**No Grid**

Generic structure of a reinforcement learning problem. The optimization methods to solve the reinforcement learning problem are mainly categorized into value function and policy search methods.

The standard reinforcement learning theory states that an agent is able to obtain a policy, which maps every state  to an action , where  is the state space (possible states of the agent in the environment) and  is the finite action space. The inner dynamics of the agent are represented by the transition probability model  at time . The policy can be stochastic , with a probability associated with each possible action, or deterministic . In each time step, the policy determines the action to be chosen and the reward  is observed from the environment. The goal of the agent is to maximize the accumulated discounted reward  from a state at time  to time  ( for infinite horizon problems) [29]. The discount factor  is defined to allocate different weights for the future rewards.

For a specific policy , the value function  in ([17](https://www.hindawi.com/journals/js/2017/3296874/#EEq19)) is a representation of the expectation of the accumulated discounted reward  for each state  (assuming a deterministic policy ):

An equivalent of the value function is represented by the action-value function  in ([18](https://www.hindawi.com/journals/js/2017/3296874/#EEq20)) for every action-state pair :

The optimal policy  shall be the one that maximizes the value function (or equivalently the action-value function), as in the following equation:

A general problem in real robotic applications is that the state and action spaces are often continuous spaces. A continuous state and/or action space can make the optimization problem intractable, due to the overwhelming set of different states and/or actions. As a general framework for representation, reinforcement learning methods are enhanced through deep learning to aid the design for feature representation, which is known as deep reinforcement learning. Reinforcement learning and optimal control aim at finding the optimal policy  by means of several methods. The optimal solution can be searched in this original primal problem, or the dual formulation  can be the optimization objective. In this review, deep reinforcement learning methods are divided into two main categories: value function and policy search methods.

2.3.1. Value Function Methods

These methods seek to find optimal , from which the optimal policy  in ([20](https://www.hindawi.com/journals/js/2017/3296874/#EEq22)) is directly derived. -learning approaches are based on the optimization of the action-value function , based on the Bellman Optimality Equation [29] for  (see ([21](https://www.hindawi.com/journals/js/2017/3296874/#EEq22))):

Deep -Network (DQN) [30, 31] method estimates the action-value function (see ([22](https://www.hindawi.com/journals/js/2017/3296874/#EEq24))) by means of a CNN model with a set of weights  as :

The CNN can be trained by minimizing a sequence of loss functions  which are optimized in each iteration  as shown in the following equation:

The state  of the DQN algorithm is the raw image and it has been widely tested with Atari games [31]. DQN is not designed for continuous tasks; thus this method may find difficulties approaching some robotics problems previously solved by continuous control. Continuous -learning with Normalized Advantage Functions (NAF) overcomes this issue by the use of a neural network that separately outputs a value function  and an advantage term , which is parametrized as a quadratic function of nonlinear features [32]. These two functions compose final , given by the following equation:with  being the state,  being the action, and , , and  being the sets of weights of , , and  functions, respectively. This representation allows simplifying more standard actor-critic style algorithms, while preserving the benefits of nonlinear value function approximation [32]. NAF is valid for continuous control tasks and takes advantage of trained models to approximate the standard model-free value function.

2.3.2. Policy Search Methods

Policy-based reinforcement learning methods aim towards directly searching for the optimal policy , which provides a feasible framework for continuous control. Deep Deterministic Policy Gradient (DDPG) [33] is based on the actor-critic paradigm [29], with two neural networks to approximate a greedy deterministic policy (actor) and  function (critic). The actor network is updated by applying the chain rule to the expected return from the start distribution  with respect to the actor parameters (see ([25](https://www.hindawi.com/journals/js/2017/3296874/#EEq27))):

DDPG method learns with an average factor of 20 times fewer experience steps than DQN [33]. Both DDPG and DQN require large samples datasets, since they are model-free algorithms. Regarding -based Guided Policy Search (-based GPS) [34] method, it learns to map from the tuple raw visual information and joint states directly to joint torques. Compared to the previous works, it managed to perform high-dimensional control, even from imperfect sensor data. DNN-based GPS has been widely applied to robotic control, from manipulation to navigation tasks [35, 36].