

# Intrinsic Motivation and Automatic Curricula via Asymmetric Self-Play

Климкин Андрей

Высшая Школа Экономики

15 января, 2018

- 1 Вспомним про RL
- 2 Предложенный метод
- 3 Несколько примеров экспериментов
- 4 Выводы и дальнейшее развитие

1 Вспомним про RL

2 Предложенный метод

3 Несколько примеров экспериментов

4 Выводы и дальнейшее развитие

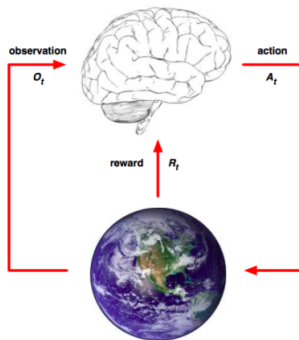
# Напоминание стандартной задачи обучения с подкреплением (ОсП)

## Агент

- совершает действия
- воздействует на среду
- получает награду
- наблюдает изменения среды
- корректирует свое поведение

## с целью

- максимизировать ожидаемую награду



# Markov Decision Processes (MDP)

- $MDP = \langle S, A, T, R, \gamma, D \rangle$  формально описывают среду для ОсП, где:
  - $S \in \mathbb{R}^n$  множество состояний.
  - $A \in \mathbb{R}^m$  множество действий.
  - $T(s' | s, a)$  вероятность перехода из состояния  $s$  в состояние  $s'$  при действии  $a$ .
  - $R$  – функция награды, где  $R : S \times A \times S \rightarrow \mathbb{R}$ .
  - $\gamma$  – параметр дисконтирования.
  - $D$  – распределение на начальное состояние  $s_0$ .
- С каждой политикой (стратегией)  $\pi$  связан функционал награды  $J^\pi$ , оценивающий ее:

$$J^\pi = E\left\{\sum_{t=0}^H \gamma_t^t r(t) \mid x_0 \sim D, \pi\right\}$$

где  $H$  – протяжение взаимодействия агента со средой.

# Policy-Based подход к решению задачи RL

- В Policy-Based подходах параметризуем саму политику  $\pi$ , а не функцию ценности, то есть  $\pi_\theta(s, a) = \mathbb{P}[a \mid s, \theta]$ .
- Функционал для оптимизации в случае скалярной награды выглядит следующим образом:

$$J(\theta) = \mathbb{E}_{\tau \sim p(\cdot | \theta)}[R(\tau) \mid x_0 \sim D, \theta]$$

- $J(\theta)$  оптимизируется путем градиентного подъема и применения Policy Gradient Theorem:

$$\begin{aligned} \nabla_\theta J(\pi_\theta) &= \int_S \rho^\pi(s) \int_A \nabla_\theta \pi_\theta(a \mid s) Q^\pi(s, a) da ds = \\ &= \mathbb{E}_{s \sim \rho^\pi, a \sim \pi_\theta}[\nabla_\theta \log \pi_\theta(a \mid s) Q^\pi(s, a)] \end{aligned}$$

- 1 Вспомним про RL
- 2 Предложенный метод
- 3 Несколько примеров экспериментов
- 4 Выводы и дальнейшее развитие

# Мотивация предложенного метода

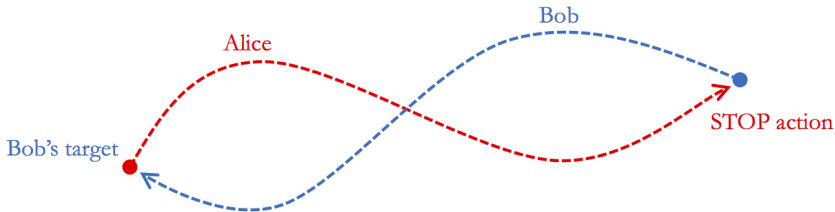
- Для того, чтобы обучить агента с помощью PG требуется **очень много эпизодов**.
- Предположим, что агенту ничего не стоит взаимодействовать с самой динамикой среды (или какой-то ее частью), но при этом трудозатратно получать сигнал (настоящую награду) за совершаемые им действия.
- Можем ли мы тогда предобучить агента *unsupervised*, чтобы потом в среде с настоящей наградой агент обучился быстрее?



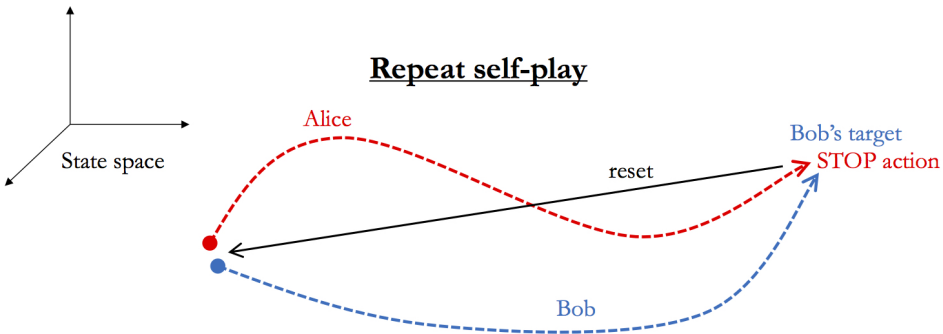


- Введем двух агентов, которые **будут играть сами с собой** (противостоять друг другу).
  - **Алиса** будет предлагать задачи, **выполняя их же сама**.
  - **Боб** будет стараться их решить после Алисы.
- Рассматриваем только среды одного из двух типов:
  - **Repeatable** - в любой момент агента можно вернуть в начальное состояние и продолжить взаимодействие (repeat self-play).
  - (Nearly) **Reverseable** - из любого состояния можно вернуться в начальное (reverse self-play).

## Reverse self-play



## Repeat self-play



# Выбор награды для Алисы и Боба

Алиса:  $R_A = \gamma \max(0, t_B - t_A)$

Боб:  $R_B = -\gamma t_B$

- Награда Алисы направлена на то, чтобы давать простые задачи Бобу, которые он не может выполнить
- Бобу необходимо как можно быстрее справиться с поставленной задачей
- Структура наград позволяет агентам автоматически строить план обучения (Automatic Curricula)

# Как же все это обучать?

- Policy Gradient:

$$\pi_A = f_{\theta_A}(s, s^0)$$

$$\pi_B = f_{\theta_B}(s, s^*)$$

$s_0$  – стартовое состояние среды

$s^*$  – несет в себе информацию о поставленной задаче Алисой Бобу

- Градиентный шаг:

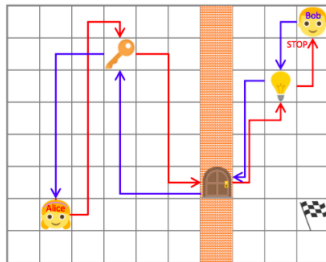
$$\Delta\theta = \sum_{t=1}^T \left[ \frac{\partial \log f(a_t | s_t, \theta)}{\partial \theta} \left( \sum_{i=t}^T r_i - b(s_t, \theta) \right) - \lambda \frac{\partial}{\partial \theta} \left( \sum_{i=t}^T r_i - b(s_t, \theta) \right)^2 \right]$$

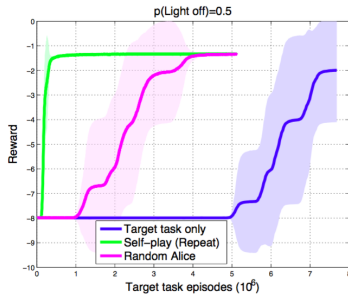
$b(s_t, \theta)$  - это baseline, например, value function.

- 1 Вспомним про RL
- 2 Предложенный метод
- 3 Несколько примеров экспериментов**
- 4 Выводы и дальнейшее развитие

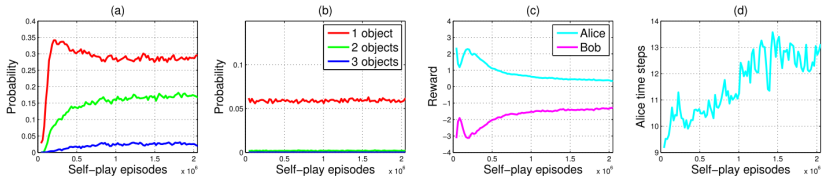
# The Mazebase

- Gridworld с двумя комнатами
- В случайных клетках располагаются ключевые объекты - ключ, лампочка, флаг
- Необходимо поднять ключ, чтобы открыть дверь и перейти в другую комнату
- Если свет выключен, то агент видит только лампочку
- Исходная задача - попасть в клетку с флагом
- В исходной задаче агент и флаг в начале эпизода находятся в разных комнатах



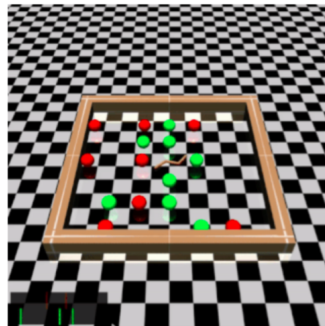


**Рис.:** Пример кривых обучения для Mazebase без предобучения, предобучения с рандомной и «Self-Play» обучаемой Алисой.



**Рис.:** Формирования Automatic Curricula: (a) показывает вероятность взаимодействия обучаемой Алисы с 1, 2, 3 ключевыми предметами, (b) тоже самое, но для необучаемой (рандомной) Алисы, (c) - кривые наград Алисы и Боба, (d) - время взаимодействия Алисы со средой.

- Управляем червячком с двумя подвижными конечностями
- В исходной постановке задачи червь получает награду  $+1$  за каждое собранное зеленое яблоко и  $-1$  за красную бомбу
- В предобучении убираем яблоки и бомбы, учим только перемещению
- Затем обучаем на сбор зеленых яблок и игнорированию бомб





# Сравнение с другими методами.

- Предобученный агент начинает получать значительную награду гораздо раньше
- В итоге сходятся приблизительно к тому же значению, что и SimHash агент

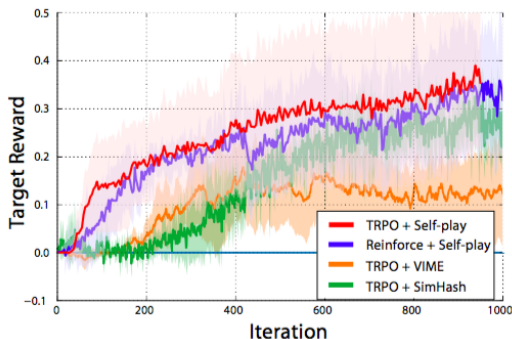
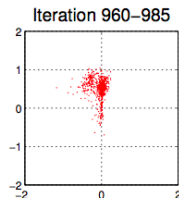
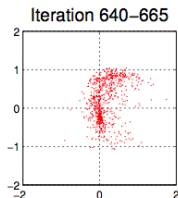
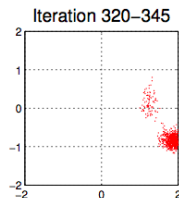
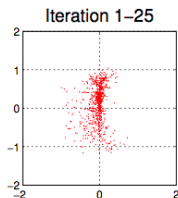


Рис.: Кривые обучения для state-of-the-art exploration методов - VIME, SIMHASH и описанного self-play метода с двумя различными Policy Gradient алгоритмами - TRPO, REINFORCE.

# Проблема локальной сходимости.

- Распределение конечных состояний  $s^*$  Алисы





- 1 Вспомним про RL
- 2 Предложенный метод
- 3 Несколько примеров экспериментов
- 4 Выводы и дальнейшее развитие

О чем работа:

- Простой unsupervised метод предобучения, позволяет агенту на некоторых средах обучаться быстрее
- Идея построения automatic curriculum - от простых задач к сложным
- Способ замены exploration (условно), так как exploration в идеале должен быть связан с настоящим ревордом

Проблема и идея для дальнейшего развития:

- **Проблема:** с какого-то момента Алиса начинает строить одинаковые таски для Боба
- **Предложенная идея:** можно сделать несколько Алис и чередовать их для Боба

-  Sukhbaatar, S., Kostrikov, I., Szlam, A., and Fergus, R. (2017). Intrinsic motivation and automatic curricula via asymmetric self-play.  
arXiv:1703.05407.
-  David Silver presentation about Policy Gradient methods

# Дополнительные графики для MazeBase

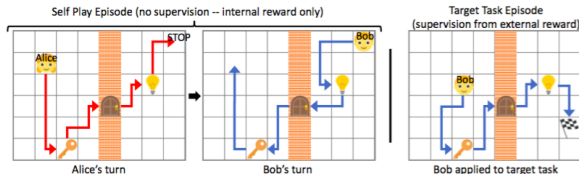


Рис.: Более детальный пример взаимодействия Алисы и Боба во время обучения, и пример взаимодействия Боба со средой в исходной среде (после пред обучения).

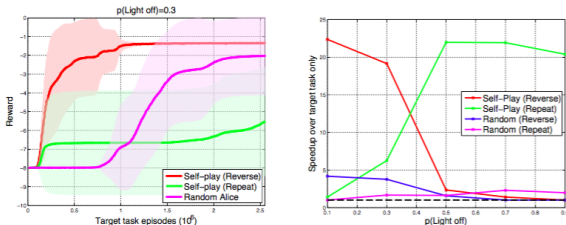


Рис.: Зависимость обучения от априорного распределения на вероятность включенной лампы.

# Дополнительные графики для SwimmerGather

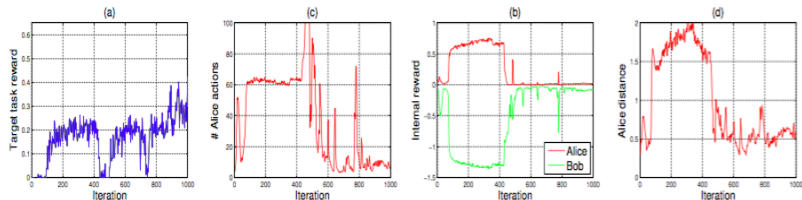


Рис.: Детали обучения агента для среды SwimmerGather: а) кривая награды на исходном taskе, б) кривые наград Алисы и Боба во время самообучения, в) количество действий, совершенных Алисой, в зависимости от номера итерации, д) расстояние, пройденное Алисой, в зависимости от номера итерации