Neural Network Reinforcement Learning for Walking Control of a 3-Link Biped Robot

Ahmad Ghanbari, Yasaman Vaghei, and Sayyed Mohammad Reza Sayyed Noorani

Abstract—**In recent years, researches on adaptive control have focused on bio-inspired learning techniques to deal with real-life applications. Reinforcement Learning (RL) is one of these major techniques, which has been widely used in robot control approaches. The implementation of artificial neural networks in RL algorithms enables more efficient optimal control strategies. This article proposes a field application of neural network reinforcement learning (NNRL) for walking control of an active simulated 3-link biped robot. The adaptive control agent consists of two neural network units, known as actor and critic for learning prediction and learning control tasks. Results of the presented control method reveal its efficiency in stable walking control of the biped robot model as a nonlinear complex dynamic task.**

Index Terms—**Adaptive control, biped robot, neural network reinforcement learning,stable walking.**

I. Introduction

Reinforcement learning (RL) is a widely used machine learning framework in which an agent tries to optimize its behavior during its interaction with its initially unknown environment to solve sequential decision problems that can be modeled as Markov Decision Processes (MDPs) [1]. In RL, the learning agent tries to maximize a scalar evaluation (reward or control cost) and modify the policies through actions. Hence, RL is an efficient framework for solving complex learning control problems.The main components of the RL algorithm are the state signal, action signal and the reward signal, which are demonstrated in Fig. 1.

There are three main elements in RL schemes [2]:

- The agent, which predicts the future reward in order to increase the reward's value with value functions. In many applications with or without having the model of the environment, value function implementation is preferred.
- The policy, one of the major elements in RL, which determines the behavior of the agent over the operation time and it may be stochastic or deterministic.
- The reward function, which demonstrates each particular time action reward value. The reward (total reward) is generally defined as the sum of the rewards over time. If the action leads to the goal, the reward will increase. Conversely, the reward will be decreased if an action distracts the agent.

Immediate or delayed rewards may be employed by assuming a discount factor for the rewards over time.

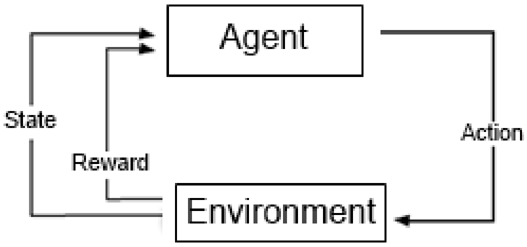

Fig. 1. Reinforcement learning [17].

In recent years, RL has been studied in several different fields such as neural networks (NN), operations research, and control theory [3]-[6]. Moreover, RL can be seen as adaptive optimal control [7]. Studies on human brain reveal that RL is a major human learning mechanism in basal ganglia [8]. The main goal of the researches is to develop NNRL agents, which are able to survive and optimize the system's behavior during their interaction with the environment. According to the importance of function approximation and generalization methods in NNRL, they have been a key research interest recently [9], [10].

A recent study by Tang *et al.* [11] has been done on trajectory tracking of an n-link robot manipulator. The proposed controller consists of two neural networks as the actor and critic with a satisfying tracking error. However, the effect of input nonlinearities, such as dead-zone input and hysteresis has not been considered in this paper. Farkaš *et al.* [12] investigated a two-layer perceptron, actor-critic architecture, and an echo-state neural-network based modules that were trained in different ways on the iCub robot action learning (point, touch, and push). They found that the trained model is able to generalize well in case of action-target combination with randomized initial arm positions and also adapt its behavior to sudden changes during motor execution. In another recent study, Bhasin *et al.* [13] combined robust integral of the sign of the error with the actor-critic architecture to guarantee the asymptotic tracking of the nonlinear system with faster convergence. However, this controller does not ensure optimally.

In addition, some of the recent studies were devoted to central pattern generators (CPGs) RL. Nakamura *et al.* used the CPG-actor-critic method for RL of a biped robot and demonstrated that the proposed method enabled the robot to walk stably and also adapt to the environment [14].

The growing popularity of NNRL algorithms in nonlinear adaptive control led us to implement this method in walking control of a biped robot. Motivated by this belief, first, efforts were made to investigate the robot's dynamics in the next section. Afterwards, the NNRL control design is presented. Then, results and conclusions are demonstrated

Manuscript received April 9, 2014; revised July 25, 2014.

The authors are with Faculty of Mechanical Engineering and Mechatronics Engineering Department, University of Tabriz, Iran (e-mail: a-ghanbari@tabrizu.ac.ir, y.vaghei91@tabrizu.ac.ir, smrs.noorani@tabrizu.ac.ir).

DOI: 10.7763/IJET.2015.V7.835

and in the end, conclusions and future outlook are provided.

II. Biped Robot's Dynamic Modeling

In this section, the dynamical model of a planar 3-link biped robot is introduced. The proposed biped robot consists of a torso, hips, and two equal length legs with no ankles or knees. Also, two torques are applied between the legs and torso. Angular coordinates definition and the masses of the torso, hips and legs of the biped robot disposition are shown in Fig. 2.It is assumed that the positive angles are computed in a clockwise manner with respect to the indicated vertical lines and all links are mass centered and the masses of the links are lumped. The walking cycle takes place on the surface level of sagittal plane. In addition to this, walking phases assumed to be successive, where only one leg (stance leg) touching the walking surface (the swing phase), and the transition from one leg to another taking place in a small length of time. The stance leg is modeled like a pivot and the swing leg is assumed to move into the frontal plane during swing the phase [15], [16]. The swing leg renters the plane of the motion when the angle of the stance leg attains a given desired value.

In this dynamic model, the walking cycle of the biped robot is defined in two parts: the model of the swing phase and another one that describes the impact event of the swing leg with the walking surface, which are discussed below.

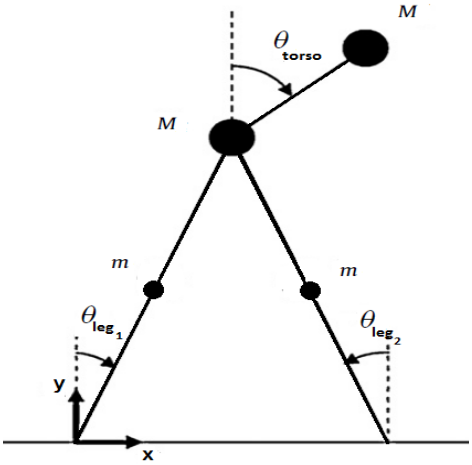

Fig. 2. 3-link biped robot.

A. Dynamic Model

A second order system obtained from the Lagrange method describes the dynamical model of the robot during the swing phase [17]:

$$M(\theta)\ddot{\theta} + C(\theta,\dot{\theta})\dot{\theta} + G(\theta) = Bu \tag{1}$$

where $\theta = [\theta_1,\theta_2,\theta_3]^T$,$u = [u_1,u_2]^T$ are the link angles and input torques to the legs. M, C, G, and B are the mass matrix, nonlinear term, the gravity term and a constant respectively.

We can write the second order differential Eq. (1) into state-spaceform by defining:

$$\dot{x} := \frac{d}{dt}\binom{\theta}{\dot{\theta}} = \begin{bmatrix} \dot{\theta} \\ M^{-1}(\theta)\left[-C(\theta,\dot{\theta})\dot{\theta} - G(\theta) + Bu\right] \end{bmatrix} \tag{2}$$

B. Impact Model

In our simulations, we have modeled the impact between the swing leg and the ground as the contact between two rigid bodies. Obtaining the velocity of the generalized coordinates after the impact of the swing leg with the walking surface in terms of the velocity and position before the impact is the main objective of this model. The proposed impact model for our biped robot is based on the rigid impact model of Ref. [18]. Moreover, we have assumed that the contact of the swing leg with the walking surface produce either no rebound nor slipping of the swing leg, and the stance leg lifting the walking surface without interaction. These assumptions are valid if the following conditions are satisfied.

- The impact occurs over an infinitesimally small period oftime;
- 2. Impulses can represent the external forces during impact.
- 3. Impulsive forces may change the velocities of the generalized coordinates instantaneously, but positions remain continuous.
- 4. The supplied actuators torques is not impulsive. Based on the previous assumptions, the impact model is expressed with Eq. (3)[19]:

$$\begin{bmatrix} M_e & -E^T \\ E & 0 \end{bmatrix}\begin{bmatrix} \dot{\theta}_e^+ \\ F \end{bmatrix} = \begin{bmatrix} M_e\dot{\theta}_e^- \\ 0 \end{bmatrix} \tag{3}$$

where $\theta_e = [\theta_1,\theta_2,\theta_3,x,y]^T$ are the generalized coordinates, M_e is the generalized mass matrix, F is the force matrix, $\dot{\theta}_e^-$ and $\dot{\theta}_e^+$, are the velocities before and after impact. Also, E is defined as following.

$$E = \begin{bmatrix} rcos(\theta_1) & -rcos(\theta_2) & 0 & 1 & 0 \\ -rsin(\theta_1) & rsin(\theta_2) & 0 & 0 & 1 \end{bmatrix} \tag{4}$$

Also, r is the links equal lengths.

III. Neural Network Reinforcement Learning Design

Typically, in machine learning, the environment is formulated as a Markov decision processes (MDPs). According to [19], [20], a MDP consists of a set of states, S, and a set of actions denoted by a. Associated with each action, there is a state transition matrix $P(a)$ and a reward function $r: S \times A \rightarrow R$, where $r(x,a)$ is the expected reward for doing the action a in state x. A policy is a mapping $\pi: S \rightarrow A$ from states to actions. This policy is both stationary and deterministic. The RL's goal is to find policy π, which maximizes the expected value of a specified function, f, of the immediate obtained rewards while following the policy π. This expected value is defined in Eq. (1).

$$J(\pi) = E\{f(r_1,r_2,\ldots)|\pi\} \tag{5}$$

Particularly, actor-only methods deal with a parameterized family of policies which has the benefit of generating a spectrum of continuous actions; however, the implemented optimization methods (policy gradient methods) have the disadvantage of high variance in the estimates of the gradient, which results in slow learning [21].

Critic-only methods use temporal difference learning and have a lower variance in the expected returns estimates. These methods usually work with discrete action space. As a result, this approach is not able to find the true optimum. [22].

The actor–critic algorithms have been shown to be more effective than value function approximation (VFA), and policy search in online learning control tasks with continuous spaces [23]. Actor-critic methods provide the advantages of actor-only and critic-only methods by computing continuous actions without the need for optimization procedures on a value function and supplying the actor with low-variance knowledge of the performance, at the same time. This leads to a faster learning process. The convergence properties of actor-critic methods usually are more satisfying than critic-only methods [24]. Fig. 3 demonstrates the overall scheme of the actor-critic RL.

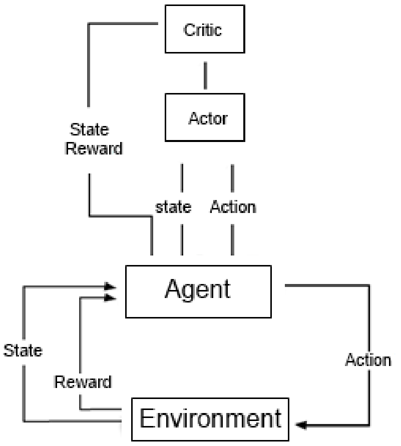

Fig. 3. Overall scheme of actor-critic reinforcement learning.

Here, our main purpose is to minimize the angle between the torso and the vertical line by means of the links' torques, applied to each of the legs' links. The control law is defined in Eq. (6).

$$u(\theta,\dot{\theta})^T = -p(\theta-\theta_d) - k(\dot{\theta}-\dot{\theta}_d) \qquad (6)$$

where the '*d*' notation reveals the desired values, resulted from the neural network training and the control constants are:

$$p = diag[0, 0, p_2, p_3]$$
$$k = diag[0, 0, k_2, k_3]$$

The learning agent alternates the control constants in each step to obtain the maximum reward. The reward is directly related to the angle between the torso and the vertical line.

The learning process continues until the robot learns how to retain its stability by the joint torque control.

IV. Results and Discussion

Motivated by the advantages of the actor-critic methods and their satisfying convergence speed besides their low variance, we have implemented these methods for nonlinear dynamic control of our simulated biped robot. In addition, we have used the feed forward neural networks in the actor and the critic in order to enhance the performance of the system.

In the first step, we have simulated the biped robot's dynamics equations, which represents the environment for the learning unit. Then, we have designed two three-layered perceptron feed forward neural networks as the actor and the critic for the NNRL agent. The network weights of the actor and the critic are variable. These weights converge to a fixed value as the learning process converges to an optimal solution. Fig. 4 illustrates the obtained reward for the iterations over the operation time.

As can be clearly seen, the reward approximately reaches its optimal value in a few iterations. This demonstrates the NNRL strength to deal with the nonlinear dynamics of the simulated biped robot.

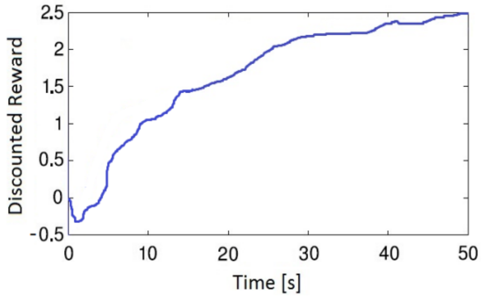

Fig. 4. Discounted reward in learning process.

The results of our simulations also reveal that the trajectory of the proposed biped robot's torso orbits in a limit cycle in phase space as depicted in Fig. 5. This, represents the stability of the robot after learning. However, some perturbations may occur before the convergence of the learning agent to an optimal solution.

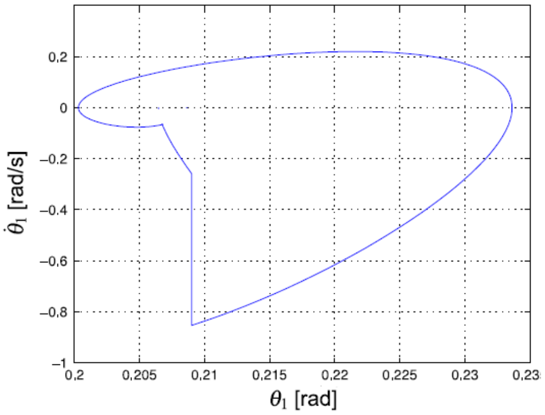

Fig. 5. Torso trajectory limit cycle.

V. Conclusion

As an interdisciplinary area, the combination of reinforcement learning (RL) and neural networks algorithms has changed the face of modern optimal control significantly and became a key research interest in various domains such as robotics, control theory, operational research, and finance. In this paper, a neural network reinforcement learning algorithm is proposed for walking control of a 3-link planar biped robot.

The proposed controller is an actor-critic reinforcement learning unit, in which the actor and the critic are two 3-layered feed forward neural networks with variable network weights. The results reveal the ability of the proposed neural network reinforcement learning method to control the stability of the robot's links after a few numbers of iterations.

REFERENCES

[1] F. L. Lewis and D. Vrabie, "Reinforcement learning and adaptive dynamic programming for feedback control," *Circuits and Systems Magazine*, vol. 9, no. 3, pp. 32-50, 2009.

[2] R. Sutton and A. G. Barto, *Reinforcement Learning, An Introduction,* Cambridge MA, 1998.

[3] L. Busoniu, R. Babuska, B. D. Schutter, and D. Ernst, *Reinforcement Learning and Dynamic Programming Using Function Approximators*, CRC Press, NY, 2010.

[4] J. K. Williams, "Reinforcement learning of optimal controls," *Artificial intelligence Methods in the Environmental Sciences*, pp. 297-327, 2009.

[5] C. Szepesvri, *Algorithms for Reinforcement Learning*, Morgan and Claypool, USA, 2010.

[6] X. Xu, *Reinforcement Learning and Approximate Dynamic Programming*, Science Press, Beijing, 2010.

[7] R. Sutton, A. G. Barto, and R. J. Williams, "Reinforcement learning is direct adaptive optimal control," in *Proc. the American Control Conference*, pp. 2143-2146, 1992.

[8] P. J. Werbos, "Intelligence in the brain: A theory of how it works and how to build it," *Neural Networks*, pp. 200-212, 2009.

[9] D. Ernst, P. Geurts, and L. Wehenkel, "Tree-based batch mode reinforcement learning," *Journal of Machine Learning Research*, vol. 6, pp. 503-556, 2005.

[10] A. G. Barto and T. G. Dietterich, "Reinforcement learning and its relationship to supervised learning," in J. Si, A. Barto, W. Powell, D. Wunsch, Eds., *Handbook of Learning and Approximate Dynamic programming*, Wiley-IEEE Press, New York, 2004.

[11] L. Tang and Y. Liu, "Adaptive neural network control of robot manipulator using reinforcement learning," *Journal of Vibration and Control*, vol. 3, June 2013.

[12] I. Farkaš, T. Malík, and K. Rebrová, "Grounding the meanings in sensor motor behavior using reinforcement learning," *Frontiers in Neurobotics*, vol. 6, no. 1, pp. 1-13, 2012.

[13] S. Bhasin , N. Sharma, P. Patre, and W. Dixon, "Asymptotic tracking by a reinforcement learning-based adaptive critic controller," *Journal of Control Theory Application*, vol. 9, no. 3, pp. 400-409, 2011.

[14] Y. Nakamura, T. Mori, M. Sato, and S. Ishii, "Reinforcement learning for a biped robot based on a CPG-actor-critic method," *Neural Networks*, vol. 20, pp. 723-735, 2007.

[15] T. M. Geer, "Passive dynamic walking," *International Journal of Robotics Research*, vol. 9, no. 2, pp. 62-82, 1990.

[16] S. L. C. Maciel, O. Castillo, and L. T. Aguilar, "Generation of walking periodic motions for a biped robot via genetic algorithms," *Applied Soft Computing*, vol. 11, pp. 5306-5314, 2011

[17] R. Kelly, R. Santibanez, and A. Loria, *Control of Robots Manipulators in Joint Space*, Springer-Verlag London Limited, 2005.

[18] Y. Hurmuzlu and D. Marghitu, "Rigid body collisions of planar kinematic chains with multiple contact points," *International Journal of Robotics Research*, vol. 13, no. 1, pp. 82-92, 1994.

[19] E. R. Westervelt, J. W. Grizzle, C. Chevallereau, J. H. Choi, and B. Morris, "Systematic design of within-stride feedback controllers for walking," in *Feedback Control of Dynamic Bipedal Robot Locomotion*, Taylor and Francis/CRC, pp. 137 – 11891, 2007.

[20] Z. Michalewicz, *Genetic Algorithms + Data Structures = Evolution Programs*, Springer-Verlag New York, Inc., Secaucus, NJ, USA, 1994.

[21] V. R. Konda and J. N. Tsitsiklis, "On Actor-critic algorithms," *SIAM Journal on Control and Optimization*, vol. 42, no. 4, pp. 1143–1166, 2003.

[22] F. S. Melo, S. P. Meyn, and M. I. Ribeiro, "An analysis of reinforcement learning with function approximation," in *Proc. the 25th International Conference on Machine Learning, Helsinki, Finland*, pp. 664–671, 2008.

[23] K. G. Vamvoudakis and F. L. Lewis, "Online actor – critic algorithm to solve the continuous-time infinite horizon optimal control problem," *Automatica*, vol. 46, no. 5, pp. 878 – 888, 2010.

[24] V. R. Konda and J. N. Tsitsiklis, "On Actor-critic algorithms," *SIAM Journal on Control and optimization*, vol. 42, no. 4, pp. 1143–1166, 2003.

Ahmad Ghanbari received his B.Sc. and M.Sc. in mechanical engineering in 1978 and 1981 respectively from University of California Pomona.

He also received his Ph.D. in control engineering in 2007 from University of Tabriz, Iran.

He is currently the head of the School of Engineering Emerging Technologies, University of Tabriz, Iran. Also, he is an associate professor in faculty of mechanical engineering at University of Tabriz, Iran.

Ghanbari is a member of ASME, ISME and also the head of the Iranian Society of Mechatronics.

His research interests include mechatronics, advanced control, biped robots, and nonlinear dynamics.

Yasaman Vaghei received her B.Sc. in mechanical engineering from Ferdowsi University of Mashhad, Iran in 2012. She is now a M.Sc. student of mechatronics engineering at School of Engineering Emerging Technologies, University of Tabriz, Iran.

She has been working in robotics field since 2004 and achieved prizes in various robotics competitions.

Miss Vaghei is a member of ASME, ISME, and Iranian Society of Mechatronics.

Her research interests include adaptive control, learning algorithms, and robotics.

Sayyed Mohammad Reza Sayyed Noorani received his M.Sc. and Ph.D in mechanical engineering in 2009 and 2013 respectively from Faculty of Mechanical Engineering, University of Tabriz, Iran.

He is currently with the Mechatronics Engineering Department, School of Engineering Emerging Technologies, and University of Tabriz, Iran.

Noorani is currently a member of Iranian Experts, and Iranian Society of Mechatronics.

His research interests include arm dynamics, biped locomotion, and mobile robots.