**ИЕРАРХИЧЕСКИЙ АЛГОРИТМ ОБУЧЕНИЯ С ПОДКРЕПЛЕНИЕМ C2PRL**

Рубашкин И.М.

*Президентский физико-математический лицей № 239*

*Rubiliam2003@gmail.com*

Обучение с подкреплением — это направление в машинном обучении, связанное с тем, как агенту(ам) следует выполнять действия в заданной среде, чтобы максимизировать суммарно некоторую награду. Среда может представлять из себя практически все что угодно: от реального мира до компьютерных игр, использовавшихся в работе. Обычно нейронные сети используются в качестве агентов. Все алгоритмы обучения с подкреплением можно поделить на две большие группы: одноуровневые и иерархические. Одноуровневые алгоритмы состоят только из одной главной части, определяющей поведение агента, например: DQN, DDPG, Actor Critic… Следующей ступенью развития стали иерархические алгоритмы: они состоят из нескольких частей, чье взаимодействие друг с другом определяет поведение агента.

Алгоритм является иерархическим. Он состоит из двух уровней. Это отражено в его названии: C2PRL – centralized 2 part reinforcement learning. Первый называется «мозг», следующий называется «исполнитель». Мозг, как понятно из названия, занимается обработкой высокоуровневых концептов. Исполнитель получает некоторые указания от мозга, дальше ему дается определенный срок на то, чтобы их выполнить. Рассмотрим процесс взаимодействия агента со средой.

В стационарном режиме, то есть без обучения, среда передает агенту информацию о текущем состоянии s, после чего она передается мозгу, который, в свою очередь, пропускает его через нейронную сеть и генерирует так называемую цель g. Это происходит раз в k итераций. Дальше текущее состояние среды и задание передаются исполнителю. Исполнителю дается k итераций, чтобы достигнуть такого состояние среды, что: $\left‖s\_{t^{'}}−g\_{t}\right‖\leq ε$ То есть состояние, близкое к тому, которое запросил мозг. Если это условие выполнено быстрее чем за k итераций, то генерация новой цели запускается досрочно. Если же агент находится в режиме обучения, то все становится немного сложнее: помимо мозга и исполнителя в игру вступает и experience replay. Это специальная структура, хранящая данные о прошедших транзакциях и выдающая их для обучения. Эмпирическими методами было выяснено, что наиболее стабилен процесс обучения в том случае, когда мозг и исполнитель обучаются отдельно. Первым обучается исполнитель, при этом вместо мозга задачи генерируются рандомно. Исполнитель должен научиться достигать произвольного состояния — это упростит процесс обучение мозга. Когда обучается мозг, исполнитель, который должен быть заранее обучен, не обучается, он работает в стационарном режиме. Иногда, если размерность входных состояний слишком велика и будет невозможно обучить мозг генерировать сложные многомерные структуры, к среде добавляется специальная сеть, чья задача – ужимать входные данные, сохраняя максимальное количество информации. Она называется компрессор.

Мозг обучается с помощью алгоритма ppo, это так называемый онлайн-алгоритм, то есть он работает только с данными, полученными текущей версией агента, после обновления параметров сети нужно собирать новые данные. Исполнитель обучается либо с помощью ddpg, либо с DQN. Это оффлайн-алгоритмы; для ускорения обучения и увеличения стабильности данные, полученные агентом, дополнительно обрабатываются с помощью hindsight experience replay. Это нужно, так как в начале обучения, когда исполнитель совсем не знает среду и действует практически наугад, очень маловероятно, что ему удастся выполнить задание мозга и получить положительное подкрепление. Поэтому оно создается отдельно, и исполнитель на каждой итерации обучения гарантированно получает часть положительно подкрепленных данных. Компрессор обучается сжимать данные с помощью архитектуры Variational Autoencoder, при допущении, что данные, необходимые агенту, имеют значительную роль во входных данных.

Алгоритм тестировался на средах Atari 2600 игр с помощью библиотеки gym. Эти видеоигры требуют быстрой реакции, стратегического мышления на несколько шагов вперед и являются относительно непростыми для игроков-людей. Состояние среды – изображения экрана консоли размером 210x160x3. Так как на одном статическом изображении нельзя проследить изменение скоростей и ускорений некоторых объектов, важных для агента, было решено использовать в качестве состояния 4 последние кадра игры. Сначала происходит ранняя обработка: все изображения переводятся из rgb в черно-белые, сжимаются до размера 84x84, после чего состояние среды имеет размерность 84x84x3. Этот тензор сжимается с помощью компрессора до 32 мерного вектора, после чего агент может с ними работать. Для всех сред использовалась одна архитектура сети (см. приложение). Алгоритм проверялся на следующих средах: AirRaid, Boxing, Breakout, Carnival, DemonAttack, IceHockey, Pong, Seaquest, SpaceInvaders. Алгоритм показал внушающие результаты: в 7 из 9 сред, в которых он тестировался, его результаты сравнимы с алгоритмом DQN (основываясь на данных из [1]) и превосходят игроков-людей.

1. Volodymyr Mnih, Koray Kavukcuoglu, David Silver, AndreiA. Rusu, Joel Veness, Marc G. Bellemare, Alex Graves, MartinRiedmiller, Andreas K. Fidjeland, Georg Ostrovski, Stig Petersen, Charles Beattie1, Amir Sadik, Ioannis Antonoglou, HelenKing, Dharshan Kumaran, Daan Wierstra, Shane Legg, Demis Hassabis. *Nature,* 2015, **518**, 529-533.

2. Lillicrap, Timothy P, Hunt, Jonathan J, Pritzel, Alexander, Heess, Nicolas, Erez, Tom, Tassa, Yuval, Silver, David, and Wierstra, Daan*. arXiv preprint arXiv:1509.02971*, 2015.

3. Andrew Levy, George Konidaris, Robert Platt, and Kate Saenko. *arXiv preprint arXiv:1712.00948*, 2019.

4. Nachum, O., Gu, S., Lee, H., and Levine, S *arXiv preprint* *arXiv:1810.01257* , 2019.

5. Ofir Nachum, Shane Gu, Honglak Lee, and Sergey Levine. *arXiv preprint arXiv:1805.08296*, 2018.

6. Marcin Andrychowicz, Filip Wolski, Alex Ray, Jonas Schneider, Rachel Fong, Peter Welinder, Bob McGrew, Josh Tobin, Pieter Abbeel, Wojciech Zaremba. *arXiv preprint* *arXiv:1707.01495,* 2017.

7. T. Schaul, D. Horgan, K. Gregor, D. Silver. *ICML,* 2015.

8. Kingma, D. P. and Welling, M. (2019). *arXiv preprint arXiv:1906.02691*, 2019.

9. Fang M, Zhou C, Shi B, et al. *International Conference on Learning Representations*, 2019.

10. John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, Oleg Klimov. *arXiv preprint arXiv:1707.06347*, 2017