

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ
ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ
ВЫСШЕГО ОБРАЗОВАНИЯ
"МОСКОВСКИЙ ГОСУДАРСТВЕННЫЙ
УНИВЕРСИТЕТ
имени М.В.ЛОМОНОСОВА"
ФИЗИЧЕСКИЙ ФАКУЛЬТЕТ
КАФЕДРА ФИЗИКО-МАТЕМАТИЧЕСКИХ МЕТОДОВ
УПРАВЛЕНИЯ

Выпускная дипломная работа

Обучение с подкреплением
в задаче поиска пути в лабиринте

Выполнил студент IV курса:
Завгородний Игорь Викторович

Научный руководитель:
Галяев А.А.

Москва
2019

1. Введение

Данная работа посвящена применению метода обучения с подкреплением (Reinforcement Learning) в задаче поиска оптимального пути в трёхмерном лабиринте. Построенные математические и программные модели применимы для описания движения агентов в различных физических системах. Например, описание движения беспилотного летательного аппарата (БПЛА), выполняющего задачи в различных слоях атмосферы, описание движения автономного подводного судна, выполняющего исследования на разной глубине, и так далее.

В результате работы был создан и протестирован алгоритм, позволяющий осуществлять оптимальное управление агентом в трёхмерном лабиринте, имитирующим атмосферу. Метод обучения с подкреплением показал эффективность при обучении агента на заданных лабиринтах, где данные не меняются с течением времени, что, безусловно, отличается от реальных процессов.

Оглавление

1.	Введение	1
2.	Теоретическое введение	3
3.	Постановка и формализация задачи	6
4.	Описание используемых методов	7
5.	Программная реализация	9
6.	Вывод	9
7.	Список используемой литературы	10

2. Теоретическое введение

Современные задачи науки и техники требуют применения современных методов, позволяющих быстро и корректно обрабатывать большие объёмы данных, ежесекундно поступающих с многочисленных датчиков. Более того, с увеличением объёма задач, стоящих перед кибернетическими агентами, усложняется их поведение. Традиционные методы программирования исчерпывают себя, делая решение современных задач неэффективным по затрачиваемому времени и используемой памяти.

Данные проблемы призван преодолеть метод машинного обучения (Machine Learning), фундаментальные основы которого были заложены еще в 1940-1950-х годах прошлого века. Однако бурное развитие подобных методов началось лишь в 1990-х годах вместе с ростом вычислительных мощностей компьютеров. Достоинством данного метода является отсутствие необходимости создавать детерминированные алгоритмы, полностью покрывающие необходимые сценарии поведения агентов. Машинное обучение позволяет создать агентов нового типа, способных обучаться и строить оптимальные алгоритмы при минимальном воздействии человека.

Существует две основных концепции машинного обучения: обучение с учителем, в котором агент обучается производить определённые действия на основании предварительно подготовленных выборок, и обучение без учителя, в котором агент самостоятельно формирует стратегию поведения, опираясь на изменения, производимые его действиями. Обучение с подкреплением принадлежит ко второму типу машинного обучения. Агент перебирает все варианты действий и из всех возможных действий выбирает те, которые принесут ему наибольшее итоговое вознаграждение. Перечисленные концепции называются "методом проб и ошибок" и "отсроченным поощрением" они лежат в основе обучения с подкреплением.

В данной работе для решения задачи поиска пути в лабиринте применяется метод обучения с подкреплением. Как было сказано ранее, одной из особенностей метода является то, что обучение агента

происходит благодаря взаимодействию с окружающей средой. Лабиринт - это и есть среда, предназначенная для экспериментального исследования, в которой движется управляемый агент. Задача поиска пути в лабиринте является одной из ключевых задач в робототехнике, решение которой позволяет создавать системы управления движением автономных роботов (дронов).

Метод обучения с подкреплением в общем виде можно представить в качестве марковского процесса принятия решений:

$$(S, A, P_a(s, s'), R_a(s, s')), \text{ где:}$$

1. S - множество возможных состояний среды,
2. A - множество возможных действий агента над средой,
3. $P_a(s, s') = P(s_{t+1} = s' | s_t = s, a_t = a)$ - вероятность, что состояние s под действием a во время t перейдёт в состояние s' ко времени $t + 1$,
4. $R_a(s, s') = R(s_{t+1} = s' | s_t = s, a_t = a)$ - вознаграждение, получаемое после перехода в состояние s' из состояния s с вероятностью $P_a(s, s')$.

Поведение агента описывается следующей цепочкой действий:

состояние \rightarrow действие \rightarrow поощрение \rightarrow состояние \rightarrow
 \rightarrow действие \rightarrow поощрение $\rightarrow \dots$

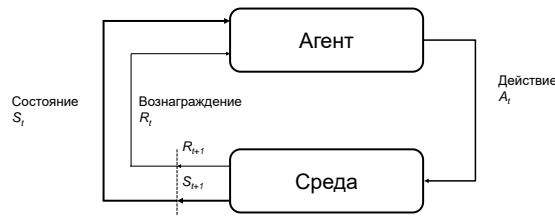


Рис. 1: SARSA-модель

В англоязычной литературе данный процесс носит название «SARSA» («State-Action-Reward-State-Action-...»).

Вводится некоторая политика (англ. *policy*):

$$\pi : S \times A \rightarrow [0, 1]$$

$\pi(a | s) = P(a_t = a | s_t = s)$ - вероятность действия a в состоянии s .

Цель агента - выбрать такую оптимальную политику π , обозначающую вероятность выбора действия a в состоянии s , чтобы при следовании ей сумма вознаграждений, получаемых от среды, была максимальна. Ожидаемая награда в момент времени t определяется как:

$$R_t = E[r_t + \gamma r_{t+1} + \gamma^2 r_{t+2} + \dots] = E \left[\sum_{k=0}^{\infty} \gamma^k r_{t+k} \right],$$

где $E[\cdot]$ - математическое ожидание, $\gamma \in (0, 1)$ - коэффициент дисконтирования (англ. *discount rate*).

Долгосрочная стратегия агента в общем случае не подразумевает преследование максимальной выгоды на каждом промежуточном шаге. Непосредственный выбор стратегии может осуществляться множеством способов. Введем функцию $Q(s, a)$, которая парам состояние-действие ставит в соответствие число. Данное число называется ценностью состояния-действия. Также на каждом временном шаге t агент получает вознаграждение r_t :

$$Q^\pi(s, a) = E_\pi[R_t | s_t = s, a_t = a] = E_\pi \left[\sum_{k=0}^{\infty} \gamma^k r_{t+k} | s_t = s, a_t = a \right],$$

где индекс π означает выбор действий в соответствии с некоторой политикой (*policy*).

Эта функция характеризует ожидаемую награду, получаемую агентом стартуя из состояния $s, s \in S$ совершая действие $a, a \in A$, и в дальнейшем действуя в соответствии с определенной политикой π .

Отсюда мы можем получить рекурсивную формулу для оценки данной функции:

$$Q_{i+1}^\pi(s, a) = E_\pi \left[r_t + \gamma \sum_{k=0}^{\infty} \gamma^k r_{t+k+1} \mid s_t = s, a_t = a \right] = \\ E_\pi \left[r_t + \gamma Q_i^\pi(s_{t+1} = s', a_{t+1} = a') \mid s_t = s, a_t = a \right]$$

Однако целью агента является - нахождение оптимальной политики π , на которой достигается максимальная ожидаемая награда. Таким образом, мы должны найти такую π^* , которая в результате нам дает максимальное значение action-value функции $Q^*(s, a)$ среди всех существующих политик. Формула для оценки оптимального значения action-value функции определяется следующим образом:

$$Q_{i+1}(s, a) = E[r_t + \gamma \max_{a'} Q_i(s', a') \mid s, a]$$

При $i \rightarrow \infty$ следует, что $Q_i(s, a) \rightarrow Q^*(s, a)$. Данный процесс называется *алгоритмом итерации значений* (англ. *value iteration algorithm*).

3. Постановка и формализация задачи

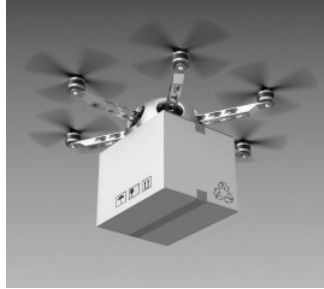


Рис. 2: Беспилотный аппарат

Рассмотрим движение беспилотного аппарата (агента) в атмосфере (испытательной среде, лабиринте). Задачей агента является сбор грузов в различных точках пространства и их доставка до точек выгрузки с наименьшими затратами топлива.

Для описания системы используются следующие величины:

1. $\vec{r}(x, y, z)$ - координата беспилотного аппарата в трёхмерном пространстве: $x \in [0; l_x]$, $y \in [0; l_y]$, $z \in [0; l_z]$;
2. $\rho(\vec{r})$ - плотность атмосферы в точке пространства: $\rho \in [\rho_{min}; \rho_{max}]$;

3. F - статус наличия груза: $F \in \{0; 1\}$.

Множество действий, доступных агентов состоит из восьми элементов:

1. Движение на Юг (south);
2. Движение на Север (north);
3. Движение на Восток (east);
4. Движение на Запад (west);
5. Движение на Вверх (east);
6. Движение на Вниз (west);
7. Сбор груза (pickup);
8. Сброс груза (dropoff);

Таким образом, состояние среды описывается четырьмя непрерывными вещественными числами, а множество действий состоит из двух значений.

Беспилотный аппарат (агент) движется в трёхмерном пространстве(среда), каждому слою атмосферы соответствует одна координата по оси Z, с каждым слоем уменьшается общий для слоя мультипликатор. Кроме того, каждому слою атмосферы соответствует определённое распределение плотности воздуха по осям X и Y, реализованное случайной расстановкой очков вознаграждения.

4. Описание используемых методов

Перед тем как описывать способы определения ценностей для пар состояний-действий, стоит указать используемые стратегии выбора действий. В данной задаче есть два принципиально разных класса состояний, следовательно, и стратегии для них тоже должны быть разные. В случае, когда производится выбор карты для озвучивания, карта выбирается следующим образом. В руке находятся наименования с наибольшим количеством карт, далее из них случайно выбирается одно

наименование. Во втором случае используется -жадная стратегия, она заключается в жадном выборе действия (действие, которое максимизирует $Q(s, a)$ с вероятностью ϵ , в остальных случаях действие выбирается случайно. Все используемые в работе методы построения оценки функции ценности пар состояний-действий основаны на методе временных различий (TD — Temporal-Difference). В TD-методах процесс обучения основывается на опыте взаимодействия агента со средой без использования модели среды. Расчетные оценки состояний (в случае задачи управления состояний-действий) в TD-методах обновляются, основываясь на других полученных оценках, т.е. они самонастраиваются [2]. Классический TD-метод используют для построения оценок ценности состояния среды. Опишем его, перед тем как перейти к случаю управления. В данной работе будут использоваться идеи многошагового TD-метода, так же известного как метод TD, и одношагового метода, или метода TD(0), который является частным случаем многошагового. В многошаговом методе имеется переменная памяти $e(s)$, соответствующая каждому состоянию. Она называется следом приемлемости [2]. На каждом временном шаге следы приемлемости для всех состояний, кроме текущего, убывают с коэффициентом γ , а след приемлемости для посещаемого на данном шаге состояния увеличивается на параметр затухания следа, — коэффициент приведения. След приемлемости все время регистрирует, посещение каких состояний имело место недавно, где смысл понятия "недавно" определяется с помощью коэффициента. Процесс оценки состояний проходит следующим образом. Во время обучения при переходе из состояний s_t в состояние s_{t+1} вычисляется величина $\delta_t = V(s_{t+1}) - V(s_t)$ где $V(s_t)$ — функция ценности состояния, аналогичная функции ценности пар состояний-действий $Q(s, a)$. Далее для всех состояний производится корректировка их ценности с использованием следов приемлемости $e(s)$ где α — коэффициент обучения. Соответственно, в случае одношагового метода никаких следов приемлемости нет, т.к. $\alpha = 0$, поэтому на каждом шаге производится только корректировка ценности состояния s_t , что можно записать в виде
$$V(s_t) \leftarrow V(s_t) + \alpha (V(s_{t+1}) - V(s_t))$$
 Одним из наиболее важных достижений в обучении с подкреплением стало развитие управления по TD-методу с разделенной оценкой ценности стратегий, известного как Q-обучение. В данной работе используется простейший одношаговый алгоритм корректировки ценностей пар состояние- действие, который основывается на одношаговом методе TD, (4) с штрихами здесь

состояния и действия $st + 1at + 1$, без штрихов $stiat$. В этом случае искомая функция ценности действия Q непосредственно аппроксимирует оптимальную функцию ценности действий, независимо от применяющейся стратегии. [3]. Альтернативой методам Q -обучения является метод SARSA (State-Action- Reward-State-Action), который основывается на модели обобщенной итерации по стратегиям с использованием TD-метода в оценочной или предсказательной части. В данной работе используется TD-метод управления с интегрированной оценкой ценности стратегий. Последовательность действий в методе SARSA() базируется на двух шагах. Первый шаг заключается в изучении функции ценности действий. Для этого необходимо оценить функцию $Q(s, a)$ для состояния s и всех действий a . Далее выбирается действие a и производится переход в следующее состояние. Второй шаг повторяет первый, только в конце шага вместо перехода производится корректировка ценностей всех пар состояний-действий [4]. По аналогии с методом TD() находится величина, а далее для всех пар состояний-действий производится корректировка оценок и корректировка всех следов приемлемости.

5. Программная реализация

Основой для решения послужила библиотека Gym от OpenAI. Библиотека содержала, рассмотренную мной задачу в упрощённом виде: обучение с подкреплением использовалось для оптимизации обработки заказов и движения такси в двумерном лабиринте.

Несмотря на кажущуюся схожесть с задачей управления беспилотным аппаратом, требовалась серьёзная доработка существующего решения:

- Требовалось обобщить задачу на случай движения в трёх измерениях;
- Требовалось изменить постановку задачи так, чтобы добавить физический и прикладной смыслы.

Обе задачи были выполнены.

6. Результаты работы

7. Вывод

8. Список используемой литературы

- [1] Комаров А. Ю., Метод обучения с подкреплением для архитектуры вероятностных автоматов.
- [2] Князятв С.А., Малинецкий Г.Г., Решение задачи распознавания блефа в игре «верю – не верю» с помощью алгоритмов обучения с подкреплением // Препринты ИПМ им. М.В.Келдыша. 2018. No 170. 21 с.
- [3] André Barreto, Will Dabney, Rémi Munos, Jonathan J. Hunt, Tom Schaul, Hado van Hasselt, David Silver, Successor Features for Transfer in Reinforcement Learning
- [4] André Barreto, Will Dabney, Rémi Munos, Jonathan J. Hunt, Tom Schaul, Hado van Hasselt, David Silver, Successor Features for Transfer in Reinforcement Learning
- [5] Romain Larocbe, Merwan Barlier, Transfer Reinforcement Learning with Shared Dynamics
- [6] Саттон Р., Барто Э. Обучение с подкреплением – Бином. Лаборатория знаний, 2012. – 400 с.