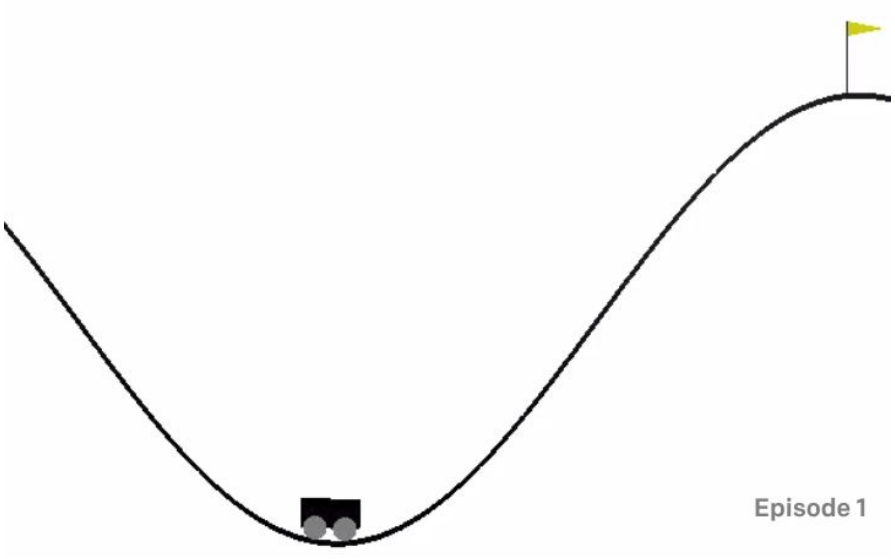
Robust Adversarial Reinforcement Learning

Презентация Зойкина Александра

Проблема с RL в реальном мире

Симуляция, в которой мы обучаем, недостаточно отражает реальные условия

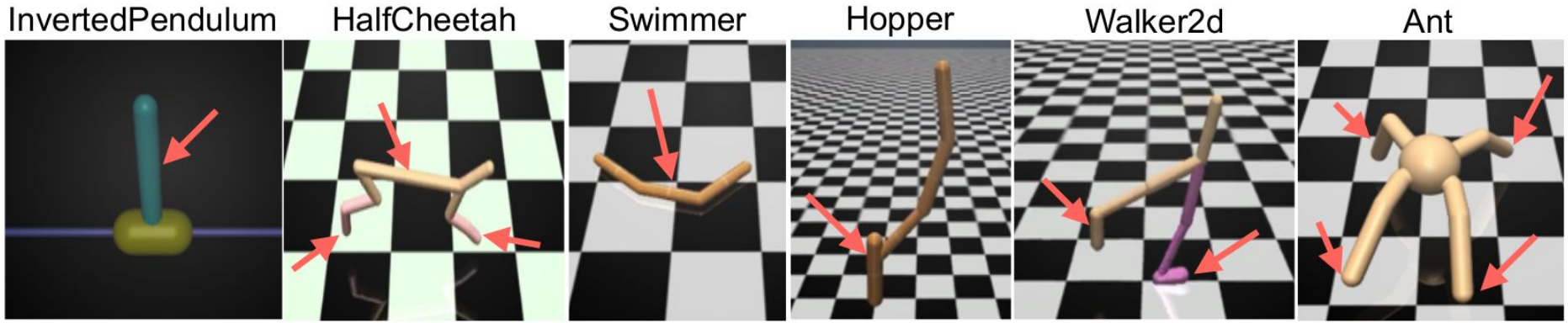


Решения

- Обучение в реальном мире - мало данных для тренировки - переобучение
- Обучение в симуляции - физика отличается, параметров намного меньше

Как промоделировать неизвестные при обучении параметры?

Введем действие дестабилизирующих сил на агента. Агент не знает характеристики этих сил.



Второй агент

- Обучим второго RL-агента(врага) для применения дестабилизирующих сил во время обучения
- Врага награждаем если первый агент(протагонист) ошибается
- В расширенном варианте враг знает о мире больше протагониста и может менять физику среды, например, менять массу протагониста

RL-задача

- $(\mathcal{S}, \mathcal{A}_1, \mathcal{A}_2, \mathcal{P}, r, \gamma, s_0)$
- S - состояния, A - действия агентов,
- $\mathcal{P} : \mathcal{S} \times \mathcal{A}_1 \times \mathcal{A}_2 \times \mathcal{S} \to \mathbb{R}$ вероятность переходов
- $r : \mathcal{S} \times \mathcal{A}_1 \times \mathcal{A}_2 \to \mathbb{R}$ награда для обоих агентов
- γ дисконт-фактор
- s_0 - начальное состояние

функция награды протагонисту

$$R^1 = E_{s_0 \sim \rho, a^1 \sim \mu(s), a^2 \sim \nu(s)}[\sum_{t=0}^{T-1} r^1(s, a^1, a^2)].$$

где μ, ν - политики 1 и 2 агентов соответственно, а a^1 и a^2 - предпринятые действия

Функция награды врага

$$R^2 \equiv R^2(\mu, \nu) = -R^1(\mu, \nu).$$

Игра с нулевой суммой

- Равновесие при

$$R^{1*} = \min_{\nu} \max_{\mu} R^1(\mu, \nu) = \max_{\mu} \min_{\nu} R^1(\mu, \nu)$$

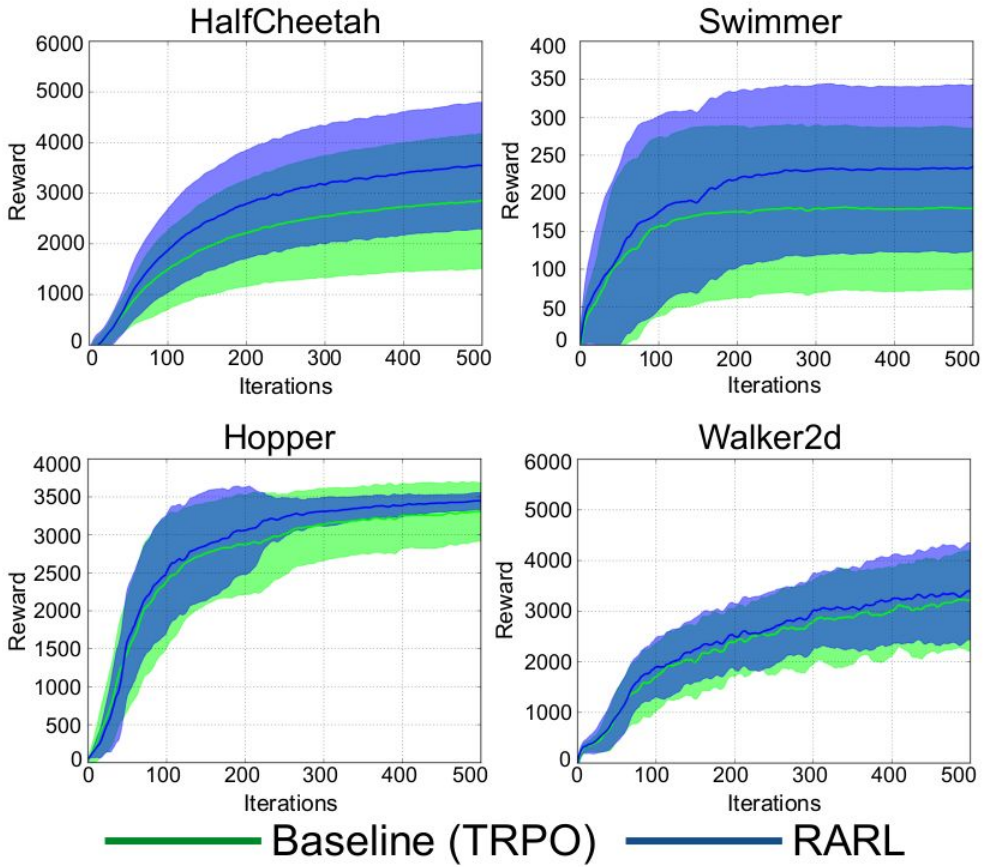
Итоговый алгоритм

- Стандартный процесс оптимизации политики
- В каждой итерации оптимизируем сначала политику протагониста, затем врага

Эксперименты

Зеленая линия - средняя награда бейзлайна. Синяя - средняя награда RARL. Площадь - разброс.

RARL везде лучше, но на Hopper еще и раброс меньше



Эксперименты

Среднее и дисперсия для лучших политик на задачах

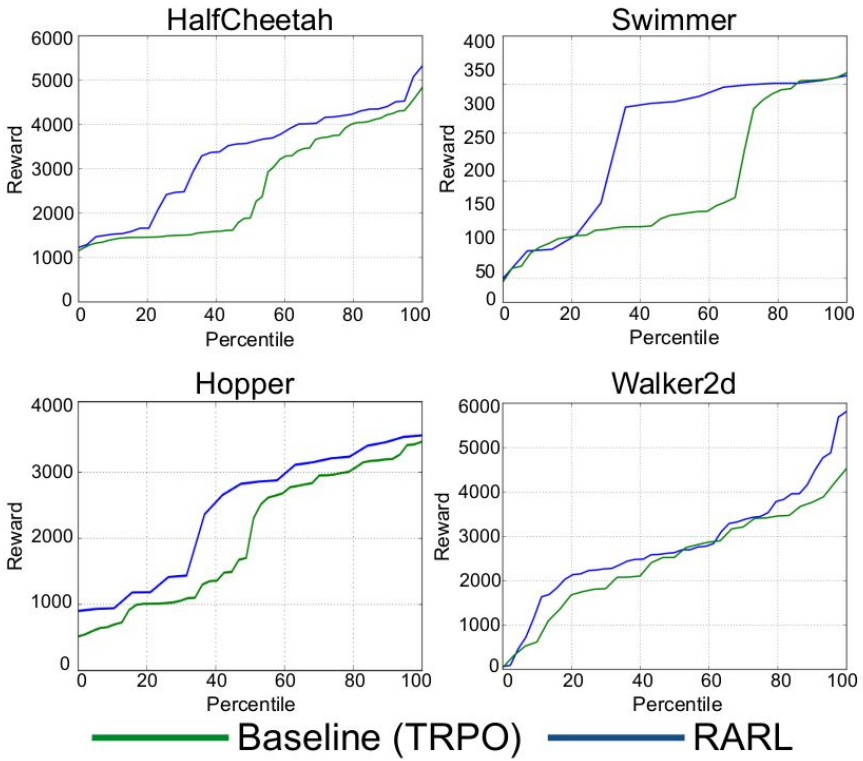
	InvertedPendulum	HalfCheetah	Swimmer	Hopper	Walker2d	Ant
Baseline	1000 ± 0.0	5093 ± 44	358 ± 2.4	3614 ± 2.16	5418 ± 87	5299 ± 91
RARL	1000 ± 0.0	5444 ± 97	354 ± 1.5	3590 ± 7.4	5854 ± 159	5482 ± 28

Эксперименты

Исследуем теперь не лучшую, а среднюю политику из 50 обученных.

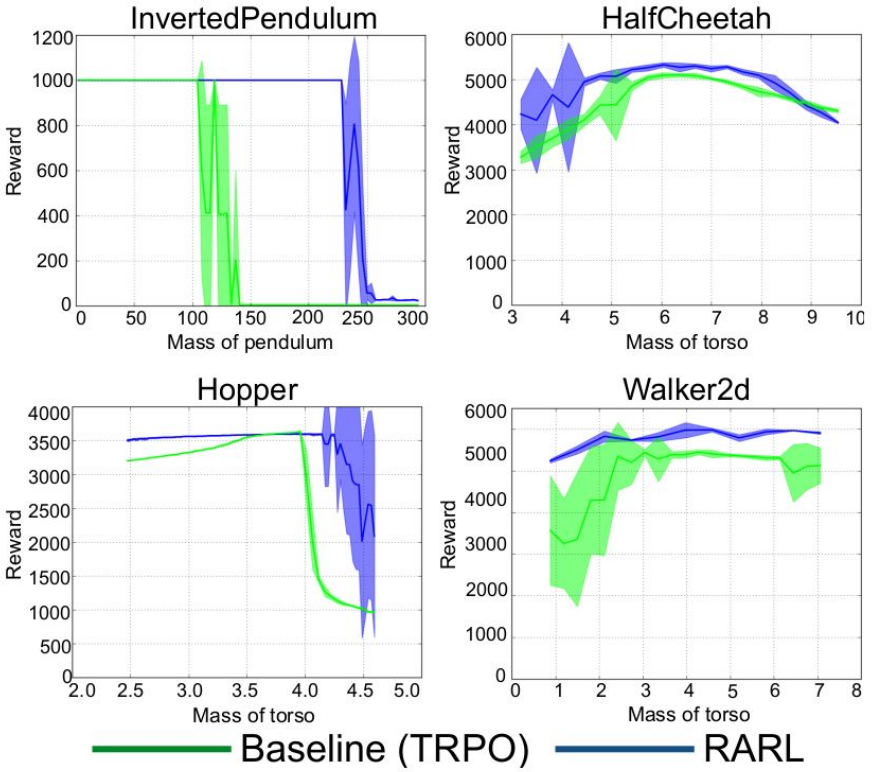
Враг обучался с **фиксированной** политикой протагониста.

На графике - награда в зависимости от перцентиля успешности политики



Эксперименты

Действия алгоритмов в зависимости от изменения внешних условий - массы агента



Список источников

Оригинальная статья

http://proceedings.mlr.press/v70/pinto17a/pinto17a.pdf