

# LibForce: C++向け強化学習ライブラリの開発

白神 雪人<sup>†</sup> 中本 幸一<sup>‡</sup>

<sup>†</sup> 兵庫県立大学応用情報科学研究科

<sup>‡</sup> 兵庫県立大学情報科学研究科

## 1 はじめに

強化学習はエージェントと呼ばれる学習機構が環境から学習しながら目的とする機能を具備するように訓練していく機械学習である [1]. 組み込み機器では、処理速度が速いことやシステム開発に適していることから C 言語が主に開発言語として用いられてきており、C++言語での開発も広まりつつある。強化学習の主な開発言語は Python であるが、C++言語で強化学習を実装することができれば、組み込み機器での展開も期待できる。C++には、Python のように強化学習のライブラリや参考資料が充実していないため、強化学習を実装する難易度は高い。そこで、強化学習ライブラリを作成し提供することで、C++での強化学習の実装を支援することができる。本稿では、強化学習ライブラリ LibForce について述べる。

## 2 強化学習の概要

強化学習では、環境とエージェントから構成される (図 1)。エージェントは環境から状態と報酬という情報を取得し、学習を行い、エージェントの内の方策に基づいて行動を決定し環境に出力する。環境はエージェントから行動を受けて、ある変化を起こし、それを状態の変化という形でエージェントに返す。報酬は行動の効果の良さを表す。本ライブラリはエージェントを構成するプログラム開発と環境の利用を支援することを目的とする。

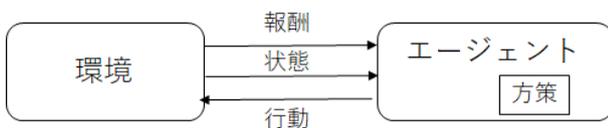


図 1: 強化学習の枠組み

**LibForce: Reinforcement learning library in C++**

Yukito Shiraga<sup>†</sup> and Yukikazu Nakamoto<sup>‡</sup>

<sup>†</sup> Graduate School of Applied Informatics, University of Hyogo

<sup>‡</sup> Graduate School of Information Science, University of Hyogo

## 3 設計方針

LibForce ライブラリは、強化学習プログラムの作成を通じて得た経験をもとに、以下の設計方針をあげた。

- D1:** 環境の構成情報をもとにエージェントの構成情報を設定するのが面倒である。この設定を隠蔽し、初心者簡単にプログラミングできるようにする。既存の強化学習ライブラリはこのような機能はなかった。
- D2:** 環境やエージェントが簡単に交換可能にできる。
- D3:** 部品となるクラスを提供し、新たなアルゴリズム開発時に再利用できる。

## 4 ライブラリの概要

本ライブラリのアーキテクチャを図 2 に示す。本ライブラリは、以下の 3 つのクラス群によって構成される。

- Agent クラス群: エージェントを実現する。エージェント実装のための `Agentinterface` 仮想クラスとそれを実装した Agent クラス (現在は Deep Q-Network(DQN)) から構成される。
- Env クラス群: 環境をラッピングする Env 仮想クラスとそれを実装したクラス (現在は OpenAi Gym 内の環境) から構成される。
- Runner: Env クラスと Agent クラスを用いて強化学習の学習ループを実行するクラス群
- Module クラス群: Agent クラスを実装するために利用可能な部品から構成される。

方針 D1 を実現するために、エージェントの実装に必要な情報を環境からもってくるようにした。具体的には環境をラップする Env クラス初期化時に、エージェントが必要なアクションや状態の構成情報を生成し、これらを Agent クラスの初期化の返り値を Agent に渡すようにした。テンプレートを使って Agent クラスと Env クラスの初期化を Runner クラスで行えるようにした。方針 D2 を実現するために、Agent と Env を多

様なエージェントや環境で利用できるよう、Runner クラスに対するテンプレートパラメタとした。方針 D3 を実現するために Module クラス群として、以下のようなエージェントプログラム作成用の部品クラスを用意した。

- Transition : エージェントと環境間の 1 回のインタラクションを管理する
- Memory : 一連の Transition, すなわち, 学習の履歴を記憶する機能を有する。
- Actor: 強化学習のアクションを決める方策を有する

## 5 ライブラリ仕様

本ライブラリで提供する主要なクラスの仕様を示す。class AgentInterface Agent クラスの実装に必要な仮想関数を定義しているクラスである。Agent クラスでは、環境からの情報を受け取りアクションの選択や方策の更新を行う。以下の仮想関数を宣言する。

```
void reset(torch::Tensor state) エージェントの初期化を行う。
```

```
torch::Tensor action(torch::Tensor state) 環境の state を受け取り、環境に指示する行動を返す。行動は方策にしたがって選択される。
```

```
void update(torch::Tensor reward, torch::Tensor done) 環境にアクションを指定し返ってきた報酬と成否を受け取り、方策の更新を行う。
```

class Envinterface Env クラスの実装に必要な仮想関数を定義しているクラスである。Env クラスはこのクラスを継承して仮想関数を実装する。Env クラス

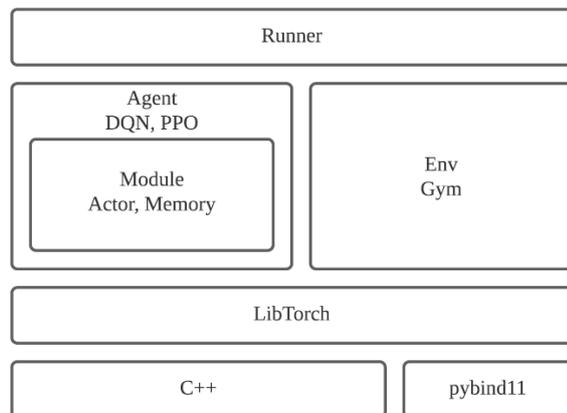


図 2: LibForce の構成

は強化学習の環境であり行動を受け取るごとに状態が変化し、それによる報酬や成否を返す。

```
std::tuple<torch::Tensor, torch::Tensor, torch::Tensor> reset() 環境の初期化を行い、初期化後の状態, 報酬, 成否のタプルを返す。
```

```
std::tuple<torch::Tensor, torch::Tensor, torch::Tensor> step(torch::Tensor action) エージェントが選択した行動を受け取り、環境で行動を実行し。現在の状態, 報酬, 成否のタプルを返す。
```

template<Agent, Env>class Runner Agent クラスと Env クラスを保有し、強化学習の学習ループを実行するクラスである。Agent テンプレートは、Agent クラスの型である。

```
template<Agent, Env> Runner() Runner オブジェクトを生成する。テンプレートで指定されたエージェントクラスと環境クラスのオブジェクトを初期化し保有する。
```

```
template<Agent, Env> void run(size_t num_episodes) 学習が終了するまでのエピソード数を指定し、強化学習の学習を実行する。
```

## 6 関連研究

強化学習ライブラリにはいくつかあるが、本ライブラリに類似の機能をもつもの RLzoo[2] である。相違点は、本ライブラリの方がパラメタ化をより行っており、エージェントの置換がより簡単であること、C++向けであることである。

## 7 おわりに

本ライブラリは現在、エージェントとして DQN を実装しているのみである。さらに、他の強化学習アルゴリズムを実装し、有効性を検証していく必要がある。

## 参考文献

- [1] R. Sutton and A. Barto. *Reinforcement Learning: An Introduction*. Bradford Books, 2nd edition, 2018.
- [2] Z. Ding, et al., Efficient Reinforcement Learning Development with RLzoo. In *Proc. 29th ACM Int'l Conf. on Multimedia*, pp. 3759–3762, 2021.