conditions | FlatFloor | SlopeUp | SlopeDown |
|---|---|---|---|
| NormalGait and LF | 3 | 3 | 2 |
| NormalGait and LB | $\infty*$ | 1 | 2 |
| UpSlopeGait and LF | 1 | $\infty*$ | 0 |
| UpSlopeGait and LB | 1 | $\infty$ | 0 |
| DownSlopeGait and LF | $\infty$ | 1 | 2 |
| DownSlopeGait and LB | $\infty$ | 0 | $\infty*$ |

As shown in Table V, RunBot can walk stably on a flat floor ($\infty$, $\infty*$) when the corresponding mode (NormalGait) with leaning the body backward is used while it can walk maximum 3 steps and then falls down when the body leans forward. Surprisingly, RunBot has also a good walking performance when the mode for walking down the slope (DownSlopeGait) with leaning the body either forward or backward is applied. On the other hand, RunBot can walk only one step when the mode for walking up the slope (UpSlopeGait) with leaning the body either forward or backward is used.

For walking up the slope, the two walking conditions, where RunBot can walk stably, are UpSlopeGait with leaning body either forward or backward. Generally, leaning the body forward is important for keeping the balance during a transition phase from walking on a flat floor to up the slope. It can walk maximum 3 steps, when NormalGait with leaning the body forward is used, while only one step when the body leans backward. However, RunBot cannot walk up the slope (0 step) when DownSlopGait with leaning the body backward is used, i.e. it immediately falls down during the swing phase, but it is still able to walk one step when the body leans forward.

In walking down the slope, there is only one walking condition, being DownSlopeGait with leaning the body backward with which RunBot is able to successfully perform the task.

On the other hand, no stability exists when UpSlopeGait with leaning the body either forward or backward is used, while it can walk up to 2 steps for the remaining walking conditions.

From the experiments described above, there are only three specific walking patterns which are suitable for the given terrain conditions (see pairs between the walking condition and the terrain condition from $\infty*$ in Table V) and which still enable RunBot to walk stably when the transition phase between walking patterns is applied. The appropriate gait of RunBot during walking on the different terrains, a flat floor, the slope up and the slope down, are shown as the stick diagrams in Fig. 5 where black (gray) shows a right (left) leg.

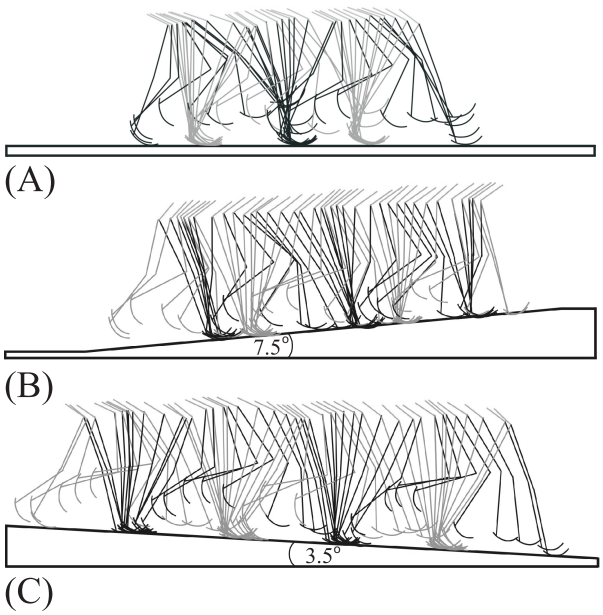


Fig. 5. Stick diagrams of RunBot walking on different terrains. (A) Walking on a flat floor by using NormalGait with leaning the body backward. (B) Walking up the slope of 7.5 degrees where the total length of the slope is 70 cm. UpSlopeGait with leaning the body forward was applied. (C) Walking down the slope of 3.5 degrees where the total length of the slope is 130 cm. DownSlopeGait with leaning the body backward was applied. The interval between any two consecutive snapshots of all diagrams is 33 ms.

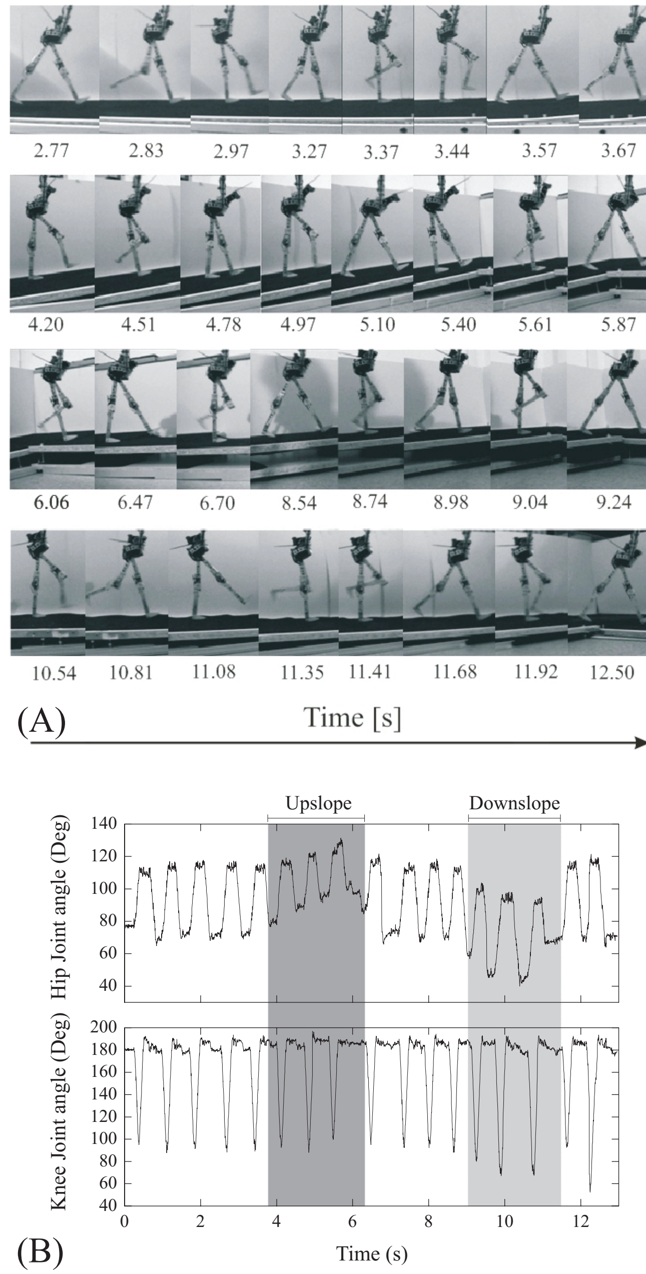


Fig. 6. (A) RunBot is walking on different terrains with the different walking patterns and it is also leaning the body according to the walking pattern and the terrain. Average walking speed was about 50 cm/s while maximum walking speed was about 55 cm/s during walking down a slope. (B) Real-time data of the angular sensor of the hip and knee joints while RunBot is walking on the different terrains: flat floor (white area), upslope (dark gray area) and downslope (light gray area).

It can be seen that RunBot obviously performs on an asymmetric gait, i.e. the step lengths between left and right legs are not equal, when it walks up the slope. While walking on a flat floor and walking down a shallow slope, its gaits are almost symmetried. Note that, the parameter sets of left and right legs are the same in all cases. This surprising phenomenon occurs because of the circular walking path of RunBot which causes the left and right legs of RunBot to traverse on different radii of the curvature where the left leg has smaller radius than the right one. Furthermore, gravitational effect, which resists forward motion, increasingly occurs during walking up the slope. As a result, RunBot can adapt its step length compatible to walk forwards on a curved trajectory.

Fig. 6A presents series of frames of RunBot during continuously walking on the different slopes. At the beginning, it walked on a flat terrain (from 2.77 s to 3.44 s) and then it approached to upslope at around 3.57 s. In this situation, its gait started to gradually change from NormalGait to UpSlopeGait as well as its body was activated to lean forward. It walked up the slope of 7.5 degrees until around 5.87 s. After that, it started to walk on flat terrain again where the gait was also changed and its body was activated to lean backward. Then, at around 9.24 s, it approached the downslope and its gait was changed from NormalGait to DownSlopeGait while the body was still leaning backward. Again, it returned to walk on flat terrain at around 12.5 s. One can also compare this frames series with real-time data of the angular position of a hip joint and a knee joint during walking in all states in Fig. 6B. The video clips of the experiments can be seen at http://www.chaos.gwdg.de/∼poramate/Runbot.html.

## V. Conclusion

RunBot with an added active upper-body component and its neural reflexive controller has been presented. It is used as the experimental platform to study dynamic walking and especially the problem of a dynamic change of the gait. Using real-time experiments, this study has shown that the different dynamic walking patterns and the transition phase between them, required to walk on different terrains, can be achieved in such simple dynamically stable bipeds, like RunBot. For future research in the following 6 months, more demanding tasks will be related to the use of additional sensors, e.g. accelerometer sensors or vision sensors, which enable RunBot to autonomously adapt its gait together with balancing itself by the use of the active body. Also, a learning technique will be applied to optimize the transition phase between the terrains.

## References


[1] M. Wisse and J. van Frankenhuyzen, "Design and construction of mike; a 2d autonomous biped based on passive dynamic walking," *In Proceedings of 2nd International Symposium on Adaptive Motion of Animals and Machines*, Kyoto, Japan, 2003.

[2] S. H. Collins, A. Ruina, R. Tedrake and M. Wisse, "Efficient Bipedal Robots Based on Passive Dynamic Walkers," *Science Magazine*, vol. 307, pp. 1082–1085, 2005.

[3] S. H. Collins, M. Wisse and A. Ruina, "A 3-D passive-dynamic walking robot with two legs and knees," *International Journal of Robotics Research*, vol. 20, no. 7, pp. 607–615, 2001.

[4] C. Paul, "Sensorimotor Control of Biped Locomotion," *Adaptive Behavior*, vol. 13, no. 1, pp. 67–80, 2005.

[5] M. Wisse, A. L. Schwab and R. Q. van der Linde, "A 3D Passive Dynamic Biped with Yaw and Roll Compensation," *Robotica*, vol. 19, no. 3, pp. 275–284, 2001.

[6] G. Endo, J. Morimoto, J. Nakanishi and G. Cheng, "An empirical exploration of a neural oscillator for biped locomotion control," *In Proceedings of IEEE International Conference on Robotics and Automation ICRA'04*, vol. 3, pp. 3036–3042, 2004.

[7] J. Pratt, "Exploiting Inherent Robustness and Natural Dynamics in the Control of Bipedal Walking Robots," *Ph.D. Thesis*, Computer Science Department, Massachusetts Institute of Technology, Cambridge, Massachusetts, 2000.

[8] S. Aoi and K. Tsuchiya, "Locomotion Control of a Biped Robot Using Nonlinear Oscillators," *Autonomous Robots*, vol. 19, no. 3, pp. 219–232, 2005.

[9] T. Geng, B. Porr and F. Wörgötter, "Fast Biped Walking with A Sensor-driven Neuronal Controller and Real-time Online Learning," *International Journal of Robotics Research*, Sage press, vol. 25, no. 3, pp. 243–259, 2006.

[10] T. Geng, B. Porr and F. Wörgötter, "A Reflexive Neural Network for Dynamic Biped Walking Control," *Neural Computation*, MIT press, vol. 18, no. 5, pp. 1156–1196, 2006.

[11] R. D. Beer, H. J. Chiel, R. D. Quinn, K. S. Espenschied and P. Larsson, "A distributed neural network architecture for hexapod robot locomotion," *Neural Computation*, vol. 4, pp. 356–365, 1992.

[12] J. C. Gallagher, R. D. Beer, K. S. Espenschied and R. D. Quinn, "Application of evolved locomotion controllers to a hexapod robot," *Robotics and Autonomous Systems*, vol. 19, pp. 95–103, 1996.

[13] H. J. Chiel and R. D. Beer, "The brain has a body: adaptive behavior emerges from interactions of nervous system, body and environment," *Trends in Neuroscience*, vol. 20, pp. 553–557, 1997.

[14] T. Wadden and O. Ekeberg, "A neuro-mechanical model of legged locomotion: Single leg control," *Biological Cybernetics*, vol. 79, pp. 161–173, 1998.

[15] A. N. Lay, "Neuromuscular Coordination during Slope Walking," *Ph.D. Thesis*, Bioengineering, School of Mechanical Engineering, Georgia Institute of Technology, 2005.

[16] T. A. McMahon, "Mechanics of Locomotion," *International Journal of Robotics Research*, vol. 3, no. 2, pp.4–28, 1984.