Qualitative Adaptive Reward Learning With Success Failure Maps: Applied to Humanoid Robot Walking

John Nassour, Vincent Hugel, Fethi Ben Ouezdou, Gordon Cheng

Qualitative Adaptive Reward Learning With Success Failure Maps: Applied to Humanoid Robot Walking

John Nassour, *Member, IEEE*, Vincent Hugel, *Member, IEEE*, Fethi Ben Ouezdou, *Member, IEEE*, and Gordon Cheng, *Senior Member, IEEE*

***Abstract*— In the human brain, rewards are encoded in a flexible and adaptive way after each novel stimulus. Neurons of the orbitofrontal cortex are the key reward structure of the brain. Neurobiological studies show that the anterior cingulate cortex of the brain is primarily responsible for avoiding repeated mistakes. According to vigilance threshold, which denotes the tolerance to risks, we can differentiate between a learning mechanism that takes risks and one that averts risks. The tolerance to risk plays an important role in such a learning mechanism. Results have shown the differences in learning capacity between risk-taking and risk-avert behaviors. These neurological properties provide promising inspirations for robot learning based on rewards. In this paper, we propose a learning mechanism that is able to learn from negative and positive feedback with reward coding adaptively. It is composed of two phases: evaluation and decision making. In the evaluation phase, we use a Kohonen self-organizing map technique to represent success and failure. Decision making is based on an early warning mechanism that enables avoiding repeating past mistakes. The behavior to risk is modulated in order to gain experiences for success and for failure. Success map is learned with adaptive reward that qualifies the learned task in order to optimize the efficiency. Our approach is presented with an implementation on the NAO humanoid robot, controlled by a bioinspired neural controller based on a central pattern generator. The learning system adapts the oscillation frequency and the motor neuron gain in pitch and roll in order to walk on flat and sloped terrain, and to switch between them.**

***Index Terms*— Experience-based learning mechanism, humanoid learning, humanoid robot walking, neurorobotics.**

I. Introduction

IN THIS paper, we bring forward an approach to better match biological models of brain-like mechanisms in learning tasks. The key point presented here is the careful combination of two usually isolated studies of two distinct brain regions, namely, the anterior cingulate cortex (ACC) and the orbitofrontal cortex (OFC). We draw upon these studies in coming up with a functional and practical computational model that has been applied to a physical humanoid robot. Fig. 1 provides a conceptual overview of this paper. We have addressed the development of a learning mechanism based on the well-known self-organizing maps (SOMs). Walking has been used as an example task, which follows from our own previous neuronal-based studies on the central pattern generator (CPG) of the spinal cord for pattern generation for walking [1].

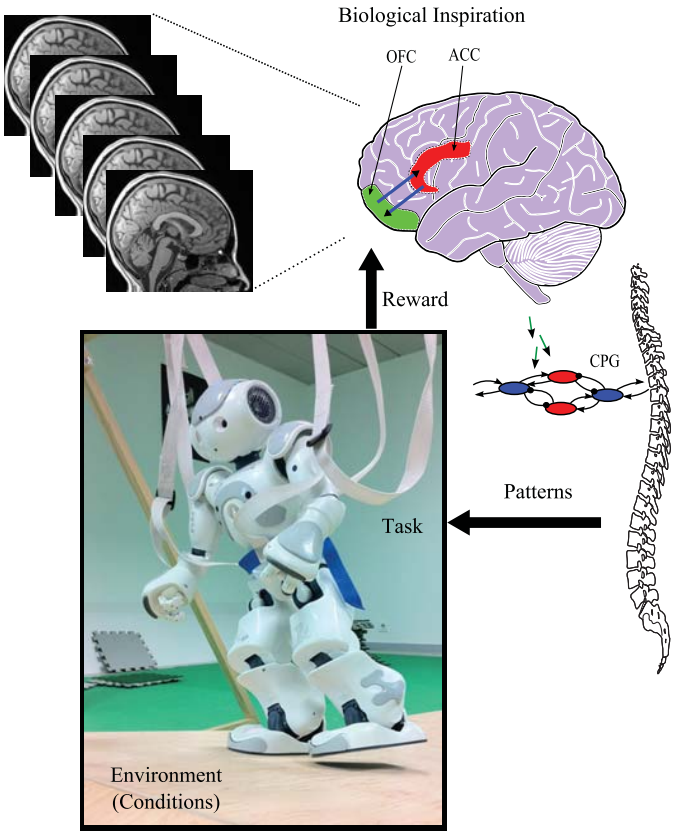

Fig. 1. Conceptual overview of our work. ACC is the interior cingulate cortex. OFC is the orbitofrontal cortex. CPG is the central pattern generator.

The adaptation property of the brain even with limited dynamic coding range enables efficient processing of different physical events such as locomotion [2]. The brain's reward system discriminates a diversity of possible rewards, which can ensure the best conditions for survival. The OFC is related to reward dealing in the brain. Damages to the OFC have shown abnormal response to changes to reward contingencies [3]. Due to the sensitivity of neurons of this cortex and to the types and the amount of rewards, OFC can be said to encode

J. Nassour is with the Institute for Cognitive Systems, Technical University of Munich, Munich 80290, Germany, and also with the Engineering System Laboratory, Versailles University, Versailles 78000, France (e-mail: nassour@tum.de).

V. Hugel and F. B. Ouezdou are with the Engineering System Laboratory, Versailles University, Versailles 78000, France (e-mail: fhugel@lisv.uvsq.fr; ouezdoug@lisv.uvsq.fr).

G. Cheng is with the Institute for Cognitive Systems, Technical University of Munich, Munich 80333, Germany (e-mail: gordon@tum.de).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

reward features into a scalar value [4]. Physiological studies demonstrated the adaptivity of the OFC in coding the reward according to the available rewards that changed in every block of trials [5]. They show how the coding of reward in this cortex can be affected by the changes in reward distribution [6]. This supports the concept that the OFC adjusts rewards information in a flexible and adaptive manner after each new stimulus [7].

Neurocognitive studies have identified an early warning system in the human brain that can avoid making past mistakes. They have shown how the brain remembers details about past dangers [8]. The ACC is activated during high-risk decision [9], and also after making mistakes [10]. This cortical area acts as an early warning system that adjusts the behavior to avoid dangerous situations. It responds not only to the sources of errors (external error feedback) but also to the earliest sources of error information available (internal error detection) [11]. It becomes active in proportion to the occurrence of likelihood of an error [12], [13]. Therefore, it can learn to identify situations where humans may make mistakes, and then help avoiding such situations occurring again [10]. It learns to predict error likelihood even for situations where no error occurred previously [14]. Through the observation of particular areas located in the cerebral cortex in the brain responsible for cognitive control, neuropsychological studies have demonstrated a switching in human learning strategies around the age of 12 years. This switch from learning with positive feedback to learning with negative feedback probably comes from a combination of brain maturing and experience [15]. It has been shown that the decision of taking risk is accomplished by activities in ACC and OFC [9]. Activity increases with failure likelihood and also reward action likelihood. The fusion of the functionalities of these two cortex areas in one mechanism gives raise to the possibility of getting a task learning system that could predict risky cases and avoid danger (e.g., learning to walk).

Computational models of learning systems such as techniques based on the associative memory such as the CMAC neural networks rely on offline trajectory generation. They first learn the joints trajectory, and then generate the learned trajectory [16]. They assume that the models of the robot and the environment are available, and therefore a stable walking pattern can be generated offline.

On the contrary, reinforcement learning techniques aim to adjust the physical actions and motor skills. It allows to the automatic determination of the ideal behavior within a specific context, in order to maximize performance. Simple reward feedback is required for the agent to learn its behavior [17]. Robot bipedal locomotion research such as those by Morimoto *et al.* [18] have improved biped walking controller using an approximated Poincaré map based on reinforcement learning. Their model controls the action between each two single support states for 2-D five-link biped robot with a U-shaped foot. Another study used CMAC as a multivariable function to approximate the Q-factor in the Q-learning to learn the foot placement for the front leg in order to walk with a constant velocity [19]. Reinforcement learning is used also as a subcontrol routine to compensate dynamic reactions of the ground around the ZMP [20].

The main difference between this paper and the above-mentioned works is the fact that we generalize learning of different tasks over varying conditions. Our method is motivated by the functions of ACC and OFC, which build on past experiences without requiring a predefined model of the environment. We propose a technique that works by learning an action-value function to follow a fixed policy by optimizing the energy of the task that keeps record of both positive and, more importantly, negative action consequences.

In this way, we aim to produce an early warning mechanism that can help avoid repeating past errors in the generation of walking patterns of a humanoid robot. It is necessary for such a mechanism to experience mistakes, as well as experience success, in order to evaluate new situations before taking any decision and performing the next action. The notion of reward adaptation is introduced in order to qualify the walking task in term of energy. The notion of adaptive vigilance threshold is also introduced; the tolerance to risk is modulated to be sure to have the same experience for success as for failure, which makes the system converge. Selection with a qualitative adaptive reward allows us to not only determine the state space of parameters in the zone of success but also to optimize the learned task. It is used to adapt the intrinsic parameters of a low level controller based on a CPG for walking on flat and sloped terrains. Experimental validation was conducted on a NAO humanoid robot [21].

The motivation of this paper is to put forward better models based on biologically plausible mechanisms [22]. This may or may not agree with all members of the research community, but this is the general direction of our research. In this paper, we highlight the importance of the different brain mechanisms and how they have been able to influence the development of real robotic control. To further carry this paper forward, we have to match the functions of the mechanisms to additional brain studies [e.g., functional nuclear magnetic resonance (FMRI) studies].

The rest of this paper is structured as follows. Section II presents the principles of our learning mechanism in detail, and then introduces the concept of vigilance. The interest of adapting vigilance is presented. The concept of qualitative adaptive reward is described. Section III details a biologically inspired neural controller for locomotion based on CPG. Three intrinsic parameters of this low level controller are studied by the proposed learning mechanism. In Section IV, we apply the proposed method on a humanoid robot in order to make it learn to walk on flat terrain. Learning to walk on sloped terrain is presented in Section V, where we focus on switching between different sloped terrains based on past experiences and sensory feedback. Finally, conclusions are given in Section VII.

II. Success–Failure Learning

The objectives of this learning mechanism are adapting the parameters of a low-level controller and detecting their domain of viability. We designate by Ω the state space of those influential parameters. The mechanism must be able to learn from negative feedback (failure) and positive

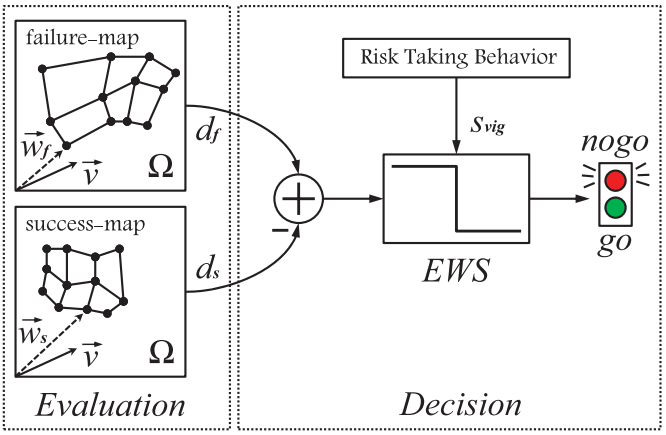

Fig. 2. Success–failure learning mechanism with evaluation and decision phases.

feedback (success). Therefore it must have experience of success and experience of failure in the state space Ω. As each action vector $\overrightarrow{v}$ from Ω leads to either success or failure, the mechanism will evaluate whether this vector belongs to a success case or to a failure case. The decision mechanism "go" or "no-go" described in [23] works as an early warning system similar to that in the ACC [10], [14]. The learning architecture is then based on these two mechanisms, and works as shown in Fig. 2.

A. Success–Failure Evaluation

To represent the knowledge in success and in failure, we define two independent neural networks which are the well-known SOMs proposed by Kohonen [24]. SOMs are widely used for classification and for visualization of high-dimensional data [25]–[27]. Success map S_m learns in case of success trials, and failure map F_m learns in case of failure trials. During the learning, the two maps will be self-organized in the state space which will be therefore divided into three zones: 1) a zone of success represented by success map; 2) a zone of failure represented by failure map; and 3) a zone of conflict that corresponds to the overlap between the two maps. The evaluation of any vector $\overrightarrow{v}$ from space Ω belonging to success or failure is defined by the distance between $\overrightarrow{v}$ and each map. The distance of a vector with a map is the minimal Euclidean norm between this vector and the closest neuron's weights vector in the state space (the winning neuron). For each $\overrightarrow{v}$ we have therefore two distances: one to S_m called d_s, and another to F_m called d_f. d_s and d_f are then used for the decision process.

B. Decision Mechanism

For a vector $\overrightarrow{v}$, the comparison between the distance d_s with the success map and the distance d_f with the failure map leads to an expected result in the case where the vector was passed to the low-level controller (trial). According to expected results, if it may lead to failure, then an early warning signal (EWS) becomes active to avoid passing into the lower level, and the decision will be "no-go." When the EWS is inactive, the decision is "go." The decision mechanism is affected by the threshold of vigilance s_{vig}, which will be detailed later.

C. Learning Algorithm

Success and failure maps represent the knowledge in success and in failure inside the state space. Maps will be initialized in the state space Ω. Then we take one vector $\overrightarrow{v}$ randomly from this space. In the phase of evaluation, we calculate the distance between this vector and all the neurons of both maps. In (1), $\overrightarrow{d}^{\,i}_{s}$ is the distance between $\overrightarrow{v}$ and the ith neuron in the success map, $\overrightarrow{w}^{\,i}_{s}$ is the weight vector of this neuron, $\overrightarrow{d}^{\,i}_{f}$ is the distance between $\overrightarrow{v}$ and the ith neuron in the failure map, and $\overrightarrow{w}^{\,i}_{f}$ is the weight vector of this neuron. For each map, the winner neuron corresponds to the smallest distance to the vector.

In the decision phase, we compare d_s with d_f, by taking into account the threshold of vigilance s_{vig} (see Section II-D), which represents the tolerance to risks. If the threshold is higher than the difference between the distance to failure map and the distance to success map, the EWS becomes active; otherwise, this signal is inactive, see (3).

The activation of the EWS indicates that $\overrightarrow{v}$ will lead to failure if it is passed into the lower level. As maps are in the learning phase, it is possible that vector $\overrightarrow{v}$ can activate the EWS at a time and inactivate it at another time, because the distances with the neurons changes. A decision of "no-go" corresponds to an active EWS, and a decision of "go" corresponds to an inactive EWS. In the case where decision is "no-go," we take another vector $\overrightarrow{v}$ randomly from Ω, and then look for the expected results by evaluation and decision phases as detailed before. In case where the decision is go ($\overrightarrow{v}$ may lead to success), the vector will be passed into the low-level controller to run a trial. There is a reward R for each trial, either negative (failure) or positive (success). Only one map learns $\overrightarrow{v}$. If the reward is negative, the failure map learns, and, if it is positive, the success map learns. Next, other vectors are randomly taken from Ω and the same steps are executed until the convergence of the maps. The convergence of the map occurs when any new vector $\overrightarrow{v}$ will not cause a marked displacement of the neurons of this map in the parameter space. The displacement can be represented by the sum of squared-weight changes for all the neurons of the map.

The following steps summarize the learning process:

1) $\forall\ (S_m, F_m) \in \Omega$
2) $\forall\ \overrightarrow{v} \in \Omega$
   a) *Evaluation:*
   the distances to the neurons of the two maps

$$\begin{cases} \overrightarrow{d}^{\,i}_{s} = -\overrightarrow{w}^{\,i}_{s} + \overrightarrow{v} \\ \overrightarrow{d}^{\,i}_{f} = -\overrightarrow{w}^{\,i}_{f} + \overrightarrow{v} \end{cases} \tag{1}$$

   the distances to the winners neurons of the two maps

$$\begin{cases} d_s = \min \parallel \overrightarrow{d}^{\,i}_{s} \parallel \\ d_f = \min \parallel \overrightarrow{d}^{\,i}_{f} \parallel \end{cases} \tag{2}$$

b) *Decision:*

$$EWS = \begin{cases} 0 \quad \text{(go)}, & \text{if}(d_f - d_s) > s_{\text{vig}} \\ 1 \ \text{(no-go)}, & \text{otherwise} \end{cases} \tag{3}$$

3) if (no-go) go to 2
else if (go) test $\overrightarrow{v}$, and get a reward R
if (R: positive) learn S_m,
else if (R: negative) learn F_m,
go to 2.

In success–failure learning, the objective is to determine the cloud of success in the state space. Success map can do this only by scanning all the space or by exploring the space around the successful trials. The first solution is eliminated because the number of trials needed for scanning all the state space is huge. Failure map makes learning faster, because it avoids testing not only previously failed tested trials but also their surrounding areas. Even the training vector is randomly selected but the decision phase will reject it before the trial if it is incorporated into the failure map area. As the state space is continuous, the vector will not be repeated; otherwise it will be needed to make precise the "accuracy" with which we can judge that there is repetition for a previously tested vector.

D. Concept of Vigilance

Psychological research suggests that some people are more tolerant to risk than others who are more cautious [28]–[30]. Vigilance is related to human learning in connection with decision making [31]. In the standard psychological assessment of risk taking, people are classified as risk-seeking or risk-averse [32].

In this paper, for robot tasks learning by success and failure maps, we introduced the concept of vigilance in order to control the learning process in the two maps (success and failure) and manage the learning cycle while avoiding or taking risks according to the system's needs.

The vigilance is represented by a threshold s_{vig} that is used to adjust the EWS in the decision mechanism. This threshold describes the tolerance of risk (see Fig. 2). By definition, the threshold of vigilance is the allowed margin of difference between the distances of state space vector $\overrightarrow{v}$ with failure map (d_f) and with success map (d_s), for which the decision mechanism still responds with "go" [see (3)]. The threshold has a limited value according to the dimensions of the state space. As learning occurs inside a unit space [e.g., in a 2-D state space, as in Fig. 3(a)], the maximum difference between d_f and d_s is equal to the diameter of the unit space [$\sqrt{2}$ in Fig. 3(a)], which corresponds to all $\overrightarrow{w}^i_s$ in a corner and all $\overrightarrow{w}^i_f$ in the opposite corner in the unit space, and $\overrightarrow{v}$ is close to $\overrightarrow{w}^i_s$. The minimum difference between d_f and d_s corresponds to $\overrightarrow{w}^i_s$ for all success map neurons in a corner, and $\overrightarrow{w}^i_f$ for all failure map neurons and the randomly selected vector $\overrightarrow{v}$ in the opposite corner. Therefore, the vigilance threshold $s_{\text{vig}} \in [-\sqrt{2}, +\sqrt{2}]$ in the 2-D unit space, and $s_{\text{vig}} \in [-\sqrt{3}, +\sqrt{3}]$ in the 3-D unit space. Therefore, as we move toward positive values of the threshold, the decision mechanism becomes more alert to risk (cautious). In the opposite, it has a tendency to take risks (courageous) [see Fig. 3(b), where D is the diameter of the space].

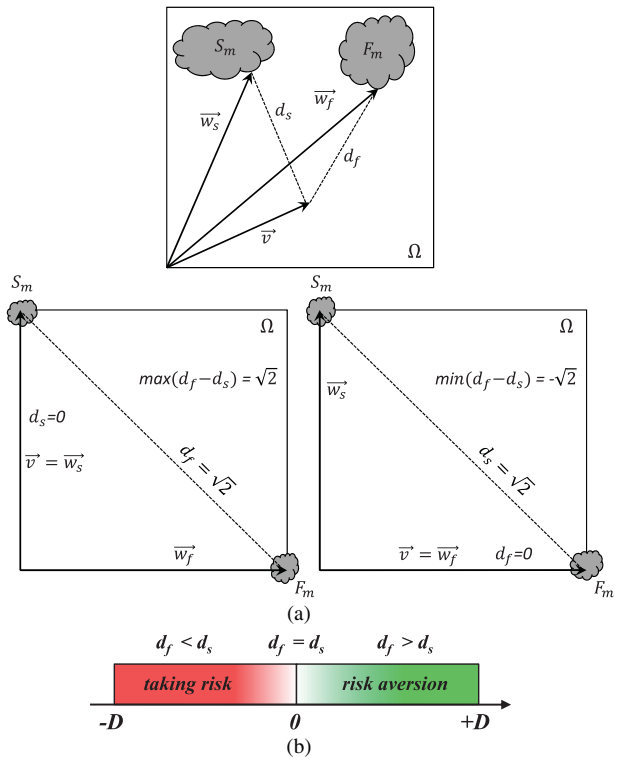

Fig. 3. (a) Distance to the neurons winner of success and failure maps. (b) Tolerance to risk.

For instance, let us suppose that $s_{\text{vig}} = 0.1$, $d_s = 0.3$, and $d_f = 0.35$. According to (3), the EWS becomes active, $\overrightarrow{v}$ will be rejected, another vector will be selected, and then the distances between the two maps will be measured. The randomly selected vector will then be tested on the robot when EWS is inactive.

E. Vigilance Adaptation

Studies show that humans reduce the probability of sampling alternatives with poor past outcomes when learning from experience [33], [34]. They show how adaptive sampling could lead to risk-averse as well as risk-seeking behaviors. Risk tendency may change according to the distribution of the uncertain alternatives and the information about foregone payoffs.

According to the vigilance threshold of success–failure learning, the system can be risky or cautious during learning. Fig. 4(a) shows the successful trial ratio for learning stages with different vigilance thresholds [35]. Learning occurred in the 2-D parameter space of a sensorimotor walking controller [36]. The first, α, denotes the dynamics of rhythmic movement in the hip joint (dynamics of extensor sensor neuron), while the second, θ, represents the amplitude of this movement (amplitude in the activity of extensor sensor neurons). It is to be noted that for a vigilance threshold $s_{\text{vig}} = 0.05$, and after 500 trials, there is 98% success and only 2% failure. As a result, only the success map converges. The area occupied by the success map with cautious behavior will be much smaller

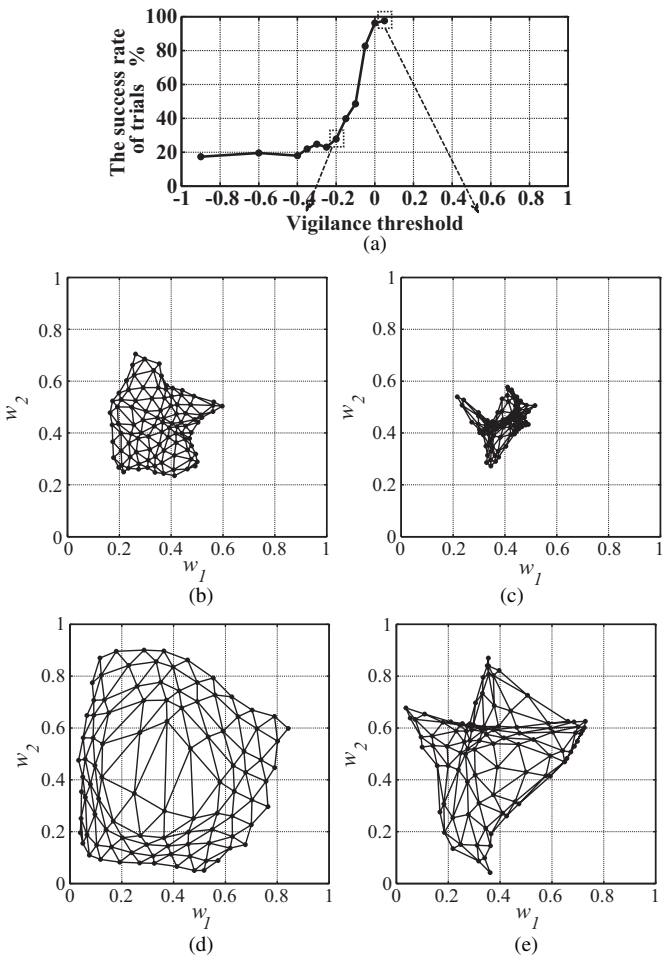

Fig. 4. Success and failure maps after learning on flat terrain with vigilance threshold $s_{\text{vig}} = 0.05$ (right) and $s_{\text{vig}} = -0.2$ (left). (a) Rate of succeeded trials as a function of the vigilance threshold. (b) Success map ($s_{\text{vig}} = -0.2$). (c) Success map ($s_{\text{vig}} = 0.05$). (d) Failure map ($s_{\text{vig}} = -0.2$). (e) Failure map ($s_{\text{vig}} = 0.05$).

than the area occupied by the success map with more risky behavior in term of vigilance threshold, see Fig. 4(c) and (b). On the other hand, with a vigilance threshold $s_{\text{vig}} = 0.05$ and the system avoiding risk, the failure map was not able to self-organize in the parameter space Ω, largely due to the lack in the number of failed trials, as the input vectors were not sufficient for learning, Fig. 4(e). On the contrary, as shown in Fig. 4(d) about taking risks, the rate of failure is more than 70%. With a smaller vigilance threshold, the system takes risks considerably, and the decision mechanism tends to accept all proposed vectors from Ω to be tested on the robot. Otherwise, no more selection occurs on the proposed pattern, which justifies the saturation on the left side in Fig. 4(a).

Therefore it is important to modulate the vigilance threshold to ensure success and failure maps learn together, converge, and avoid the saturation areas in Fig. 4(a).

For instance, the number of successful and failed trials can be used to influence risk-taking and risk-avoiding behaviors. Increasing the current vigilance threshold if the number of failed trials is greater than the number of successful trials will lead the system to risk-avoiding behavior. Decreasing that threshold if the number of failed trials is smaller than that for succeeded trials will lead to risk-taking behavior.

F. Qualitative Adaptive Reward Learning (QARL)

In the proposed success–failure learning, the success map learns all successful trials with the same importance. However, successful trials can be qualified differently according to a desirable criterion. The objective is to influence learning by the trial quality. This can be done by introducing the quality of trial as a weighted reward into the map. Each trial will have its own weighted reward representing the objective criterion to be optimized. During each learning step, neurons will get closer to trials with high rewards rather than to trials with low rewards. After enough number of trials, this will result in a shift of the map into a spatial area associated with the highest rewards.

The quality of a trial $\eta(k)$ is expressed as a number ranging from $\eta_{\min}$ to $\eta_{\max}$. However, this range cannot be determined at the beginning of learning. This is because no previous experience, neither for success nor for failure, is available at the beginning.

Most reinforcement-learning-based robotic walking studies use predefined constants to determine the maximum and the minimum reward or to determine the multiplier factors [37], [38]. In their definition of the reward function, maximum and minimum values are used to normalize the rewards [37]. These parameters represent the minimum and maximum score for walking speed and for the zero moment point, which cannot be estimated without extensive experiments on the robot [37]. One of the challenges is to adjust these parameters automatically and adapt them by learning. Therefore, adaptation is needed to redetermine the range limits $\eta_{\min}$ to $\eta_{\max}$ after each trial. Let us denote the input data by a n-dimensional vector $\upsilon(k) = [\zeta_1(k), \zeta_2(k), \ldots, \zeta_n(k)]$. Here, k is the index of input data in a trial sequence. Let weights vector for the ith neuron in the map be $w_i(k) = [\mu_{i1}(k), \mu_{i2}(k), \ldots, \mu_{in}(k)]$, where k denotes the index in the sequence in which the neurons are generated. The updated weights vector $w_i(k+1)$ is calculated as

$$w_i(k+1) = w_i(k) + \gamma(k).h_{\text{ci}}(k).\rho(k).[\upsilon(k) - w_i(k)] \quad (4)$$

where $\gamma(k)$ is the learning rate which is a scalar factor that defines the size of the correction. Its value decreases with the step index k. The index i refers to the neuron under processing, and c is the index of the neuron winner [that has the smallest distance from the input vector $\upsilon(k)$].

The factor $h_{\text{ci}}(k)$ is the neighborhood function. It is equal to 1 when $i = c$ and its value decreases when the distance between the neuron w_i and w_c increases (e.g., one choice for a neighborhood function is to use a Gaussian kernel around the winning neuron).

The factor $\rho(k)$ denotes the qualitative adaptive reward of $\upsilon(k)$ which is computed iteratively as

$$\rho(k) = \begin{cases} \rho_{\max} & k = 0 \\ \frac{\rho_{\max} - \rho_{\min}}{\eta_{\max} - \eta_{\min}}(\eta(k) - \eta_{\min}) + \rho_{\min} & k > 0 \end{cases} \quad (5)$$

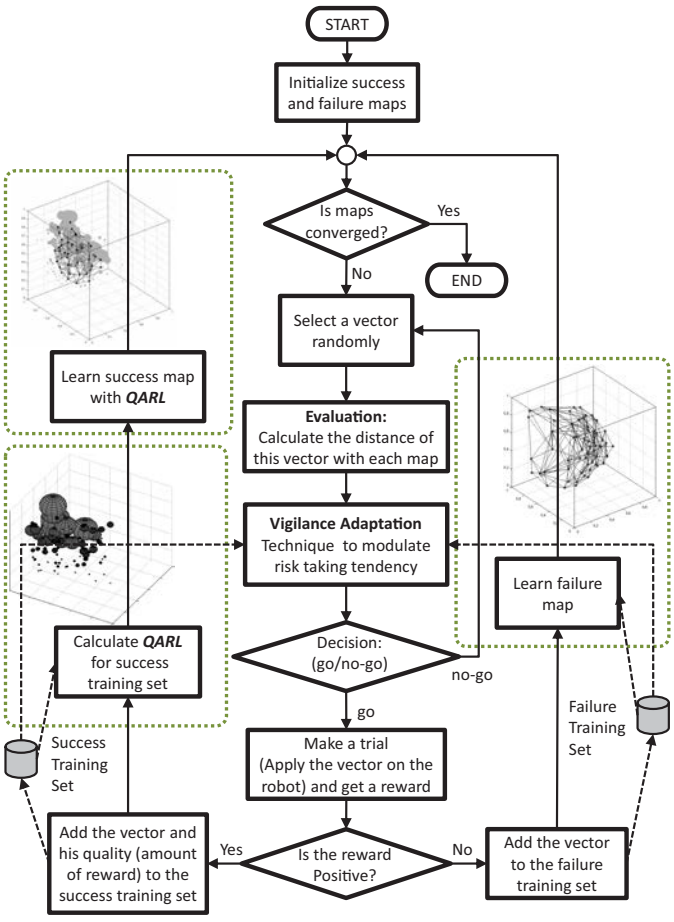

Fig. 5. Flow diagram for success–failure learning with the vigilance adaptation concept and qualitative adaptive reward.

where

$$\begin{cases} \eta(k) = F(v(k)) \\ \eta_{\max} = \max(\eta(k = 0, \ldots, K)) \\ \eta_{\min} = \min(\eta(k = 0, \ldots, K)). \end{cases} \tag{6}$$

The function F allows us to obtain the criterion $\eta(k)$ for the trial that corresponds to $v(k)$. For instance, for a bowling robotic arm, $\eta(k)$ can denote the efficiency of the throw by combining the obtained result and the energy spent by the actuators. K is the index of the current trial. Maximal and minimal rewards $\rho_{\min}$ and $\rho_{\max}$ are predefined from the trainer.

When the success map learns after the first successful trial, the reward will be maximal. After the second successful trial, the trial with highest quality will match the maximal reward, and the trial with the lowest quality will match the minimal reward. A scaling between maximal and minimal rewards will occur for any new successful trial. A trial that matched a high reward at the start of learning phase could match a low reward at the end of learning.

By introducing the concept of QARL, it will be possible to scale the quality of a trial according to the quality in previous experiences even when starting from scratch. After learning, the optimal parameter is presented by the success map neuron that is close to the trial with maximum reward in training set. The general diagram of the proposed technique is presented in Fig. 5.

The SOM has been employed as a clustering technique because it guarantees safe switching between two different behaviors, e.g., some neurons can match highly efficient walking patterns, while others can match patterns with high walking velocity. Intermediary neurons are in charge of such a switching. This can also play a role in having not only one solution for the walking problem but also other possible solutions.

Compared to a k-nearest neighbor algorithm, where an object is classified by a majority vote of its neighbors, this method can be used to calculate the minimum distance between the tentative input v and each map. SOM is employed in success–failure not only to judge the candidate samples but also to represent all tested samples according to their efficiencies.

While K-means clustering is able to classify success and failure and separate them, it is unable to quantify the success because cluster centers do not necessarily match the higher efficiency of the learned task.

The proposed algorithm can be regarded as a policy search method. Different search methods have been proposed previously for reinforcement learning on autonomous robot controllers [39], [40]. Policy gradient method is one of the most accepted approaches. It was widely used in robotics and in walking controller [37], [38]. Policy gradient reinforcement learning (PGRL) is an optimization technique that guarantees the convergence to at least a local optimum, unlike the other RL search methods. The convergence to a global optimum cannot be guaranteed unless starting with the right initial condition; this limits the flexibility of this method as such a dependency cannot be established easily.

Due to the random samplings before the decision phase and due to the vigilance adaptation technique, QARL can guarantee the convergence to all successful clusters in the state space. In addition, the use of SOM helps in representing the successful clusters although they are separated in the state space.

Evolutionary computation methods are widely used in robotics for parameter optimization. The common method of evolutionary computation is genetic algorithms (GAs) [41] that generate solutions to optimization problems using techniques inspired by natural evolution [42]. They are more likely to converge toward a global optimum than PGRL techniques; furthermore, they can solve problems with multiple solutions. However, they have limitations in robotic applications where researchers are interested in the way to get the solution rather than the solution itself in order to build auto-adaptive and autonomous robots. GAs can provide the solution provided that the fitness function is well described beforehand. In QARL, we are interested in the way to the solution in order to build an auto-adaptive algorithm that can adjust the controller parameters in dealing with environmental changes.

As it is based on learning from success and failure trials, the proposed method (QARL) can be considered as a RL method. In other RL methods, both negative and positive rewards can be used, and the difference of the efficiency among the successful cases can also be considered. However, there is an essential difference between the proposed method and other RL methods much like a kind of multiarmed bandit (MAB)

problem. In the MAB problem, an arm can lead to success with some trials and to failure with others trials. MAB is based on the success probability in the building of its prior tree. In QARL, learning and sampling occur in continuous space. Therefore the number of samples for trials is unlimited (not only multiarms). Unlike classical MAB that has stochastic property, in our method a trial that leads to success will never lead to failure, even if it is tested again; and a trial that leads to failure will never lead to success.

When applying an RL algorithm with continuous space, a neural network, such as a multilayer perceptron (MLP), is usually used as a function approximator. Although MLP can distinguish data that are not linearly separable, which is the case between regions of success and region of failure, MLP will not be able to determine success matches with high efficiency from any other region. This qualitative approach is one of the QARL principles.

The concept of qualitative adaptive reward with success–failure learning will be applied to humanoid robots. The humanoid NAO robotic platform is used in our experiments. Based on QARL, the robot learns to walk on flat terrain and constructs the experience for success and for failure. Then, learning to walk on sloped terrain will be presented, and the robot will construct its experiences in walking on sloped terrain. The objective is to achieve success–failure learning in a space of intrinsic parameters of a low-level controller for locomotion.

III. Bioinspired Neural Control for Locomotion

Biological evidence suggests that locomotion is mainly generated at the spinal cord, by a combination of a CPG and reflexes receiving adjustment signals from the brain, particularly from the cerebrum and the cerebellum [43]–[45]. Locomotion is the result of the dynamic interaction between the CPG and the connected feedback mechanisms. It has been shown that the CPG is able to generate basic locomotor patterns according to the descending pathways that can control the locomotion tasks [46]. The feedback that dynamically adapts the locomotor pattern to the environment originates from muscles and skin afferents, as well as from the basic senses (vision, audition, vestibular). The CPG is a neural mechanism that can produce rhythmic patterned outputs without rhythmic sensory or central inputs [47], [48]. It can generate periodic motor commands for rhythmic movements such as locomotion [49]. Studies also have shown that the CPGs are localized in the lower thoracic and lumbar regions of the spinal cord [50]. These aforementioned studies have been taken into account in the designing of the robot's locomotion gait in order to realize a mechanism for robust locomotion, especially on legged robots [38], [51]–[54]. Different models of neural oscillators are widely used to generate rhythmic motion [55]–[59]. Such oscillations generated by two mutually inhibiting neurons are described by a set of differential equations (e.g., a Matsuoka oscillator [55]). Rowat and Selverston's [60] model of rhythmic neuron can generate different types of patterns, not only oscillatory ones. The membrane currents of the neuron in this model are separated into two classes, fast and slow, in accordance with their time responses. The sum of all fast currents is modeled by a single fast signal, and a single slow current is used to model the sum of all slow signals. This model has two differential equations: one for membrane potential V, derived from current's conservation; and one for lumped slow current q, derived from current's activation, as

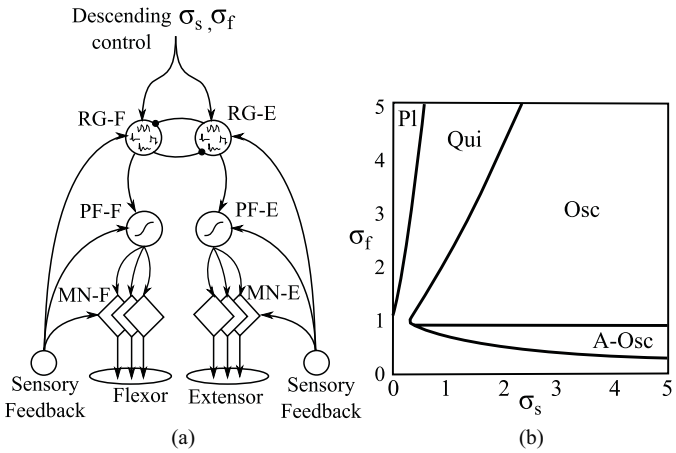

Fig. 6. Model of one joint controller and its motion patterns. (a) Model's scheme. CPG with three levels: rhythm generator, pattern formation, and motor neuron level. (b) Different intrinsic behaviors observed on a joint according to parameters of rhythmic neuron (σ_s, σ_f): quiescence (Qui), almost an oscillator (A-Osc), oscillator (Osc), and plateau (Pl).

$$\tau_m . \frac{dV}{dt} = -(\text{fast}(V, \sigma_f) + q - i_{inj}) \tag{7}$$

$$\tau_s . \frac{dq}{dt} = -q + q_\infty(V) \tag{8}$$

where τ_m is the membrane time constant for the fast current, and τ_s is the time constant for the slow current. The ratio of τ_s to τ_m is about 20 as in [60]. In this paper, $\tau_m = 0.05$, and $\tau_s = 1$ for all rhythmic neurons. The injected current is i_{inj}. An idealized current–voltage curve for the lumped fast current is given by $\text{fast}(V, \sigma_f) = V - A_f.\tanh((\sigma_f / A_f)V)$. A_f is the width of the N-shape in the fast current–voltage curve. The fast current can represent the sum of a leak current and an inward Ca^{++}. The dimensionless shape parameter for current–voltage curve is given by $\sigma_f = (g_{\text{Ca}}/g_L)$. g_L is a leak conductance and g_{Ca} is the calcium conductance. $q_\infty(V)$ is the steady-state value of the lumped slow current, which is given by $q_\infty(V) = \sigma_s(V - E_s)$. $q_\infty(V)$ is linear in V with a reversal potential E_s. σ_s is the potassium conductance g_K normalized to g_L. σ_s is given by $\sigma_s = (g_K/g_L)$. q and i_{inj} have the dimension of an electrical potential. A true current is obtained by multiplying the model current by a leak conductance g_L. V, E_s, i_{inj}, and q are given in millivolts while τ_s and τ_f are expressed in milliseconds. With different values of the modeling parameters, different intrinsic behaviors can be achieved: quiescence, almost an oscillator, endogenous oscillator, depolarization, hyperpolarization, and plateau. In this paper, as we are interested in bipedal walking, which is periodic, only oscillatory patterns will be used, but different behaviors in the activity of these neurons can be used in robot's locomotion to achieve different locomotion tasks such as asymmetrical gaits. Fig. 6(a) shows the wiring diagram for one

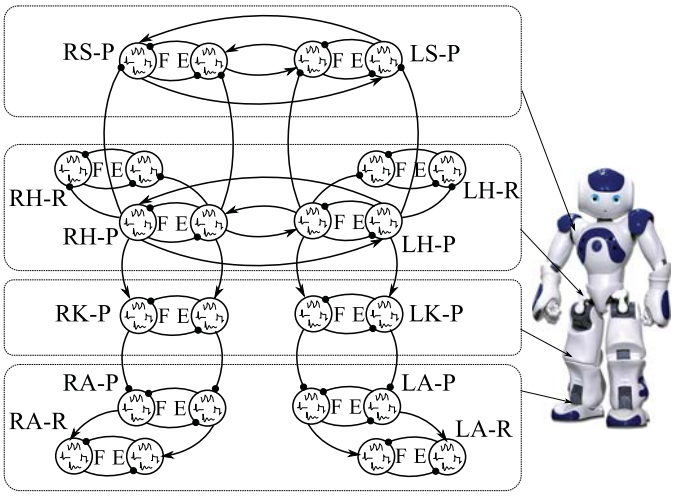

Fig. 7. Coupling circuitry between rhythm generator neurons. F stands for flexion neuron and E for extension neuron. RS-P and LS-P are right and left pitch shoulder rhythmic neurons. LH-P and LH-R are pitch and roll rhythmic neurons for the left hip.

robot's joint. It can be separated into three layers: 1) rhythm generation neurons (RG); 2) pattern formation neurons (PF); and 3) motor neurons (MN). Sensory feedback shapes the activity of these neurons. In the analytical study, after observing the phase relationship of a joint while altering σ_s and σ_f in the rhythm generator neurons, different motion behaviors were observed at the joint. Fig. 6(b) shows the distribution of motion patterns in space of σ_s and σ_f. Varying σ_s and σ_f in the RG of a joint will change its motion pattern. The four detected basic motion patterns can lead the robot to achieve some complex movements depending on synaptic circuits between joint CPGs [1]. Walking gaits can be composed from basic synchronized patterns. The synchronization between patterns is ensured by coupling the joints' CPGs. Fig. 7 shows the proposed coupling circuits between the rhythm generator neurons for the hip pitch and roll, the knee pitch, and the ankle pitch and roll, and the shoulder pitch joints of a NAO humanoid robot. With such simple coupling, the robot can carry out walking task from basic oscillatory patterns. With different coupling circuits, another task can be achieved. The principle of our proposed circuit for walking is described by the activity between the CPGs, which is regulated by excitatory synaptic connections [see Fig. 7]. For example, the RG neuron extensor in the left hip pitch (LH-P E) excites the RG neuron flexor in the right hip pitch (RH-P F), and inhibits the RG neuron extensor in the left hip roll (LH-R E) and the RG neuron extensor in the left shoulder pitch (LS-P E).

IV. Learning to Walk

In this section, we apply the architecture proposed in the previous sections, as conceptually presented in Fig. 1, in order to learn efficient walking for a bipedal humanoid robot, i.e., NAO.

A. Walking Efficiency

We used success–failure learning with QARL to learn in a space of intrinsic parameters of the CPG controller (motor neuron gain in pitch, motor neuron gain in roll, and the dynamics of rhythmic generator neurons represented by σ_s). The optimization of walking efficiency was studied in term of energy as in [61].

Most of biomechanics studies on human movement focus on the efficiency of movement [61]. During flexion and extension of the joints, muscles release and absorb mechanical energy. When a muscle is exerting an active force and is being lengthened by external forces at the same time, the mechanical energy is absorbed, and muscle is said to do negative work. It is said to do positive work when the muscle is shortening as it develops a force. The efficiency with which a muscle operates is defined in [61] by

$$\text{efficiency} = \frac{\text{mechanical work done}}{\text{metabolic energy consumed}} \tag{9}$$

where the mechanical work done on the muscle is considered as negative, while that done by the muscle is positive. The metabolic energy consumed by a muscle is generally defined as the entirety of its chemical processes [62]. This paper is also generalized from a muscle to whole body movements like walking and running [63], [64].

Inspired by biomechanical studies, the efficiency of walking for a humanoid robot can be described in a similar fashion. In this case, the mechanical work done is the robot's displacement energy during walking while the metabolic energy consumed can be represented by the energy consumed by actuators as in (9).

B. QARL in Humanoid Walking

Our objective is to simultaneously learn and optimize walking. The robot learns to walk a 1.5-m trajectory with start and end lines. In case of successful trials, the trainer sends a reward signal to the robot by caressing the head equipped with electrostatic sensors. Electric power is calculated at each instant as

$$P(t) = \sum_{i=1}^{n} R_i . I_i^2 \tag{10}$$

where n is the number of electric motors. I_i and R_i are the electric current and the electric resistance for motor number i. The required electric E_e energy for all the trajectory is expressed as

$$E_e = \int_{t=t_0}^{T} P(t).dt \tag{11}$$

where t_0 is the trial start time, and T is the trial end time, i.e., when the robot reaches the finish line. The kinetic energy of a trial is given by

$$\begin{cases} E_k = \frac{1}{2}.m.v_a^2 \\ v_a = \frac{\Delta d}{\Delta t} \end{cases} \tag{12}$$

where v_a is the average velocity for the entire trajectory, Δd is the trajectory length, Δt is the time difference between start and end of a trial, and m is the robot's mass. The walking efficiency is calculated for each trial as

$$\eta = \frac{E_k}{E_e}. \tag{13}$$

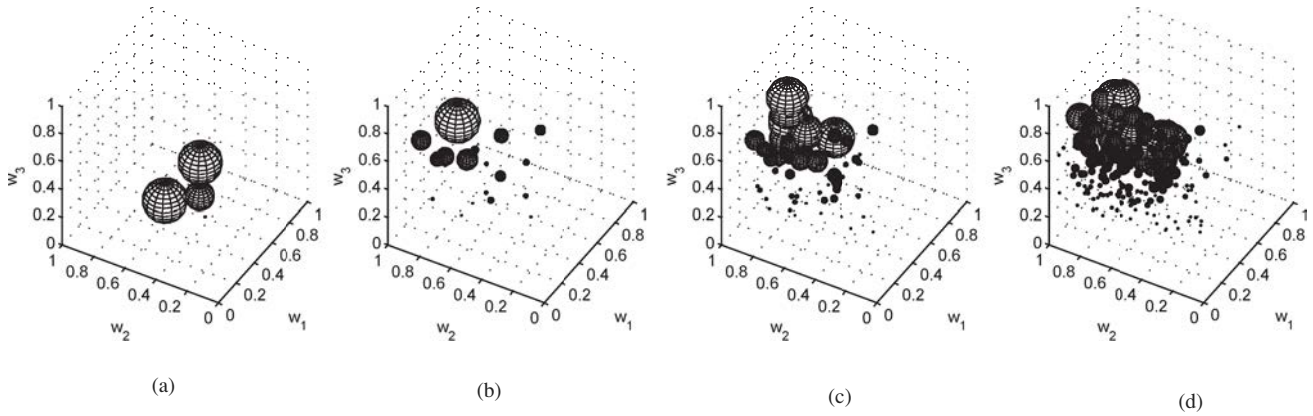

Fig. 8. Successful trials' reward related to walking efficiency for learning success map. w_1 is motor neuron gain in pitch, w_2 is motor neuron gain in roll, and w_3 is σ_s which is related to the oscillation frequency. (a) Reward after fourth success. (b) Reward after 50 trials. (c) Reward after 150 trials. (d) Reward after 500 trials.

The introduction of the efficiency for success map learning will shift the neurons of this map into the area in which the walking efficiency is high. This is done by using the concept of QARL. Fig. 8 shows the QARL for success map in the beginning of learning (after four successful trials) and at the end of learning. Each sphere corresponds to a successful trial whose diameter represents the reward of this trial in the success map. It is to be noticed that the trial corresponding to the maximum reward at the start of learning, indicated by a circle, will have a small reward at the end of learning. The interest of using this technique is to make success–failure learning search for new trials in the space area where walking efficiency in term of energy is high. In other words, this leads to learn and optimize in a defined space. Fig. 9 shows success maps after learning to walk on flat terrain with and without the technique of qualitative adaptive reward. In Fig. 9(a), the success map learns all successful trials with the same opportunity, i.e., with the same reward. In Fig. 9(b), the success map learns successful trials in accordance with their qualitative adaptive rewards. Trials with high reward influence success map neurons more than trials with low reward. Therefore, the success map will be attracted to the area where the reward is high. This is influenced by the differences between highest and lowest rewards (scaling range limits: $[\rho_{\min}, \rho_{\max}]$), see (5). In this paper, $\rho_{\min}$ and $\rho_{\max}$ are set to 0.1 and 2.5. The application of QARL influences the success map neurons to match more efficient patterns in the studied space. Some walking patterns represented by success map neurons learned without QARL show less efficient walking. These effects were reduced when QARL was applied. Regarding the learning frameworks with and without the application of QARL shown in Fig. 9, performance was increased by 60% after applying QARL. This was calculated by the ratio of the highest efficiency neurons in both success maps (with and without QARL). The ratio of the lowest efficiency of the neurons of success maps has increased by 40%. In order to provide sufficient precision in the network for our task, we have empirically selected a $5 \times 5 \times 5$ dimensional network space to represent the success and failure maps. Learning occurred with 500 trials for each case. Without applying the auto-adjustable vigilance technique, the number of successful trials has increased 16% after applying QARL. Computationally, all the processing of this learning framework in simulations as well as on the real robot can be performed in real time, thus making our approach feasible for training on the real robot. Within the same cycle, joint angle commands are calculated in real time and sent to joint motor circuit boards of NAO every 10 ms. This is done inside a high-priority thread on the robot. Physically, each trial requires about 3 min, which includes learning and the experimental setup. A complete learning session in the robot usually takes about one week.

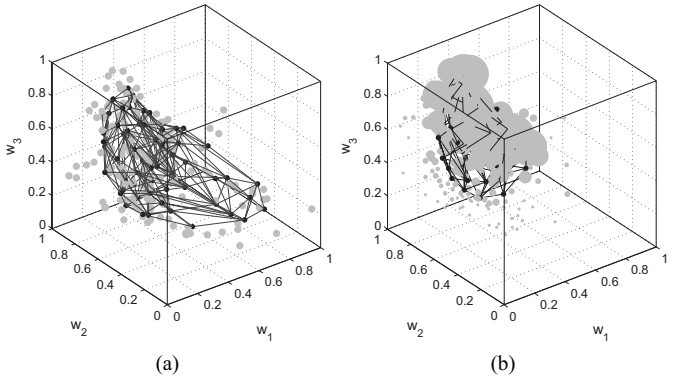

Fig. 9. Effect of QARL on success map. (a) Success map after learning with the same reward for all successful trials. (b) Success map after learning with adaptive reward. Gray spots represent successful trials reward. Note that the map on the right moves into the area where rewards are high (representing high efficiency).

Both learning frameworks shown in Fig. 9 start from scratch. After 200 trials, we noticed that the rate of success to failure when applying QARL is higher than without it. However, the rate of success can be increased by controlling the threshold of vigilance. This is the objective of the next section.

C. Adaptive Vigilance in Humanoid Walking

The vigilance threshold is auto-adjusted in order to have the same experience for success as for failure according to Algorithm 1.

Algorithm 1 Vigilance Adaptation

$\forall S_{\text{vig}} \in [-D, +D]$ (initialization)
if $(N_s > N_f)$ then take risks : $S_{\text{vig}} = S_{\text{vig}} - \text{step}$
elseif$(N_s < N_f)$ then avoid risks : $S_{\text{vig}} = S_{\text{vig}} + \text{step}$
else no change

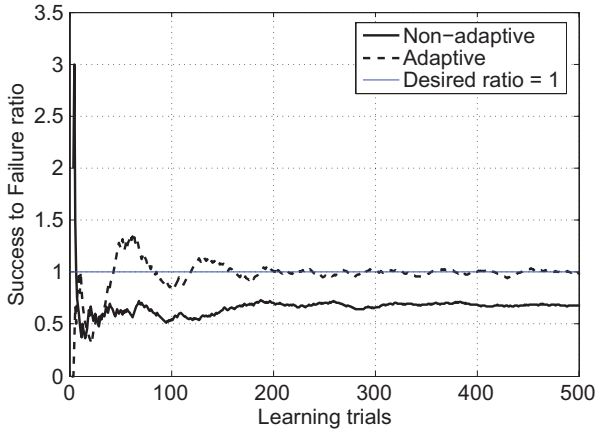

Fig. 10. Success to failure ratio with and without adaptive vigilance in learning to walk on flat terrain.

Here, N_s and N_f denote the number of successful and failed trials, respectively. step describes the change of vigilance threshold to have a desired behavior for risks. It is defined by training, step $= 0.01$ in this paper. D is the diameter of the space ($D = \sqrt{3}$ in the 3-D unit space).

When the success to failure ratio is always less than 1, the threshold of vigilance will gradually increase until a new threshold value that leads to EWS activity for all randomly generated vectors (in our experiment after 1000 samples have been rejected sequentially), i.e., no more vectors can realize the condition in the decision making phase when applied on the robot. As a consequence, the threshold of vigilance decreases a step, and then starts the search with random vectors in the space. Decreasing S_{vig} will find executable samples in the space that can be applied on the robot to achieve a trial.

Fig. 10 shows the rate of success and the rate of failure in learning to walk on flat terrain with and without vigilance adaptation. It is to be noted that the success to failure ratio N_s/N_f shows unpredicted changes in the beginning of learning. After 100 learning trials, due to the vigilance threshold adaptation this ratio stays around 1, which contributes to the convergence of the success and failure maps. In case of nonadaptive vigilance, S_{vig} was fixed experimentally to -0.15, and the ratio stabilizes at 0.65. Adapting the vigilance ensures the same size of training sets to learn success map and failure map, because both maps have the same number of neurons (clusters).

V. Learning to Walk on Sloped Terrain

In this paper, the transfer of learning between different walking situations is not addressed. We assume that there is a success map and a failure map for each situation. Two stages of learning have been implemented on 10° upward slope and 10° downward slope. For each condition, learning starts from scratch. Vigilance adaptation and QARL concepts are

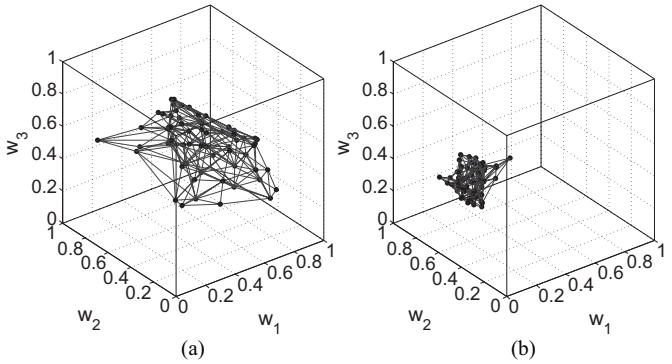

Fig. 11. Success map after learning with reward on sloped terrain. (a) 10° downward slope. (b) 10° upward slope.

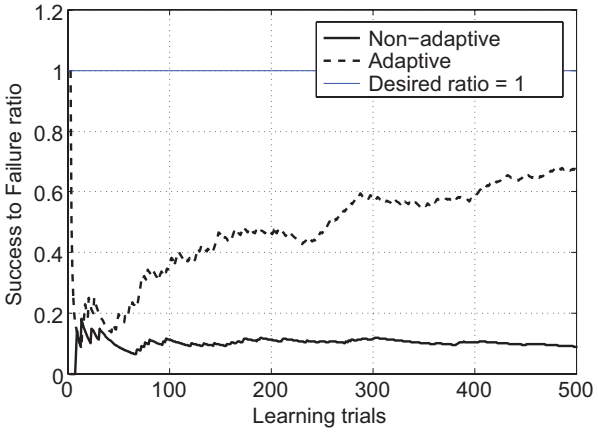

Fig. 12. Success to failure ratio with and without adaptive vigilance in learning to walk on 10° upward slope.

used. The initial angular positions have the same values for all learning stages. Only the ankle pitch joint is initialized from stance position in order to keep the torso pitch around 10° vertically during walking.

Fig. 11 shows success maps after learning to walk downhill on the left, and uphill on the right. The two maps and the map responsible for walking on flat terrain [Fig. 9(b)] occupy different areas in the learning space. It is to be noted that the success map for walking downhill occupies a greater area in the state space than the area occupied by the success map for walking uphill. However, that difference in size does not mean the result is much better; it is mostly related to the complexity of the task (e.g., walking uphill being more difficult than walking downhill, the pattern space for uphill condition is smaller than the pattern space for the downhill condition).

Fig. 12 shows the rate of success and the rate of failure in learning to walk on inclined uphill terrain with and without vigilance adaptation.

A. Vigilance Adaptation

Vigilance is auto-adjusted in order to have the same experience for success as for failure ($N_s = N_f$). This ensures that each map has enough data for training. In case of fixed vigilance $S_{\text{vig}} = -0.15$, the ratio stabilizes around 0.1. This leads only the failure map to converge unlike the success map. The difference between this steady value and that with walking

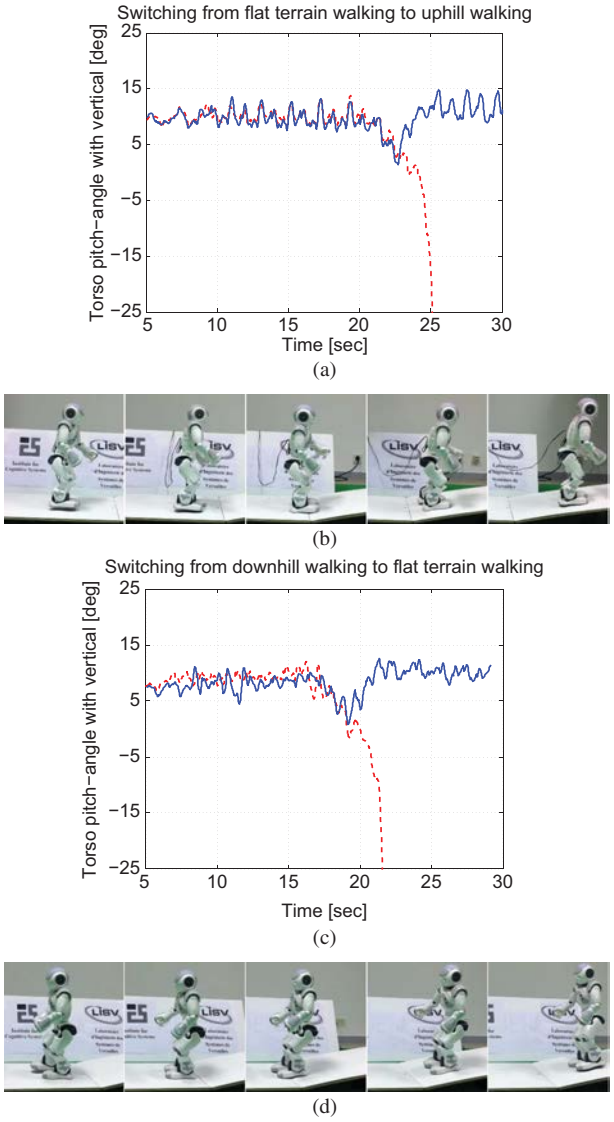

Fig. 13. Walking on different sloped terrains. Switching occurs between success map neurons in order to adapt to the new situation. (a) Torso pitch angle during walking from flat ground to upward slope, with and without the switching. (b) Walking from flat ground to upward slope with switching between success maps neurons. (c) Torso pitch angle during walking from downward slope to flat ground, with and without switching. (d) Walking from downward slope to flat ground with switching between success maps neurons.

on flat terrain proves that learning to walk uphill is more difficult than learning to walk on flat ground. To guarantee success map convergence without vigilance adaptation, too many learning trials are needed. Therefore this delays the convergence. Due to the vigilance adaptation, this ratio looks moving toward 1, and even some more learning trials are needed to reach the wanted ratio. A current development in the proposed algorithm for vigilance adaptation will lead to improvement in the convergence speed of success to failure ratio into the desired value.

VI. Switching Between Different Sloped Terrains—Exploiting Learned Experiences

This part shows how to walk on different terrain slopes and to switch between them, by exploiting previously learned experiences. Inertial sensors are used to detect the change of terrain slope during walking. When detection occurs, the walking pattern switches from a success map related to the walk on previous terrain slope to a success map related to the walk on the new terrain slope.

The inertial sensor is also used to adjust the center of oscillation of ankle joints in order to keep the robot torso close to the vertical with a small inclination in the walking direction. For the NAO robot, we keep this angle close to 10° with the vertical direction.

Fig. 13(a) shows torso pitch angle during the walk on different slopes, switching from flat ground to an uphill inclined terrain. Without using this technique the robot falls (indicated by the dashed line). As a compensation technique, switching occurs between a neuron in the success map responsible for walking on flat terrain into a neuron of another success map responsible for walking on inclined terrain. Therefore, the robot succeeds to continue walking on the new uphill terrain. Fig. 13(c) shows the torso pitch angle during switching from downhill to flat terrain. When the torso pitch angle reaches a predefined threshold, switching occurs gradually between a neuron of success map responsible for walking on downhill and a neuron of success map responsible for walking on flat terrain. Fig. 13(b) and (d) shows snapshots of a NAO humanoid robot achieving the walking task on different terrain slopes and switching between them (a video is available on: http://web.ics.ei.tum.de/ nassour/naowalking.wmv.).

VII. Conclusion

This paper proposed a neurobiological-inspired learning algorithm. The notion of qualitative adaptive reward was introduced in order to simultaneously learn and optimize the task. The objectives of the mechanism were to learn from mistakes and to avoid making them again. This was done by building on experiences of past mistakes and successes. We showed how these two experiences could build themselves through the stages of evaluation, decision, and then trials. Learning successful trials with reward related to walking efficiency makes success map match trials where the efficiency is high. The adaptive vigilance technique allows having an experience to success as to failure. It can be said that the negative reward is as important as positive reward. This mechanism was implemented and validated on an NAO humanoid robot, which allowed it to learn to walk on flat ground as well as sloped terrain.

References

[1] J. Nassour, P. Hénaff, F. B. Ouezdou, and G. Cheng, "A study of adaptive locomotive behaviors of a biped robot: Patterns generation and classification," in *Proc. 11th Int. Conf. Simul. Adapt. Behavior: Animals Animats*, 2010, pp. 313–324.

[2] A. Fairhall and W. Bialek, "Adaptive spike coding," in *The Handbook of Brain Theory and Neural Networks*, M. Arbib, Ed., 2nd ed. Cambridge, MA: MIT Press, 2002.

[3] S. D. Iversen and M. Mishkin, "Perseverative interference in monkeys following selective lesions of the inferior prefrontal convexity," *Experim. Brain Res.*, vol. 11, no. 4, pp. 376–386, 1970.

[4] S. J. Thorpe, E. T. Rolls, and S. Maddison, "The orbitofrontal cortex: Neuronal activity in the behaving monkey," *Experim. Brain Res.*, vol. 49, no. 1, pp. 93–115, 1983.

[5] L. Tremblay and W. Schultz, "Relative reward preference in primate orbitofrontal cortex," *Nature*, vol. 398, no. 6729, pp. 704–708, Apr. 1999.

[6] S. Kobayashi, O. P. de Carvalho, and W. Schultz, "Adaptation of reward sensitivity in orbitofrontal neurons," *J. Neurosci.*, vol. 30, no. 2, pp. 534–544, Jan. 2010.

[7] L. Tremblay and W. Schultz, "Modifications of reward expectation-related neuronal activity during learning in primate orbitofrontal cortex," *J. Neurophysiol.*, vol. 83, no. 4, pp. 1877–1885, 2000.

[8] T. Singer, B. Seymour, J. O'Doherty, H. Kaube, R. J. Dolan, and C. D. Frith, "Empathy for pain involves the affective but not sensory components of pain," *Science*, vol. 303, no. 5661, pp. 1157–1162, Feb. 2004.

[9] M. Cohen, A. Heller, and C. Ranganath, "Functional connectivity with anterior cingulate and orbitofrontal cortices during decision-making," *Cognit. Brain Res.*, vol. 23, no. 1, pp. 61–70, 2005.

[10] J. W. Brown and T. S. Braver, "A computational model of risk, conflict, and individual difference effects in the anterior cingulate cortex," *Brain Res.*, vol. 1202, pp. 99–108, Apr. 2008.

[11] R. B. Mars, M. G. Coles, M. J. Grol, C. B. Holroyd, S. Nieuwenhuis, W. Hulstijn, and I. Toni, "Neural dynamics of error processing in medial frontal cortex," *Neuroimage*, vol. 28, no. 4, pp. 1007–1013, Dec. 2005.

[12] W. J. Gehring, M. G. H. Coles, D. E. Meyer, and E. Donchin, "The error-related negativity: An event-related potential accompanying errors," *J. Psychophysiol.*, vol. 27, p. S34, Apr. 1990.

[13] J. Hohnsbein, M. Falkenstein, and J. Hoorman, "Error processing in visual and auditory choice reaction tasks," *J. Psychophysiol.*, vol. 3, p. 32, May 1989.

[14] J. W. Brown and T. S. Braver, "Learned predictions of error likelihood in the anterior cingulate cortex," *Science*, vol. 307, pp. 1118–1121, Feb. 2005.

[15] L. Van Leijenhorst, P. M. Westenberg, and E. A. Crone, "A developmental study of risky decisions on the cake gambling task: Age and gender analyses of probability estimation and reward evaluation," *Develop. Neuropsychol.*, vol. 33, no. 2, pp. 179–196, 2008.

[16] C. Sabourin, O. Bruneau, and G. Buche, "Control strategy for the robust dynamic walk of a biped robot," *Int. J. Robot. Res.*, vol. 25, pp. 843–860, Sep. 2006.

[17] R. S. Sutton and A. G. Barto, *Reinforcement Learning: An Introduction*. Cambridge, MA: MIT Press, 1998.

[18] J. Morimoto, J. Nakanishi, G. Endo, G. Cheng, C. G. Atkeson, and G. Zeglin, "Poincaré-map-based reinforcement learning for biped walking," in *Proc. IEEE Int. Conf. Robot. Autom.*, Barcelona, Spain, Mar. 2005, pp. 2381–2386.

[19] C.-M. Chew and G. A. Pratt, "Dynamic bipedal walking assisted by learning," *Robotica*, vol. 20, no. 5, pp. 477–491, 2002.

[20] D. Katić and M. Vukobratović, "Control algorithm for biped walking using reinforcement learning," in *Proc. 2nd Serbian-Hungarian Joint Symp. Intell. Syst.*, 2004, pp. 1–12.

[21] D. Gouaillier, V. Hugel, P. Blazevic, C. Kilner, J. Monceaux, P. Lafourcade, B. Marnier, J. Serre, and B. Maisonnier, "Mechatronic design of NAO humanoid," in *Proc. IEEE Int. Conf. Robot. Autom.*, Dec. 2009, pp. 769–774.

[22] G. Cheng, S.-H. Hyon, J. Morimoto, A. Ude, J. G. Hale, G. Colvin, W. Scroggin, and S. C. Jacobsen, "CB: A humanoid research platform for exploring neuroscience," *Adv. Robot.*, vol. 21, no. 10, pp. 1097–1114, 2007.

[23] K. Matsumoto, W. Suzuki, and K. Tanaka, "Neuronal correlates of goal-based motor selection in the prefrontal cortex," *Science*, vol. 301, no. 5630, pp. 229–232, 2003.

[24] T. Kohonen, *Self-Organizing Maps* (Information Sciences), vol. 30. Berlin, Germany: Springer-Verlag, 1995.

[25] A. Iosifidis, A. Tefas, and I. Pitas, "View-invariant action recognition based on artificial neural networks," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 23, no. 3, pp. 412–424, Mar. 2012.

[26] N. Manukyan, M. Eppstein, and D. Rizzo, "Data-driven cluster reinforcement and visualization in sparsely-matched self-organizing maps," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 23, no. 5, pp. 846–852, May 2012.

[27] S. Ulbrich, V. de Angulo, T. Asfour, C. Torras, and R. Dillmann, "General robot kinematics decomposition without intermediate markers," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 23, no. 4, pp. 620–630, Apr. 2012.

[28] J.-L. van Gelder, R. E. de Vries, and J. van der Pligt, "Evaluating a dual-process model of risk: Affect and cognition as determinants of risky choice," *J. Behav. Decision Making*, vol. 22, no. 1, pp. 45–61, 2009.

[29] B. Pawlowski, R. Atwal, and R. I. M. Dunbar, "Sex differences in everyday risk-taking behavior in humans," *Evol. Psychol.*, vol. 6, no. 1, pp. 29–42, 2008.

[30] P. Horvath and M. Zuckerman, "Sensation seeking, risk appraisal, and risky behavior," *Personal. Individual Differences*, vol. 14, no. 1, pp. 41–52, 1993.

[31] H. Ahn and R. W. Picard, "Affective-cognitive learning and decision making: A motivational reward framework for affective agent," in *Proc. 1st Int. Conf. Affect. Comput. Intell. Interact.*, Oct. 2005, pp. 1–8.

[32] X. Wang, D. Kruger, and A. Wilke, "Toward the development of an evolutionarily valid domain-specific risk-taking scale," *Evol. Psychol.*, vol. 5, no. 3, pp. 555–568, 2007.

[33] J. Denrell, "Adaptive learning and risk taking," *Psychol. Rev.*, vol. 114, no. 1, pp. 177–187, 2007.

[34] J. G. March, "Learning to be risk averse," *Psychol. Rev.*, vol. 103, no. 2, pp. 309–319, Apr. 1996.

[35] J. Nassour, P. Henaff, F. B. Ouezdou, and G. Cheng, "Experience-based learning mechanism for neural controller adaptation: Application to walking biped robots," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, St. Louis, MO, Oct. 2009, pp. 2616–2621.

[36] T. Geng, B. Porr, and F. Wörgötter, "Fast biped walking with a sensor-driven neuronal controller and real-time online learning," *Int. J. Robot. Res.*, vol. 25, no. 3, pp. 243–259, 2006.

[37] T.-H. S. Li, Y.-T. Su, S.-W. Lai, and J.-J. Hu, "Walking motion generation, synthesis, and control for biped robot by using PGRL, LPI, and fuzzy logic," *IEEE Trans. Syst., Man, Cybern. B, Cybern.*, vol. 41, no. 3, pp. 736–748, Jun. 2011.

[38] G. Endo, J. Morimoto, T. Matsubara, J. Nakanishi, and G. Cheng, "Learning CPG-based biped locomotion with a policy gradient method: Application to a humanoid robot," *Int. J. Robot. Res.*, vol. 27, pp. 213–228, Feb. 2008.

[39] G. Grudic, V. Kumar, and L. H. Ungar, "Using policy gradient reinforcement learning on autonomous robot controllers," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, Oct. 2003, pp. 406–411.

[40] J. Peters, S. Vijayakumar, and S. Schaal, "Reinforcement learning for humanoid robotics," in *Proc. IEEE-RAS Int. Conf. Humanoid Robots*, Sep. 2003, pp. 1–20.

[41] J. R. Koza, *Genetic Programming II: Automatic Discovery of Reusable Programs*, 1st ed. Cambridge, MA: Bradford Book, May 1994.

[42] D. E. Goldberg, *Genetic Algorithms in Search, Optimization, and Machine Learning*, 1st ed. Reading, MA: Addison-Wesley, 1989.

[43] G. Orlovsky, T. Deliagina, and S. Grillner, *Neuronal Control of Locomotion: From Mollusc to Man*. New York: Oxford Univ. Press, 1999.

[44] D. A. McCrea and I. A. Rybak, "Organization of mammalian locomotor rhythm and pattern generation," *Brain Res. Rev.*, vol. 57, no. 1, pp. 134–146, Jan. 2008.

[45] G. T. Brown, "The intrinsic factors in the act of progression in the mammal," *Proc. Royal Soc. London*, vol. 84, no. 572, pp. 308–319, Dec. 1911.

[46] S. Rossignol, R. Dubuc, and J.-P. Gossard, "Dynamic sensorimotor interactions in locomotion," *Physiol. Rev.*, vol. 86, no. 1, pp. 89–154, 2006.

[47] C. M. A. Pinto and M. Golubitsky, "Central pattern generators for bipedal locomotion," *J. Math. Biol.*, vol. 53, no. 3, pp. 474–489, 2006.

[48] S. Hooper, "Central pattern generators," *Current Biol.*, vol. 10, pp. 176–177, Apr. 2000.

[49] A. D. Kuo, "The relative roles of feedforward and feedback in the control of rhythmic movements," *Motor Control*, vol. 6, no. 2, pp. 129–145, 2002.

[50] O. Kiehn and S. J. Butt, "Physiological, anatomical and genetic identification of CPG neurons in the developing mammalian spinal cord," *Progr. Neurobiol.*, vol. 70, no. 4, pp. 347–361, 2003.

[51] H. Kimura, S. Akiyama, and K. Sakurama, "Realization of dynamic walking and running of the quadruped using neural oscillator," *Auton. Robots*, vol. 7, no. 3, pp. 247–258, 1999.

[52] G. Taga, "Nonlinear dynamics of human locomotion: From real-time adaptation to development," in *Adaptive Motion of Animals and Machines*. Tokyo, Japan: Springer-Verlag, 2006, pp. 189–204.

[53] A. J. Ijspeert, "Central pattern generators for locomotion control in animals and robots: A review," *Neural Netw.*, vol. 21, no. 4, pp. 642–653, 2008.

[54] J. Morimoto, G. Endo, J. Nakanishi, and G. Cheng, "A biologically inspired biped locomotion strategy for humanoid robots: Modulation of sinusoidal patterns by a coupled oscillator model," *IEEE Trans. Robot.*, vol. 24, no. 1, pp. 185–191, Feb. 2008.

[55] K. Matsuoka, "Sustained oscillations generated by mutually inhibiting neurons with adaptation," *Biol. Cybern.*, vol. 52, no. 6, pp. 367–376, Oct. 1985.
[56] D. R. McMillen, G. M. D'Eleuterio, and J. R. Halperin, "Simple central pattern generator model using phasic analog neurons," *Phys. Rev. E, Stat. Phys. Plasmas Fluids Rel. Interdiscip. Topics*, vol. 59, no. 6, pp. 6994–6999, 1999.
[57] J. Nakanishi, J. Morimoto, G. Endo, G. Cheng, S. Schaal, and M. Kawato, "Learning from demonstration and adaptation of biped locomotion," *Robot. Auton. Syst.*, vol. 47, nos. 2–3, pp. 79–91, 2004.
[58] L. Righetti, J. Buchli, and A. J. Ijspeert, "Dynamic Hebbian learning in adaptive frequency oscillators," *Phys. D*, vol. 216, no. 2, pp. 269–281, 2006.
[59] R. Ludovic, B. Jonas, and I. A. Jan, "Adaptive frequency oscillators and applications," *Open Cybern. Syst. J.*, vol. 3, no. 2, pp. 64–69, 2009.
[60] P. Rowat and A. Selverston, "Learning algorithms for oscillatory networks with gap junctions and membrane currents," *Netw., Comput. Neural Syst.*, vol. 2, no. 1, pp. 17–41, 1991.
[61] B. Abernethy, *The Biophysical Foundations of Human Movement*, 2nd ed. Champaign, IL: Human Kinetics, 2005.
[62] A. C. Guyton and J. E. Hall, *Textbook of Medical Physiology*, 11th ed. Amsterdam, The Netherlands: Elsevier, Sep. 2006.
[63] N. Berryman, M. Gayda, A. Nigam, M. Juneau, L. Bherer, and L. Bosquet, "Comparison of the metabolic energy cost of overground and treadmill walking in older adults," *Eur. J. Appl. Physiol.*, vol. 112, pp. 1–8, Aug. 2011.
[64] R. Margaria, *Biomechanics and Energetics of Muscular Exercise*, 1st ed. New York: Oxford Univ. Press, Nov. 1976.