# Survey on Reinforcement Learning and Adaptive Dynamic Programming for Feedback Control

*Shaun Pritchard \* Email:* [*spritchard2021@fau.edu*](mailto:spritchard2021@fau.edu) *\* Florida Atlantic University \* CAP6629 \* N. Zhen \*Apr 1, 2022*

**Introduction**

The following paper outlines the adaptive programming paradigms used in reinforcement learning. Researchers claim that living organisms learn by acting on their environment of deserving reward stimuli and adjusting their actions in response to improve the rewards. Using action-based or reinforcement learning, natural systems can exhibit optimal behavior. This paper outlines practical computational methods for implementing reinforcement learning using mathematical formulations. This paper outlines Adaptive Dynamic Programming(ADP )methods that reveal insights into the designs of controllers for man-made systems that both learn and exhibit optimal Behavior. We investigate specific methods in reinforcement learning to define optimal behavior in comparison to the behavior of natural intelligence systems. To begin with, the research compares how living organisms interact with their environments and use those interactions to improve their actions to survive and grow. It explains how Darwin demonstrated that species modify their actions in response to interactions with the environment over long timescales, leading to natural selection and the survival of the fittest. Adam Smith stated that the relative balance and wealth of nations are determined by the actions of corporate entities on a global scale. Through the use of simple reinforcement and punishment stimuli, Ivan Pavlov modified the behavior patterns of dogs by inducing conditional reflexes. Reinforcement learning (RL) is defined as the modification of actions based on interactions with the environment. Rewards or punishment are related to actions in RL. In the sense that the agent understands reward versus punishment, it implies goal-directed behavior. Research in RL is based on several dynamic systems, including Feedback Control Theory, which is the study of ways to develop reliable control systems for human-engineered systems. Among these are control systems for aircraft, ships, race cars, robot systems, industrial processes, building temperature, climate control systems, and many others. There is a group of reinforcement learning techniques known as Approximate or Adaptive Dynamic Programming (ADP) (also known as Neurodynamic Programming) concerning human-based engineered systems which is the basis of this survey. The following are the main ideas and algorithms of reinforcement learning: Approximate Dynamic Programming. The following research covers multiple algorithms used in building ADP systems and advanced approximate systems. The algorithms discussed in this research are as follows; Dynamical Systems and Optimal Feedback Control, Dynamical Systems, Optimal Control for Discrete-Time Systems, Goal Directed Optimal Performance, Bellman’s Optimality Principle and Dynamic Programming, Policy Iteration (PI) Algorithm, Policy Iteration, Value Iteration, and Fixed Point Equations, Value Iteration (VI) Algorithm, Optimal Control Solution for the DT LQR, Policy Iteration and Value Iteration for the DT LQR, Generalized Policy Algorithm for LQR, ADP-Temporal Difference (TD) and Value Function Approximation (VFA), Reinforcement Learning, ADP, and Adaptive Control, ADP- On-Line Reinforcement Learning Optimal Control, Q Learning and Dual Learning, Fixed Point Equation for Q Function, Q Function for Reinforcement Learning Using Policy or Value Iteration, Q Function for LQR Case, Dual or Gradient Learning, Reinforcement Learning and ADP for Continuous-Time Systems [1].

For brevity I will only elaborate on core algorithms in this survey many of these algorithms are preliminary. The algorithms of discussion in the sections of this paper are as follows; ADP- On-Line Reinforcement Learning Optimal Control, Reinforcement Learning and Adaptive Control, Q Learning and Dual Learning, Reinforcement Learning and ADP for Continuous-Time Systems, followed by a conclusion [1].

**Value Function approach**

Approximately Dynamic Programming (ADP), or Neuro-Dynamic Programming (NDP), determines the optimal control solution online forward in time using measurements along the system trajectory. Ding methods for solving the dynamic programming problem forward in time in real-time and for approximating the value function. The optimal control solution using dynamic programming is a backwards-in-time procedure. There are two key

ingredients: temporal difference (TD) error and value function approximation (VFA). As a fixed point equation, the Bellman equation is a consistency equation that the value must satisfy if it is consistent with the current control policy. For reinforcement learning algorithms, fixed-point equations can generally be used, provided they are formulated properly. The Basic Concepts are transformed into forward-in-time online solutions using the TD Error (Time Difference Error), based on the Bellman equation, which defines a time-varying residual equation error as follows. This yields the best approximation to the value corresponding to using the current policy



The TD equation can be solved online as a nonlinear Lyapunov equation without knowing the dynamics of the system, simply using the data measured along its trajectory. To further simplify the TD equation, use the Kronecker product is used as follows. 

**ADP- On-Line Reinforcement Learning Optimal Control**

**This** substitutes into the Bellman TD equation to obtain The equation is a fixed point equation. It is a consistency equation that is satisfied at each time k for the valuecorresponding to the current policy. As an iterative procedure for solving the TD equation may be used, including Policy Iteration and Value Iteration. Using an online policy iteration algorithm to solve the optimal control problem by measuring data along the trajectory of a system, an online reinforcement learning algorithm is provided to solve the optimal control problem. The On-Line Value Iteration Algorithm was implemented into a policy, in order to determine the least squares solution to converge the regression vector and determine an improved policy. Essentially, reinforcement learning algorithms work on the basis that the actor in a control action in the system environment relays the system output while the policy update, improvement, and reward response are persisted to the system environment.

**Reinforcement Learning and Adaptive Control**

It is possible to perform adaptive control either directly, wherein one estimates the controller parameters directly, or indirectly, by first estimating the parameters of the system model, then computing the controller. The researchers show that reinforcement learning is an indirect adaptive controller wherein the parameters of the Values are estimated, the control is computed then optimal control is directly computed in terms of the learned parameters as a direct adaptive control schema.

**Q Learning approach**

By learning the value function by reinforcement learning methods, the optimal value and the optimal control are stored as functions of the state vector apart from the value function, in which system dynamics must be known. In the value function learning approach or HDP, one requires knowledge of the system dynamics. One method for avoiding knowing any of the system dynamics is to take partial derivatives on the control input that does not go through the system by using the Q function approximation . In ADP several algorithms are derived to evaluate the Q function approximation or fixed techniques that learn the Q-function.

**Fixed Point Equation for Q Function**

Fixed Point Equation for Q Function is the bellman equation for Q which utilizes an approximation structure for the Q-function as follows. 

**Q Function for LQR Case**

****

**Q Function for Reinforcement Learning Using Policy or Value Iteration**

Motivated by the LQR example, for nonlinear systems one assumes a parametric approximator or NN.

**Q Learning Policy Iteration Algorithm**



**Dual or Gradient Learning**

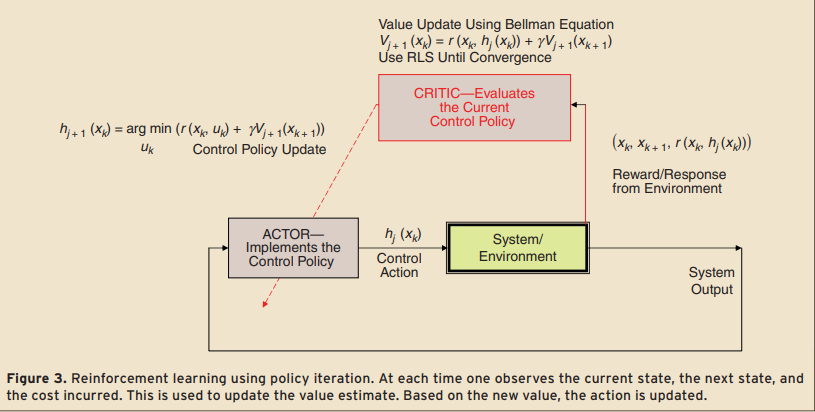


HDP reinforcement Learning methods based on the value can be determined using the Bellman or fixed point equation as follows.

****

**Reinforcement Learning and ADP for Continuous-Time Systems**

For continuous time systems or time series problems, ADP has found solutions. The development of reinforcement learning for continuous-time systems has lagged behind that for discrete-time systems. Where the continuous-time nonlinear dynamical system through a CT Policy Iteration (PI) Algorithm where either algorithm requires knowledge about the internal system dynamics function and the reinforcement measured on each time interval. The reinforcement learning time interval T need not be the same at each iteration as shown in Figure 1 from the research paper [1].



**Conclusion**

ADP Reinforcement Learning controllers and the human brain are very similar from a computational perspective. Based on samples from the environment, the actor critic structure learns the parameters of a function that describes the actor's performance. The researcher suggest that at the end of each performance evaluation episode, the critic passes this information to the actor structure that will use it to adjust for improved performance At all times, the actor must perform continuous time control for the system in which optimal performance of the way in which the actor/critic structure works. furthermore, while searching for optimal control policies points out the existence of two-time scales for the mechanisms involved. It was demonstrated that even with limited information about the states of the system measured from the sensors, and extracted from the system only at specific time values, the Critic is still capable of evaluating the unlimited time performance of a system associated with a given control policy defined in terms of the Actor parameters. According to this paper, the critic learns the cost function associated with a certain control behavior by computing the temporal difference error signals [2]. There is an argument that the temporal difference error between the received and expected rewards is physically encoded in the dopamine signal generated by the basal ganglia in mammals.

As a whole, ADP is a better method than Direct Utility Estimation as it runs trials to learn the model of the environment by estimating the utility of a state as a sum of rewards for being in that state and the discounted reward of being in the next state. As a result of the presented algorithms in this study, the Time Difference Error can also be considered in order to achieve higher performance with ADP

# **References**

|  |  |
| --- | --- |
| [1] | F. L. Lewis, "IEEE Circuits and Systems Magazine Reinforcement Learning and Adaptive Dynamic Programming for Feedback Control," *IEEE Circuits and Systems Magazine ,* vol. 9, p. 9, 2009. |
| [2] | G. H. F. L. L. T. P. C. M. Said G. Khan, "Reinforcement learning and optimal adaptive control: An overview and implementation examples," *Elsevier,* vol. 36, no. 1, pp. 42-59, 2012. |