Abstract—The Bio-inspired Behavior-Based Bipedal Locomotion Control (B4LC) system consists of control units encapsulating feed-forward and feedback mechanisms, namely motor patterns and reflexes. To optimize the performance of motor patterns and reflexes in terms of stable locomotion on both even and uneven terrains, we present a learning scheme embedded in the B4LC system. By combining the Particle Swarm Optimization (PSO) method and the Expectation-maximization based Reinforcement Learning (EM-RL) method, a learning unit is comprised of an optimization module and a learning module embedded in the hierarchical control structure. The optimization module optimizes the motor patterns at hip and ankle joints with respect to energy consumption, stability and velocity control. The learning module generates compensating torques against disturbances at the ankle joints by combining the basis function derived from state information and the policy parameters. The optimization and learning procedures are conducted on a simulated robot with 21 DoFs. The simulation results show that the robot with optimized motor patterns and learned reflexes performs a more robust and stable locomotion on even and uneven terrains.

I. INTRODUCTION

Various studies are focusing on bipedal locomotion in complex environment. Among the most successful approaches, the classical control methods, e.g. Zero-Moment Point method, make a significant contribution to the development of bipedal control [1] [2]. However, limitation of those approaches is that robots developed accordingly possess limited skills in achieving human-like locomotion in terms of energy, stability and computational burden [3]. A promising alternative is to control the bipeds with biologically-inspired methods [4] [5].

By transferring some of the key findings in biomechanics and biology of human locomotion control to a biped robot, the Bio-inspired Behavior-Based Bipedal Locomotion Control (B4LC) system presented in [3] emerged. The B4LC system allows for various bipedal motions, e.g. stable standing against unexpected disturbances [6] and cyclic walking on even terrain while additionally rejecting external pushes [7]. Based on behavior-based control framework, this system is organized in hierarchical levels as depicted in Fig. 1. It is defined by the flow of stimulation, inhibition and modulation among six classes of control units. Corresponding to the brain of humans, locomotion modes are applied within the highest layer of the control system. The locomotion mode for cyclic walking will be activated after walking initiation.

The SPGs for walking are stimulated to activate five walking phases, which are weight acceptance, propulsion, stabilization, leg swing and heel strike respectively, according to the corresponding kinematic and kinetic events. The activated walking phase is in charge of managing the passive joints and stimulating the local control units in feed-forward and feedback manners, namely motor patterns and local reflexes.

Fig. 1. The hierarchical structure of the B4LC system: the Locomotion Modes, the SPGs, the Motion Phases and the Local Reflexes and the Motor Patterns. [3].

Although the B4LC system enables stable bipedal locomotion, one major problem remains: the motor patterns and reflexes with manually tuned parameters can hardly adapt to changing circumstances automatically. Even small changes of one unit’s parameters can result in large deviations due to the complicated interaction of motor patterns and reflexes. So far the parameters of controllers are defined experimentally by comparing the kinematic data of bipedal robot and humans [3] [6]. However, as the search space is enormous, this turns out to be impractical to achieve an optimal overall behavior. Therefore, adaptive control schemes with respect to stability, energy consumption and velocity control are designed in this paper. As motor patterns and reflexes are encapsulated in feed-forward and feedback units, we will implement two different methods to optimize the performance of those control units in Section III and IV respectively.

II. STATE OF THE ART

Several researches have been conducted on searching the optimal motor primitives in bipedal control [8] [9]. Among the most successful approaches, stochastic optimization methods, such as Particle Swarm Optimization (PSO) [10] and Genetic Algorithm (GA) [11], can be easily applied to optimize the bipedal locomotion in terms of trajectories, energy consumption and robustness. Sets of parameters for

---

*This work was funded by the European Commission 7th Framework Program under the project H2R (no.60069).

Q. Liu, J. Zhao, S. Schütz and K. Berns are with the Robotics Research Lab, University of Kaiserslautern, 67663 Kaiserslautern, Germany {liu, zhao, schuetz, berns}@cs.uni-kl.de
controllers at ankle, knee and hip joints are searched with optimized objective functions. One advantage over numerical optimization methods is that it requires no complex modeling of the robotic system.

Moreover, tremendous efforts have been made to extend the possibilities of machine learning techniques for robotic applications [12] [13]. One especially promising approach is the Policy Gradient Reinforcement Learning (PGRL) method, which has been successfully used for dynamic control of bipedal robots [14] [15]. However, the complicated modeling of the robots results in a high complexity of the learning procedure [16]. Furthermore, an accepted alternative poses the Expectation-maximization based Reinforcement Learning (EM-RL) [17]. Compared to PGRL, it has the advantage of not requiring a learning rate parameter. The approach shows satisfactory performance when applied in the context of learning dynamic motor primitives, i.e. the complex task of pancake flipping and Ball-in-a-Cup [18] as well as optimizing bipedal motions [19].

III. MOTOR PATTERN OPTIMIZATION

A. The Principle of Motor Patterns

In the B4LC system, motor patterns produce uniform patterns of torque for one or more joints in a feed-forward manner. The generated torques are modulated by the motion phases. Described in (1), a sigmoid function defines the shape of the motor patterns.

\[
\tilde{\tau} = A \cdot \begin{cases} \frac{1}{2} + \frac{1}{2} \sin(\pi(\frac{T_1}{T_1} - \frac{1}{2})) & 0 \leq t < T_2 \\ 1 & T_1 \leq t < T_2 \\ \frac{1}{2} - \frac{1}{2} \sin(\pi(\frac{T_3 - T_2}{T_3 - T_2} - \frac{1}{2})) & T_2 \leq t < T_3 \end{cases} \tag{1}
\]

\(A, T_1, T_2\) and \(T_3\) are the parameters of the motor patterns defining the maximum torque, the starting time of maximum torque, the ending time of maximum torque and the total time till the torque reaches zero again. Some examples are illustrated in Fig. 2.

The motor patterns \textit{Leg Prop} (LP) acting at the ankle joints during walking phase \textit{propulsion} and \textit{Active Hip Swing} (AHS) acting at the hip joints during walking phase \textit{leg swing} play significant roles in cyclic walking. By applying torques at ankle and hip joints respectively, the main contribution of them is to accelerate the body forward and swing the leg ahead. In [3] the parameters of LP and AHS have been defined experimentally. However, there’s space for improvement regarding stability, energy consumption and velocity control. Thus, optimizing the parameters of motor patterns automatically is an essential prerequisite of a successful bipedal control. As motor patterns are pure feed-forward units, we apply the stochastic optimization method PSO to optimize those parameters. This method is well suited for defining the optimal parameters of open-loop controllers in continuous search space and hence is able to optimize the behaviors of AHS and LP.

B. Particle Swarm Optimization

Eberhart first introduced PSO in [20]. The movements of a particle is guided in the direction of its own best known position and the best know position of the entire swarm. The velocity of a particle in one iteration can be derived from:

\[
v_p = \omega v_p + c_1 \cdot Rand_1( B_p - x_p ) + c_2 \cdot Rand_2( B_g - x_p ) \tag{2}\]

\(v_p\) and \(x_p\) represent the velocity and the position of a particle respectively. \(B_p\) is the known best position of the particle in all past iterations, while \(B_g\) is the known best position of any particle in the entire swarm in all past iterations. \(c_1\) and \(c_2\) are the parameters of the acceleration constants respectively. \(Rand_1\) and \(Rand_2\) are random values ranging within [0, 1]. The inertia weight \(\omega\) controls the impact of the previous velocities on the current velocity. Larger inertia weights tend to favor global searching, while smaller values lead to a more local searching strategy.

During every iteration, the fitness function, as shown in (5), evaluates a given solution regarding the objectives. The best position of particles \(B_p\) and the best position of the entire swarm \(B_g\) are updated by comparing the fitness values in each iteration. The position of a particle \(x_p\) is updated according to (3).

\[
x_p = x_p + v_p \tag{3}\]

C. Implementation of the Optimization in the B4LC

Activated by the stimulation signals of the corresponding motion phases, the optimization module feeds generated outputs into each motor pattern as parameters \(T_1, T_2\) and \(A\). The overall time of motor pattern \(T_3\) is kept constant as originally introduced in [3].

The position of a particle is determined by the parameter of motor patterns LP and AHS. As depicted in Fig. 3, 6-dimensional vectors \(\vec{f}_{i,t}\) consisting of \([T_{a1}, T_{a2}, A_a]\) and \([T_{h1}, T_{h2}, A_h]\), are generated by optimization module, where \(i\) is the number of particles and \(t\) represents the iteration times. The particle positions are initialized randomly in the search space. In each iteration, the motor patterns LP and AHS generate torques \(\tau_a\) and \(\tau_h\) at ankle and hip joints. To increase the optimization speed, the search space is constrained to a range based on the previous experience, as
depicted in (4).
\[
\begin{align*}
T_{a1} &\in [0.0, 0.6] & T_{h1} &\in [0.0, 0.4] \\
T_{a2} &\in [0.6, 0.9] & T_{h2} &\in [0.4, 0.7] \\
A_{a} &\in [0.0, 1.0] & A_{h} &\in [0.0, 1.0]
\end{align*}
\] (4)

The fitness function reflects the optimization goals including robustness, stability, speed and energy consumption:
\[
F = k_1 f_1 - k_2 f_2 - k_3 f_3 - k_4 f_4
\] (5)

\[
k_1, k_2, k_3 \text{ and } k_4 \text{ are the weightings of different goals, and}
\]
\[
f_1 = s_w
\] (6)
\[
f_2 = \sum_{i=0}^{i_t} \Delta X_{com i}/s_w
\] (7)
\[
f_3 = \sum_{i=0}^{i_t} (V_{ref} - V_i)^2/i_t
\] (8)
\[
f_4 = \sum_{i=0}^{i_t} \tau^h_{i}/i_t
\] (9)

\(i\) is the time step and \(i_t\) is the total number of the achieved time steps in the corresponding iteration. \(f_1\) is defined as a function that is linear to the successful walking steps \(s_w\). \(f_2\) evaluates the stability of the robot by accumulating the error value of the extrapolated Center of Mass (Xcom), where \(\Delta X_{com i}\) is the deviation of Xcom at time step \(i\). \(f_3\) represents the deviation of the walking speed from the given reference. \(V_i\) is the actual walking speed at time step \(i\) and \(V_{ref}\) represents the reference speed. \(f_4\) contains the accumulated torques generated by the reflex Lock Hip (LH) which is utilized to stop the flexion of the hip at a desired angle towards the end of the swing phase. Hence \(\tau^h\) is the braking torque generated by LH at time step \(i\).

IV. ADAPTIVE REFLEXES WITH EM-RL

A. The Principle of Reflexes

Reflexes produce control outputs depending on the presence of corresponding sensory information. Some of the reflexes show linear and nonlinear relation between sensory data and control output. As the basic locomotion under normal circumstances is produced by the motor patterns, external disturbances are compensated by the reflexes. To maintain stable locomotion on rough terrains, the Control Forward Velocity (CFV) reflexes at the ankle joints are optimized in the following section. As the parameters of reflexes are constant in all situations, the control outputs can hardly cope with all possible changes of the sensory feedback. Accordingly, instead of applying PI controllers in those reflexes, we employ neural networks to provide the reflexive actions. The EM-RL introduced in [17] is used to learn the parameterized neural networks.

B. Expectation-maximization based Reinforcement Learning

The overall advantages of EM-RL can be concluded as: first, this method does not require a learning rate which has to be exactly revised during the learning process [17]. Second, episodic learning can be easily implemented so that the previous experience of the agent in the estimation of new exploratory parameters can be used [18].

Similar to PGRL, parameterized policies are used in EM-RL. The goal of the learning process is to find the parameters with maximum expected returns during a whole episode regarding the corresponding policy. An action is selected by the basis function and the policy parameters:
\[
a(t) = \theta^T \Phi(s),
\] (10)

where \(a(t)\) is the action derived at the time step \(t\) of an episode, \(\Phi(s)\) the basis function based on state \(s\) and \(\theta\) the policy parameters. The undiscounted cumulative reward in the episode \(e\) can be given as the return \(R(e)\), which is presented in (11).
\[
R(e) = \sum_{t=1}^{t_e} r(t),
\] (11)

where \(t_e\) is the duration of the episode and \(r(t)\) is the instant reward at time \(t\). The policy parameter \(\theta_e\) at the episode \(e\) is updated to a new parameter \(\theta_{e+1}\) as described in (12).

\[
\theta_{e+1} = \theta_e + \frac{1}{\omega(k)} \Phi < \Omega(k) > \frac{\langle \omega(k) \rangle}{\langle \Omega(k) \rangle}
\] (12)

The relative exploration between the parameters used in the past episodes with the best rewards and the parameters used in the current episode is given by a vector difference \((\theta_k - \theta_e)\). The relative exploration is weighted by the corresponding returns of the episodes. To decrease the number of episodes for new policy searching, a form of importance sampling technique, denoted by \(< \cdot >_{\omega(k)} \) in (12), is used to adapt the learning [18]. Thus, the re-use of the previous episodes and corresponding parameters is possible. The sampler is defined in (13).
\[
< f(\theta_k, e_k) >_{\omega(k)} = \sum_{k=1}^{\sigma} f(\theta_{ind(k)}, e_{ind(k)})
\] (13)

\(\sigma\) is defined as a fixed parameter representing the number of episode utilized by the sampler. \(ind(k)\) is an index function which returns the index of the \(k\)-th episodes with best rewards in the entire episodes. The update of the policy parameters stops until \(\theta_{e+1} = \theta_e\). The algorithm of EM-RL is shown in Algorithm 1.

\begin{algorithm}
1: Input: initial policy parameters \(\theta_0\)
2: repeat:
3: Perform episode using policy described in (10) and collect all immediate rewards for \(t = \{1, 2, ..., t_e + 1\}\).
4: Get episode return based on (11).
5: Find out the best \(\sigma\) episodes and update the policy parameters \(\theta_e\) of iteration \(e\) based on (12).
6: Until: \(\theta_{e+1} = \theta_e\)
\end{algorithm}
C. Implementation of the Learning in the B4LC

To achieve stable walking on uneven ground, we implement the CFV reflexes at the ankle joints during five walking phases. The CFV reflexes will produce the reflexive torques to reject the occurring disturbances in the sagittal plane. In case the forward velocity is too high, the plantar-flexion of the ankle joints will slow down the pendulum-like movement of the upper body over the stance leg, and vice versa. Neural networks are utilized to generate compensating torques. A learning module using the EM-RL learning method is implemented in the B4LC system to learn the parameters of those neural networks.

The working principle of the learning module is shown in Fig. 4. When the learning process starts, the sensory information is provided as state inputs to the learning module. The time step \( t \) during learning is initialized as zero when an episode begins and is updated every 25 ms. At each time step, a new action is calculated and fed into the CFV reflexes as the control input. The action is applied in form of a torque command exerted at the ankle joint.

\[
\text{RL module} \xrightarrow{s(t) \in \{s_1, s_2, s_3\}} \text{Reward Evaluation} \xrightarrow{\Delta X_{\text{com}}(t)} \text{Control Forward Velocity} \xrightarrow{f(t)} \text{Biped Robot}
\]

Fig. 4. The structure of the RL module.

The robot learns to perform a locomotion on uneven ground with randomly initialized policy parameters \( \theta_0 \). In accordance with the episodic learning principle, the policy parameters are not updated until an episode is finished. During learning process, one episode ends and a new episode starts when the biped falls down or a robust locomotion is achieved. This is assumed when the amount of walking steps reaches 30. Then, the undiscounted cumulative rewards of this episode are calculated and the policy parameters are updated.

The state of the biped robot is represented by a 3-dimensional vector \( s = [x_1, x_2, x_3] \), where \( x_1 \) and \( x_2 \) denote the position and velocity of robot’s CoM, while \( x_3 \) is the progress value of the stance phase. The reward function \( f(t) \) at the time step \( t \) can be expressed using the deviation value of the extrapolated center of mass generated by a higher-level reflex controller \( X_{\text{com Correction}} \) in form of:

\[
f(t) = e^{-k|\Delta X_{\text{com}}|}
\]

\( k \) stands for the gain parameter and \( \Delta X_{\text{com}} \) represents the deviation of the extrapolated center of mass \( X_{\text{com}} \) at the time step \( t \).

V. OPTIMIZATION AND LEARNING PROCEDURES

The optimization and learning scenario is set up on a 3D simulated robot with 21 degrees of freedom and 1.8m height. The coordinate system in the simulation is defined as: the origin is located in the pelvis of the biped, the x-axis is the robot’s walking direction, the y-axis is the lateral direction and the z-axis is the vertical direction.

First the motor patterns LP and AHS are optimized with PSO method during locomotion on flat ground. As limited disturbances are introduced in this situation, the CFV reflexes are not stimulated. We expect that the optimized motor patterns LP and AHS are able to perform stable walking on flat ground. Afterwards, to improve the capability of the disturbance rejection, the CFV reflexes and the optimized motor patterns LP and AHS are activated simultaneously. The CFV reflexes are learned with EM-RL method until robot is capable of achieving robust locomotion control on uneven ground.

A. Optimization Procedure

The robot is set up to start optimization with locomotion on a flat ground as presented on the left side of Fig. 5. The parameters generated by optimization module reflecting the position of a specific particle are fed into LP and AHS in the walking phase \( \text{propulsion} \) and \( \text{leg swing} \) respectively. When the pitch and roll angle of the robot exceed certain thresholds or a robust walking behavior emerges after a fixed number of successfully performed steps, the test of this particle is finished. Once all particles are tested, the best position of each particle as well as of the entire swarm are updated. Consequently, a new iteration is started with the updated velocities and positions of the particles. The average fitness values for each iteration are shown in Fig. 6.

![Fig. 5. The left side of the figure shows the setup in which the biped starts the optimization process by walking on even terrain. The middle and right side of the figure show the learning procedure of the reflexes. After 5-walking steps, the biped achieves stable locomotion and reaches the uneven ground.](image)

![Fig. 6. The figure shows the returns of fitness values during the optimization process. The yellow line presents the fitness value of the known best particles in all past iterations. The blue line indicates the average fitness value of all particles in each iteration. The horizontal line stands for the maximum and minimum value of the fitness value in each iteration.](image)

At the beginning, we use a high inertia weight \( \omega \), as introduced in (2), to follow a global searching strategy. After
50 iterations, the fitness values of the best known particles are constant, and the average fitness value of the particles shows no more significant improvement. To explicitly search the optimal parameters, we shift to a local searching strategy by decreasing the inertia weight $\omega$ after 160 iterations. The average fitness value in each iteration is significantly increasing afterwards and navigates closer to the best known fitness value. Subsequently the robot is able to perform stable and robust walking under the given external circumstances.

B. Learning Procedure

The robot is set up to start a stable locomotion on even ground with the optimized motor patterns, as shown in the middle and right side of Fig. 5. After 5 walking steps, the robot reaches an uneven ground consisting of structures with a maximum height of 4 cm. The learning module for the CFV reflexes on both legs are switched on. Again, when the robot is falling or a maximum number of successfully performed walking steps is achieved, one learning episode is finished. The return of the episode is calculated by summing up the undiscounted rewards.

It takes over 500 episodes to complete the learning process of the CFV reflexes. The actor’s parameters $\theta$ are initialized randomly. The average returns in each iteration are shown in Fig. 7. Before the first 20 episodes, the robot was hardly achieving stable walking for more than 5 steps. While after about 45 episodes, the biped was able to cope with the disturbances from the uneven ground resulting in the average return per episode being increased from 15 to 110.

VI. SIMULATION RESULTS

To verify the adaptive optimization and learning results proposed in Section V, experiments in different scenarios are conducted. The simulated biped is again used as the plant for locomotion on even and uneven terrains with optimized motor patterns and reflexes.

A. Locomotion on Even Ground

To evaluate the performance of motor patterns during normal walking, we conducted walking experiments on flat ground with optimized and non-optimized LP and AHS. As limited disturbances are involved during this scenario, the reflexes CFV are not activated to compensate the external perturbations. The reference velocity is set as 1.2 m/s. Running the experiments with optimized and non-optimized motor patterns, the torques generated at the ankle and hip joints are illustrated in Fig. 8. The data is normalized to one gait cycle and averaged over the 30 successive walking steps. The robot with optimized parameters shows less and smoother torque during activation. The torque profiles show smaller peak values during activation of LP and AHS after optimization. Looking at the full gait cycle, this leads to a longer swing phase and shorter stance phase compared to the robot with non-optimized parameters.

The comparison of performance regarding stability, energy consumption and velocity control averaged over 30 experimental runs before and after optimization is presented in Table I. The accumulated values of generated torques at the ankle and hip joints during one walking cycle decrease by 18.22 % and 11.83 % respectively. Furthermore, the accumulated $\Delta X_{com}$ is reduced from 2.721 m to 1.232 m. The velocity deviation with respect to the given reference velocity after optimization is significantly improved.

<table>
<thead>
<tr>
<th></th>
<th>Accumulated joint torques during one cycle</th>
<th>Velocity deviation during 10 seconds (%)</th>
<th>Accumulated $\Delta X_{com}$ during one cycle</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>ankle (Nm)</td>
<td>hip (Nm)</td>
<td></td>
</tr>
<tr>
<td>Before</td>
<td>3842.2</td>
<td>746.3</td>
<td>8.3</td>
</tr>
<tr>
<td>After</td>
<td>3142.1</td>
<td>657.9</td>
<td>0.04</td>
</tr>
<tr>
<td>Improvement</td>
<td>18.22%</td>
<td>11.83%</td>
<td>8.26</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>1.489</td>
</tr>
</tbody>
</table>

B. Locomotion on Uneven Ground

For the evaluation of the learned reflexes, We have conducted walking experiments on uneven ground. To compensate the disturbances induced by rough terrain, the learned reflexes CFV as well as the optimized motor patterns are activated simultaneously. The pelvis position in the z-axis during locomotion is presented in Fig. 9. The right leg stance phases and the left leg stance phases are represented by the red and blue areas respectively. The deviation of the pelvis position between gait cycles is influenced by the change of height of the terrain.

Consequently, to compensate the unexpected disturbances, the reflexes CFV generate ankle torques based on the actor
parameters derived in Section V. The produced torques with respect to $\Delta X_{com}$ are shown in Fig. 10. It can be seen that the trajectories of the correction value vary in large ranges both during left and right stance phases. This means that the $X_{com}$ values change and instability arises due to the unexpected disturbances caused by the uneven ground. Consequently, ankle torques are produced accordingly to prevent the biped from falling over in the sagittal plane. The profiles of the torque values correspond tightly to the correction value. In case the correction is too high, the $X_{com}$ deviates from the approximated reference trajectories. Therefore, an ankle dorsiflexion can be observed and additional forward torques are applied at the ankle joint of the stance leg. Likewise a large $X_{com}$ deviation will lead to an ankle planter-flexion and reduce the bipedal velocity by implementing rearward ankle torques.

![Fig. 9. The pelvis position in the z-axis is presented. The red and blue areas indicate the right leg stance phase and left leg stance phase respectively.](image)

Fig. 9. The pelvis position in the z-axis is presented. The red and blue areas indicate the right leg stance phase and left leg stance phase respectively.

![Fig. 10. The correction value stands for the $\Delta X_{com}$ which is calculated from the higher-level reflex controller $X_{com} \text{ Correction}$. The ankle torques around y-axis are implemented on both left and right ankle.](image)

Fig. 10. The correction value stands for the $\Delta X_{com}$ which is calculated from the higher-level reflex controller $X_{com} \text{ Correction}$. The ankle torques around y-axis are implemented on both left and right ankle.

VII. CONCLUSIONS

In this paper, we have presented the implementation of the learning modules both for motor patterns and reflexes in the B4LC system. A PSO-based optimization approach is applied to learn the parameters of the feed-forward motor patterns AHS and LP. The performance of the simulated biped is optimized in terms of stability, energy consumption and velocity control. The optimized parameters have been used to conduct the walking experiments on even ground. This optimization method allows for the generation of the basic bipedal locomotion without manually tuned parameters. Additionally, based on the EM-RL method, a learning approach for the reflexes has been implemented and the reflexes CFV at ankle joint are learned. The torques are generated by a parameterized policy. By calculating the return of each episode, the policy parameters are updated. Comparing the robots performance after optimization and learning to the setup presented in [3] shows that walking on rough terrain became more robust and stable. In future, the learning of locomotion on more challenging environments, e.g. uphill and downhill walking, will be investigated with the suggested schemes.

REFERENCES


[14] K. Hitomi, T. Shibata, Y. Nakamura, and S. Ishii, “Biped locomotion on more challenging environments, e.g. uphill locomotion on rough terrain became more robust and stable. In future, the learning of locomotion on more challenging environments, e.g. uphill and downhill walking, will be investigated with the suggested schemes.


[25] K. Hitomi, T. Shibata, Y. Nakamura, and S. Ishii, “Biped locomotion on more challenging environments, e.g. uphill locomotion on rough terrain became more robust and stable. In future, the learning of locomotion on more challenging environments, e.g. uphill and downhill walking, will be investigated with the suggested schemes.


