Reinforcement Learning for Combinatorial Optimization

R

Di Wang
https://orcid.org/0000-0002-7992-7743
University of Illinois Chicago, USA

INTRODUCTION

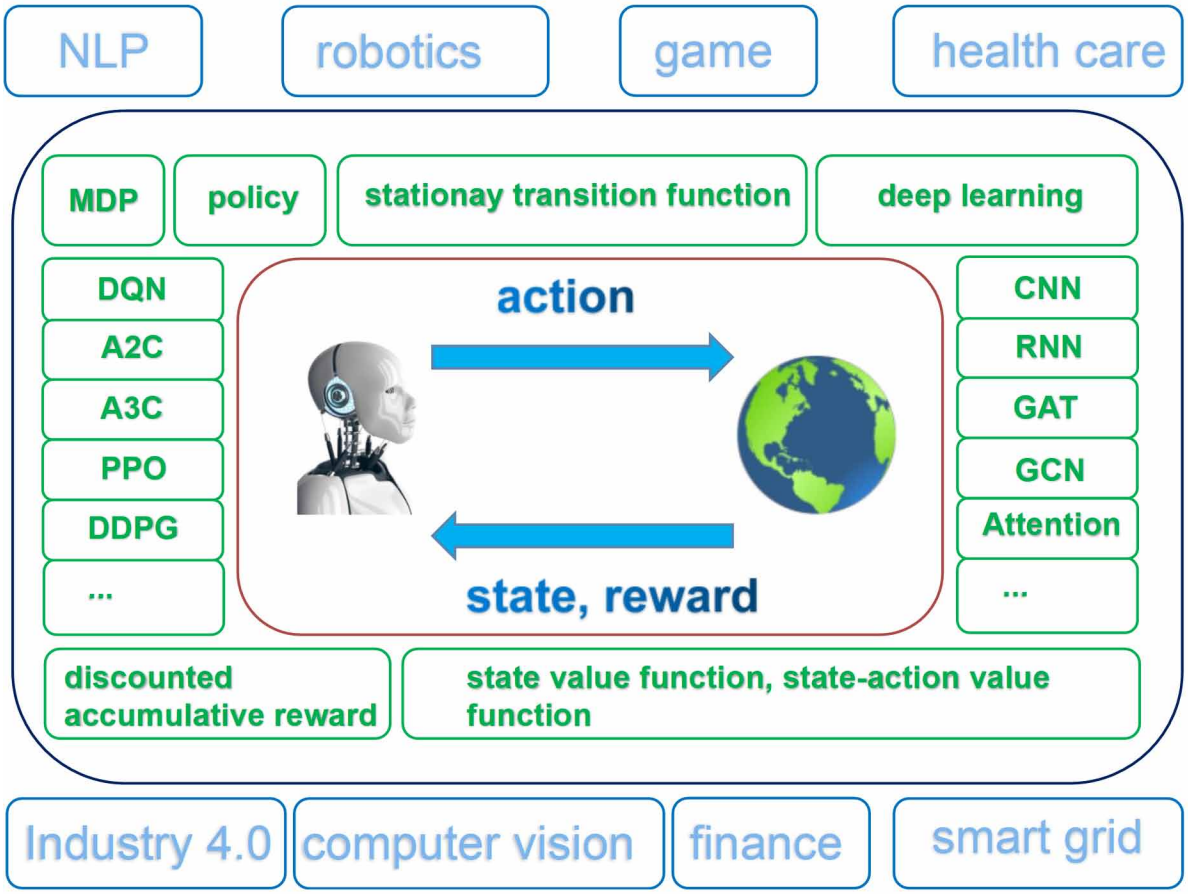
Reinforcement learning (RL) is a subfield of machine learning (ML) that tells what actions the intelligent agent should take to maximize the received cumulative rewards (Arulkumaran, 2017) from environments. Building on Markov Decision Process (MDP), a discrete-time stochastic control process, reinforcement learning is suitable for temporal decision-making problems with long-term and short-term reward trade-offs. Researchers introduce deep learning into reinforcement learning to fully utilize the advanced GPU computation ability and potent representation property of neural networks. Moreover, exploiting deep neural networks (DNN) as function approximators instead of using limited-capacity value memory tables make it possible for DRL to solve large-scale problems. The effectiveness of DRL has been proved with applications across multiple industries, including robot control (Yuan, 2022), computer games (Peng, 2022), natural language processing (Du, 2022), healthcare (Oh, 2022), finance (Asgari, 2022) and others.

As shown in Figure 1, many terminologies and concepts are included in deep reinforcement learning. DRL's four essential constituent parts are states, actions, rewards, and stationary transition functions. Specifically, the state represents the observable environment, which refers to the agent's surroundings or the target optimization problem. Action denotes the behaviors of the agent or decision variables of optimization problems, while policy refers to actions at the sequence of time steps. The stationary transaction function records probabilities of transitioning from one state to another. The reward is the evaluation of the action providing DRL the directions of the training process, because of which, DRL belongs to semi-supervised learning algorithms. Unlike supervised learning algorithms, the ground truth is not provided. Theoretically, DRL updates parameters of neural networks greedily towards the gradient of maximizing the expectation of rewards. In detail, the policy evaluation step and the policy estimation step are the essential steps in the training process. Figure 2 illustrates an example of the cooperation between these two steps. The policy evaluation step iteratively estimates state function $v_{\pi,}$ where π denotes the adopted policy. An accurate evaluation can provide accurate instruction to the policy improvement step. The policy improvement step greedily generates better policies, providing more data for training the policy evaluation step.

Furthermore, the discounted accumulative rewards can be approximated by two main function approximators, the state value function or state-action value function (also known as Q value). The main goal of DRL is to maximize the expectation of the cumulative rewards (also known as return). Based on the initial actions' availability, the return's expectation can be modeled as state value function or state-action value function (Q value). Q value measures the quality of an action at a given state followed by a particular policy. State value measures the expectation of future discounted rewards starting from the given state and followed by a specific policy.

DOI: 10.4018/978-1-7998-9220-5.ch170

Figure 1. Main concepts and terminologies in deep reinforcement learning. The deep reinforcement learning algorithm contains the state, actions, reward, and stationary transition function. State value function and state-action value function are two approximations of the discounted accumulative reward. Besides, DQN, A2C, A3C, and PPO are state-of-art DRL algorithms, while CNN, RNN, GAT, and GCN are state-of-art neural network architectures.



Value-based and policy-based methods are the two main categories of the model-free DRL methods. DQN (Osband, 2016), as the value-based method, is suitable for discrete-action space problems. Actor-critic (Konda, 2000), as the policy-based method, is suitable for both discrete-action and continuous-action space problems. Policy-based methods directly model the agent's policy as a parametric function. Value-based methods first compute the value-action function, then the agent's policy corresponds to picking an action that maximizes the Q value. For complex CO problems, the action space involves discrete actions and continuous actions. For instance, in the energy aware VRP, the remaining task points are discrete actions, while the charged energy is continuous actions. Thus, this chapter focuses on policy-based actor-critic DRL methods.

For developing proper neural networks, researchers adopt different state-of-art neural network architectures like convolutional neural network (CNN), recurrent neural network (RNN), graphical convolutional neural network (GCN) and others. However, different architectures have different characteristics. For instance, RNN is able to extract sequential features from the inputting sequence, while CNN is famous for tackling pictures data. GCN can keep the translation invariance and extract nodes/edges features