Роботизированное обучение по демонстрациям с использованием диффузионной модели и алгоритмов обучения с подкреплением

Гао Тяньцы

Московский государственный технический университет имени Н. Э. Баумана

Аннотация. В работе предлагается двухэтапный метод обучения робота по демонстрациям, сочетающий диффузионную генеративную модель и онлайн-дообучение методом обучение роботов по демонстрациям. На офлайн-фазе диффузионная модель использует ограниченный набор экспертных демонстраций и генерирует синтетические «псевдодемонстрации», позволяя расширить вариативность и охват исходного датасета. Это избавляет стратегию от узкой специализации и повышает её способность к обобщению. На онлайн-фазе робот с уже предобученной стратегией корректирует свои действия в реальной среде (или в высокоточной симуляции), что существенно снижает риски небезопасных действий и уменьшает число необходимых взаимодействий. Дополнительно введена параметрически-эффективная донастройка, сокращающая вычислительные затраты на онлайн-обучение, а также ценностное руководство, ориентирующее генерацию новых данных на области состояний и действий с высокими оценками Q. Эксперименты на задачах из набора D4RL (Hopper, Walker2d, HalfCheetah) показывают, что наш подход достигает наибольшей накопленной награды при меньших вычислительных затратах по сравнению с альтернативами. Анализ t-SNE свидетельствует о смещении синтетических данных в области пространства с высокими оценками Q, способствуя ускоренному обучению. Полученные результаты подтверждают перспективность предлагаемого метода для робототехнических приложений, где важно совмещать ограниченный объём демонстраций, безопасность и эффективность онлайн-фазы.

Ключевые слова: обучение роботов по демонстрациям, диффузионные генеративные модели, обучение с подкреплением.

Введение

Обучение роботов по демонстрациям (англ. *Learning from Demonstrations*, LfD) остаётся одной из важнейших проблем в робототехнике, поскольку во многих промышленных и сервисных сценариях требуется оперативная адаптация робота к новому навыку на основе ограниченного числа демонстрационных данных при одновременном соблюдении требований безопасности и минимизации объёма онлайн-взаимодействий [1, 2]. При этом один из существенных недостатков простого копирования поведения (англ. *Behavior Cloning*, BC) заключается в том, что исходный набор демонстраций может обладать недостаточной вариативностью и

охватом состояний, что в свою очередь нередко приводит к узкой специализации получаемой стратегии и возникновению ошибок при отклонении от обучающего распределения [3, 4].

Настоящая работа предлагает двухфазный метод, в котором:

1. Офлайн-фаза: применяется диффузионная генеративная модель (Denoising Diffusion Model) для многомодального моделирования имеющихся экспертных демонстраций и генерации дополнительных (виртуальных) образцов. Это даёт возможность восполнить неполноту исходного датасета и сформировать расширенную выборку «псевдодемонстраций» [5].

2. Онлайн-фаза: используется алгоритм Proximal Policy Optimization (PPO) для уточнения стратегии в реальной среде (или высокоточной имитации), но при жёстком лимите на число взаимодействий. Поскольку на данном этапе стратегия уже предварительно обучена офлайн (включая синтетические траектории из диффузионной модели), величина проб и ошибок, а также риск некорректных действий существенно снижаются.

В работе последовательно изложены следующие аспекты. Сначала описывается архитектура диффузионной модели, включая механизм пошагового добавления и удаления шума в траекториях, а также способ обучения сети, непосредственно предсказывающей шум (подход DDPM). Затем выводится формула PPO с учётом «обрезки» (*clipping*) отношения вероятностей и обучения сети ценности, что даёт объяснение стойкости этого алгоритма к резким изменениям стратегии. Далее предложенные компоненты — диффузионная модель и онлайн-обучение PPO — объединяются в единый конвейер (включающий офлайн- и онлайн-фазы), позволяющий как расширить охват обучающего распределения, так и достичь высоких итоговых наград при ограниченном числе онлайн-шагов.

В заключительной части работы приводятся комплексные эксперименты со следующими основными элементами:

- сравнение различных вариантов генерации дополнительных данных (Flow-based, базовая диффузия и диффузия с учётом ценностного руководства) и случая отсутствия генерации (No-Gen);
- демонстрация влияния параметрически-эффективной донастройки (PET), снижающей вычислительные затраты на онлайн-обучение при сохранении качества генерации [6-8];
- использование t-SNE-визуализации для исследования распределений «состояние-действие» и соответствующих Q-оценок в низкоразмерном пространстве;
- показано, что предложенный метод (Diff+PET+VG) обеспечивает наилучшие результаты по накопленной награде при наименьших вычислительных затратах (GPU Time) по сравнению с рассмотренными альтернативами.

Таким образом, описанный подход объединяет многомодальную генерацию, эффективное клонирование поведения и безопасное онлайн-обучение, что делает его перспективным для широкого круга робототехнических приложений, где имеется ограниченное число демонстраций и предъявляются жёсткие требования к уровню риска и затратам на обучение.

Методология

В данном разделе мы предлагаем и детально излагаем метод обучения робота по демонстрациям на основе сочетания диффузионных моделей и обучения с подкреплением. Сначала рассмотрим, как с помощью диффузионной вероятностной модели можно в офлайн-режиме смоделировать многомодальное распределение ограниченных экспертных демонстраций и, опираясь на это, сгенерировать дополнительные синтетические данные для расширения обучающей выборки [9, 10]. Далее опишем метод онлайн-доводки на основе PPO и с помощью формул покажем

его преимущество с точки зрения ограничения масштаба обновления стратегии [11, 12]. Наконец, рассмотрим процесс объединения этих двух подходов в единый конвейер роботизированного обучения по демонстрациям [13, 14].

A. Диффузионная модель: многомодальное моделирование демонстраций и синтез

В разделе A описывается, каким образом мы используем диффузионную модель для многомодального захвата и расширения (augmentation) исходных демонстраций, предоставленных экспертом.

A.1 Определение диффузионного процесса

Пусть D_{demo} обозначает датасет демонстраций, предоставленных экспертом, содержащий несколько траекторий $\{\tau_i\}_{i=1}^{N}$. Каждая отдельная траектория

$$\boldsymbol{\tau} = (\boldsymbol{s_0}, \boldsymbol{a_0}), \dots, (\boldsymbol{s_T}, \boldsymbol{a_T})$$

может быть «выпрямлена» (flatten) и рассматриваться как высокомерное наблюдение в пространстве $\mathbb{R}^d \times (T+1)$, где $s_t \in S \subset \mathbb{R}^{d_s}, a_t \in A \subset \mathbb{R}^{d_a}, d = d_s + d_a$. Диффузионная модель приближает истинное распределение демонстрационных данных путём определения прямой и обратной марковской цепи в этом пространстве.

Прямая диффузия (Forward Diffusion):

$$\boldsymbol{q(\tau_t \mid \tau_{t-1}) = \mathcal{N}(\tau_t; (1-\beta_t)\tau_{t-1}, \beta_t I), t = 1, \dots, T,} \quad (1)$$

где $\beta_t \in (0,1)$ — заранее заданная последовательность интенсивностей шума;

$\tau_0 \equiv \tau$ обозначает исходную демонстрационную траекторию. При достаточно большом t вектор τ_T приближается к изотропному гауссову распределению $\mathcal{N}(0, I)$.

Обратная диффузия (Reverse Diffusion): необходимо выучить параметризированное распределение: $p_\theta(\tau_{t-1} \mid \tau_t)$, которое приближало бы соответствующее апостериорное распределение $q(\tau_{t-1} \mid \tau_t)$. При хорошей аппроксимации мы можем начинать выборку с $\tau_T \sim \mathcal{N}(0, I)$ и, постепенно «удаляя шум», получить траекторию $\tau_0 \approx \tau$.

На рис.1 показаны ключевые блоки и потоки данных диффузионной модели: верхние компоненты соответствуют зашумлению (прямая диффузия), а нижние — денойзинговому процессу (обратная диффузия).

А.2 Функция потерь и предсказание шума

В работе DDPM предложен упрощённый метод обучения [15]: сеть $\varepsilon_\theta(\cdot)$ непосредственно предсказывает шум ε, содержащийся в τ_t. Это позволяет свести задачу обучения обратного распределения к задаче минимизации среднеквадратичной ошибки:

$$\boldsymbol{\tau_t = \sqrt{\bar{\alpha}_t}\tau_0 + \sqrt{1-\bar{\alpha}_t}\epsilon, \bar{\alpha}_t = \prod_{i=1}^{t}(1-\beta_i),}$$
$$\boldsymbol{L(\theta) = \mathbb{E}_{\tau_0 \sim q(\tau), \epsilon \sim \mathcal{N}(0,I), t \sim \mathrm{Uniform}\{1,\ldots,T\}} \|\epsilon - \epsilon_\theta(\tau_t, t)\|^2.} \quad (2)$$

Здесь τ_t — траектория, зашумлённая на t-м шаге прямой диффузии.

Когда модель ϵ_θ обучена, можно постепенно восстанавливать τ_0 по формуле:

$$\boldsymbol{\tau_{t-1} = \frac{1}{\sqrt{1-\beta_t}}\left(\tau_t - \beta_t \varepsilon_\theta(\tau_t, t)\right) + \sigma_t z, z \sim \mathcal{N}(0, I),}$$

где $\sigma_t \approx \beta_t$ (или выбирается по иной схеме). Таким образом, сеть ε_θ пошагово удаляет гауссов шум, генерируя синтетические траектории $\hat{\tau}$, близкие к исходным демонстрациям.

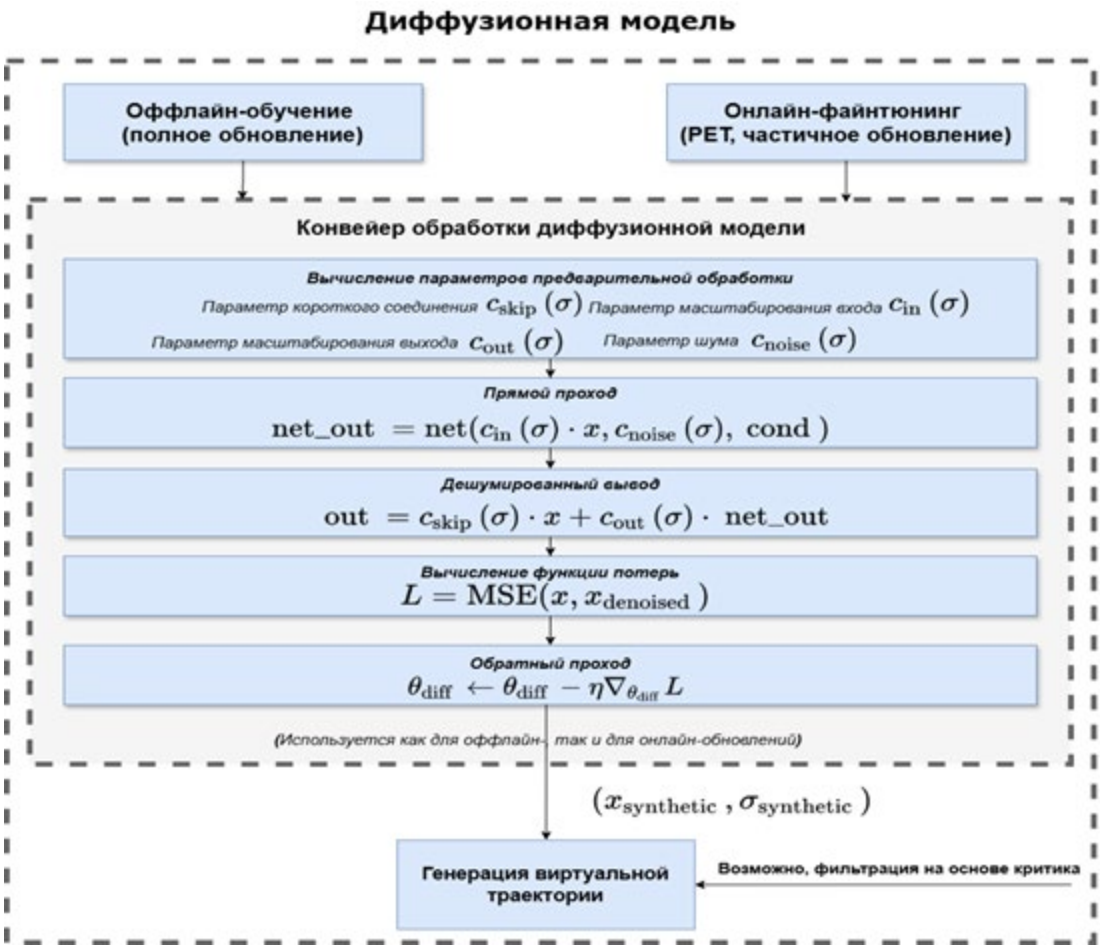

Рис. 1. – Конвейер обработки диффузионной модели.

A.3 Преимущества многомодальности и дополнение данных:

Обученная модель ε_θ способна сэмплировать «псевдодемонстрации» D_{syn}. Эти сгенерированные образцы статистически похожи на исходные, но содержат случайные возмущения или альтернативные варианты действий, обеспечивая многомодальность. Объединив их с исходными демонстрациями D_{demo}, формируем расширенный набор:

$$\boldsymbol{D_{\text{aug}} = D_{\text{demo}} \cup D_{\text{syn}}}.$$

Далее, в офлайн-режиме можно провести обучение поведения (BC, Behavior Cloning), минимизируя:

$$\boldsymbol{\min_{\theta} L_{\text{BC}}(\theta), \quad L_{\text{BC}}(\theta) = \mathbb{E}_{(s,a)\sim D_{\text{aug}}}[-\log \pi_\theta(a \mid s)]}.$$

Таким образом получается начальная стратегия π_θ, которая не только унаследовала знания эксперта, но и приобрела более широкий спектр действий благодаря сгенерированным данным.

B. Дообучение методом PPO

Теперь рассмотрим, как использовать расширенный набор данных на этапе онлайн-обучения с подкреплением (RL) для корректировки стратегии.

B.1 Целевая функция RL и функция преимущества

Будем моделировать задачу принятия решений роботом как процесс Маркова (MDP) (S, A, P, R, γ). На шаге t, выполняя действие a_t в состоянии s_t, агент получает мгновенное вознаграждение $r_t = r(s_t, a_t)$ и переходит в следующее состояние согласно $P(s_{t+1} \mid s_t, a_t)$. Стратегия $\pi_\phi(a \mid s)$ определяет распределение действий при заданном состоянии. Её качество оценивается ожидаемой суммой дисконтированных наград:

$$\boldsymbol{J(\phi) = \mathbb{E}_{\tau\sim\pi_\phi}[\sum_{t=0}^{\infty} \gamma^t r(s_t, a_t)]}.$$

С учётом идеи градиента стратегии [16], её оптимизацию по ϕ можно записать как:

$$\boldsymbol{\nabla_\phi J(\phi) = \mathbb{E}_{\pi_\phi}\left[\nabla_\phi \log \pi_\phi(a_t \mid s_t) A_t\right]}$$

где A_t — функция преимущества, которую часто оценивают как $A_t = Q(s_t, a_t) - V(s_t)$[16] или с помощью GAE [11].

B.2 Вывод формулы PPO

Хотя алгоритмы на основе градиента стратегии просты концептуально, при масштабном обучении может возникать «коллапс стратегии» или нестабильность. PPO предлагает ввести в функцию оптимизации отношение вероятностей и операцию «обрезки» (clipping), чтобы ограничивать расхождение между старой (ϕ_{old}) и новой (ϕ) стратегиями [11]:

$$\boldsymbol{r_t(\phi) = \frac{\pi_\phi(a_t \mid s_t)}{\pi_{\phi_{\text{old}}}(a_t \mid s_t)}}$$

Тогда целевая функция с обрезкой принимает вид:

$$\boldsymbol{L_{\text{clip}}(\phi) = \mathbb{E}_t[min(r_t(\phi)A_t, \text{clip}(r_t(\phi), 1-\epsilon, 1+\epsilon)A_t)]}$$

При добавлении в функцию потерь члена для обучения ценности:

$$\boldsymbol{L_{\text{value}}(\psi) = \mathbb{E}_t\left[\left(V_\psi(s_t) - G_t\right)^2\right]}$$

где G_t — целевая награда или оценка по временным разностям, итоговая функция PPO имеет вид:

$$\boldsymbol{\max_{\phi,\psi} L_{\text{clip}}(\phi) - c_1 L_{\text{value}}(\psi) + c_2 H(\pi_\phi)}$$

где $H(\pi_\phi)$ — энтропия стратегии, c_1 и c_2 — весовые коэффициенты.

На рис.2 показана структура PPO (Actor–Critic), процесс формирования цикла «получить данные (онлайн), вычислить A_t, обновить стратегию π_ϕ». Внизу условно показано, что часть данных (или их начальное распределение) происходит из расширенного набора D_{aug} .

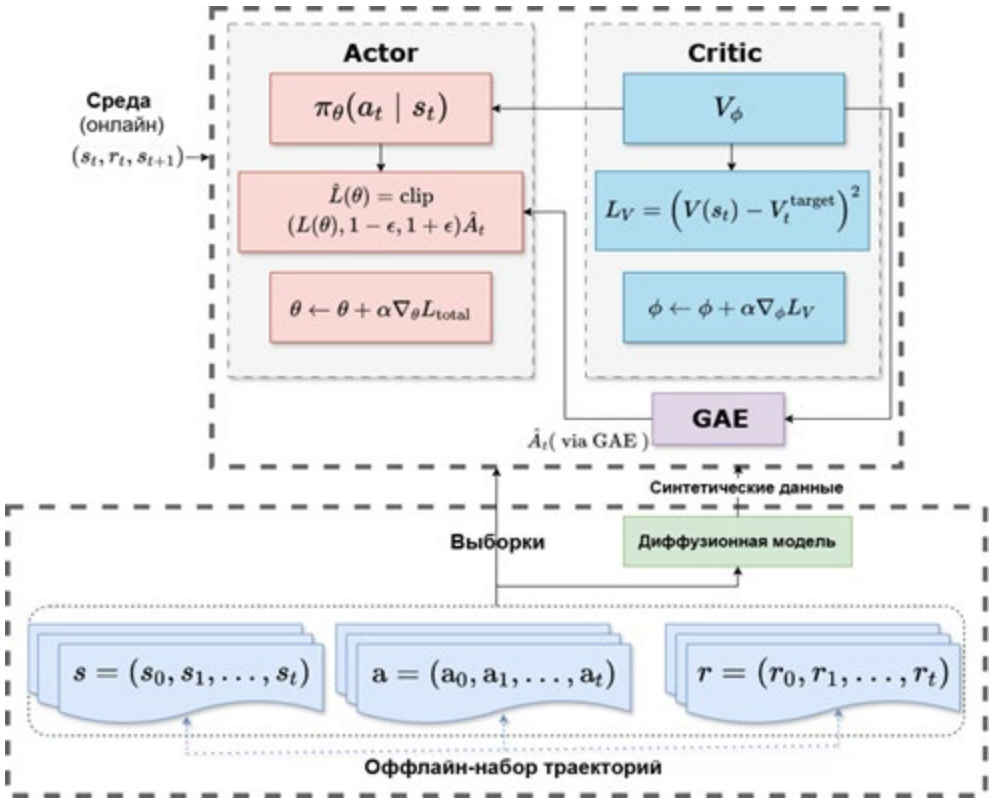

Рис. 2. – Архитектура PPO (Actor–Critic) с учётом расширенного набора данных.

B.3 Мелкомасштабная доводка при инициализации по демонстрациям

Поскольку на офлайн-этапе уже получена стратегия π_{θ_0}, предварительно обученная на расширенном датасете D_{aug}, то онлайн-обучение может быть запущено непосредственно с этой «экспертной» начальной точки. Робот при взаимодействии со средой собирает ограниченное число новых траекторий, вычисляет функцию преимущества A_t (напр., в соответствии с GAE) и обновляет стратегию по формуле (1). Благодаря тому, что π_{θ_0} уже обладает достаточным уровнем качества, объём проб и ошибок существенно сокращается, что одновременно снижает риск небезопасных действий и уменьшает общее время на поиск стратегии.

При необходимости можно ввести дополнительные механизмы ограничения масштаба обновления параметров — например, сохранять часть демонстрационных данных в онлайн-выборке или накладывать регуляризацию по KL-дивергенции, — чтобы предотвратить избыточное отклонение текущей политики от исходных экспертных траекторий. Такой подход позволяет поддерживать баланс между точной подстройкой стратегии под реальную среду и сохранением знаний, полученных из офлайн-фазы.

C. Объединение диффузионной модели и обучения с подкреплением: общая схема

Разобравшись с (1) принципами диффузионного моделирования (раздел A) и (2) онлайн-обучением PPO (раздел B), опишем, как эти подходы соединяются в единый процесс обучения робота по демонстрациям. Ключевая идея заключается в двухфазной структуре:

1. **Офлайн-фаза**. С помощью диффузионной модели глубоко

моделируется ограниченный набор демонстраций, генерируется синтетический набор D_{syn}, который объединяется с исходными данными D_{demo} в D_{aug}. Затем проводится обучение поведения (BC), формируя стратегию π_{θ_0}.

2. **Онлайн-фаза**. Используя улучшенную начальную стратегию π_{θ_0}, применяем PPO для доводки на реальном роботе или в высокоточной симуляции, получая «настоящие» вознаграждения и тем самым повышая качество конечной политики.

Как показано на рис. 3, розовые блоки соответствуют диффузионной модели, фиолетовые — сети PPO, а справа схематично отражён процесс «виртуальной» генерации траекторий и их фильтрации.

Таким образом, рис.3 иллюстрирует, как виртуальные траектории (сверху) и реальные демонстрации (слева) сливаются в расширенный датасет D_{aug}, который используется для предобучения (BC). Затем в онлайн-режиме (центральная часть рисунка) алгоритм PPO обновляет политику, собирая реальные (или симулированные) награды. При значительном изменении среды возможно повторное (пусть и частичное) дообучение диффузионной модели.

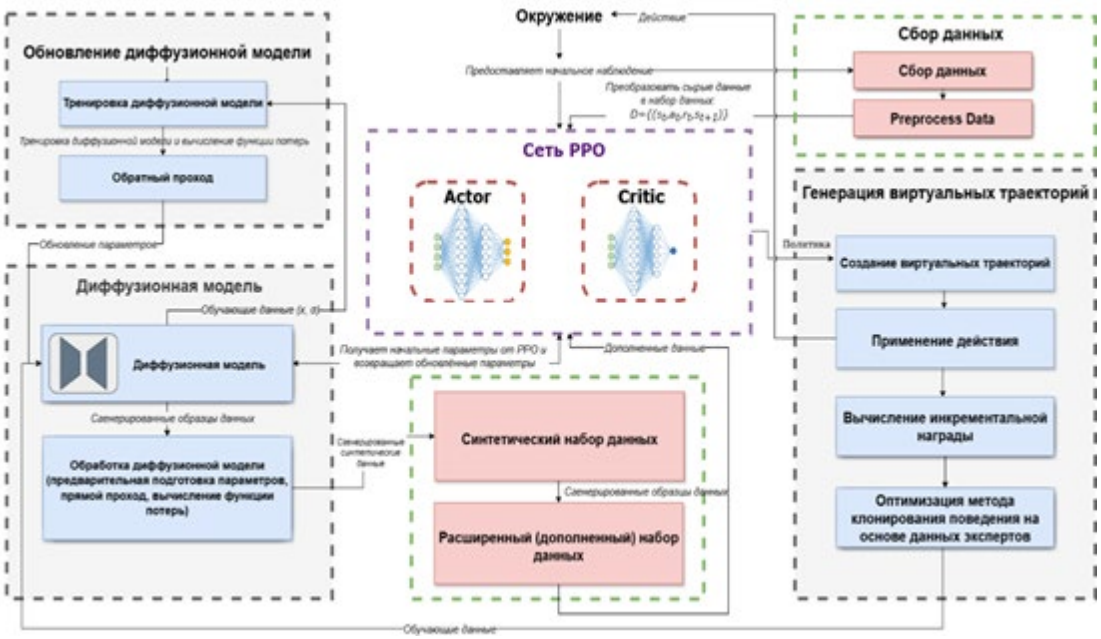

Рис. 3. – Схематическое представление объединённого процесса обучения с использованием диффузионной модели и PPO.

C.1 Офлайн-обучение диффузионной модели и дополнение данных

1. Исходные демонстрации и прямая диффузия

Входные данные: набор экспертных траекторий $D_{\text{demo}} = \{\tau_i\}_{i=1}^{N}$. При прямой диффузии траектория τ многократно зашумляется, формируя $\tau_1, \tau_2, \ldots, \tau_T$. Сеть $\varepsilon_\theta(\tau_t, t)$.

2. Предсказание шума и многократная обратная диффузия

Обучаем $\varepsilon_\theta(\tau_t, t)$, минимизируя MSE по формуле (2). Полученная модель способна восстанавливать τ_0 из $\tau_T \sim \mathcal{N}(0, I)$.

3. Генерация синтетических данных

Иногда требуется отбраковывать некорректные выборки. Итоговый синтетический набор D_{syn} объединяется с D_{demo} в D_{aug} (см. (5)).

C.2 Предобучение стратегии и онлайн-доводка

1. Офлайн-предобучение стратегии

- Клонирование поведения (BC): минимизируем $-\log \pi_\theta(a \mid s)$ на D_{aug}, получая начальную стратегию π_{θ_0}. При желании можно применить иные

методы офлайн-RL (CQL, BCQ и т.д.), но BC достаточно эффективно.

• Инициализация сети ценности: параллельно можно обучить простую сеть V_{ψ_0}, используя приблизительные награды или метки «успешности» в D_{aug} .

2. Онлайн-доводка по PPO

• В реальной среде или в точном симуляторе разворачиваем π_{θ_0} и собираем небольшое количество новых данных $\{(s_t, a_t, r_t)\}$.

• Оцениваем функцию преимущества $\hat{A}_t$ (к примеру, с GAE) и обновляем стратегию, максимизируя $L_{\text{clip}}(\theta)$ (1), одновременно обучая сеть ценности.

• Поскольку стратегия уже «экспертная», на доводку в большинстве случаев нужно мало итераций, что снижает затраты на онлайн-взаимодействия и уменьшает риск небезопасных действий.

Таким образом, предложенный метод сочетает в себе мощную многомодальную генерацию (диффузионная модель, раздел A) и безопасное/стабильное онлайн-обучение (PPO, раздел B), объединённые в общий офлайн/онлайн-конвейер (раздел C). В практическом плане это позволяет:

- расширять исходные демонстрации за счёт синтетических траекторий,
- экономить ресурсы за счёт уменьшения объёма онлайн-взаимодействий,
- поддерживать дальнейшую донастройку диффузионной модели при меняющихся условиях.

Эксперименты и результаты

В данном разделе проверяется эффективность предлагаемого подхода, сочетающего диффузионную генеративную модель и онлайн-обучение с подкреплением в формате «офлайн + ограниченная онлайн-фаза». Цель —

сравнить разные схемы генерации данных (Flow-based, чисто Diffusion, Diffusion с ценностным руководством) и вариант без генерации (No-Gen), а также проанализировать влияние параметрически-эффективной донастройки (PET) и ценностного руководства (VG).

А. Настройки экспериментов

1. Среды и наборы данных

- *Hopper, Walker2d, HalfCheetah* (из D4RL на симуляторе MuJoCo). Состояния и действия — непрерывные вектора.
- Используются датасеты *medium* или *medium-expert*, отражающие ограниченный, но качественный набор офлайн-демонстраций.
- Онлайн-фаза жёстко лимитирована по числу взаимодействий (обычно несколько сотен тысяч шагов).

2. Процедура обучения:

- Офлайн-этап: (а) обучается или дообучается генеративная модель (Flow-based или Diffusion) на исходных демонстрациях; (б) сгенерированные данные объединяются с реальными, формируя расширенный набор для предобучения стратегии.
- Онлайн-этап: применяется PPO при ограниченном числе реальных шагов взаимодействия. Для варианта Diff+PET+VG часть параметров замораживается (PET), а генерация учитывает Q-фильтрацию (VG).

3. Сравниваемые методы

- No-Gen: без генерации; используется лишь исходный офлайн-набор + небольшая онлайн-доводка (PPO).
- Flow-based: дополнительная генерация с помощью Flow-модели (напр. RealNVP).
- Diff-NoVG (или просто Diffusion): диффузионная модель для генерации **без** ценностного руководства.
- Diff+PET+VG (наш): диффузионная модель + параметрически-

эффективная донастройка (PET) + ценностная фильтрация (VG).

4. Метрики

• Cumulative Reward (накопленная награда): среднее вознаграждение стратегии валидационных эпизодов (выше — лучше).

• GPU Time (загрузка/время на GPU): усреднённое время вычислений на эпоху; отражает вычислительную эффективность.

• Визуализация (t-SNE): для анализа распределения (s, a)-пар и оценки Q-сети в низкомерном пространстве.

B. Результаты по награде и использованию GPU

B.1 Награда и вычислительные затраты (GPU Time)

На рис.4 представлены кривые накопленной награды (слева) и времени вычислений на GPU (справа) при обучении в трёх средах: *Hopper, Walker2d, HalfCheetah* (по 200 эпох). Изначально каждая стратегия офлайн-предобучена, а затем дообучается онлайн, но при ограниченном числе взаимодействий. Итоговые количественные показатели (ближе к 200-й эпохе) сведены в таблице 1.

Таблица №1.

Итоговые результаты

Среда	Метод	Reward (к финалу)	GPU Time, сек/эпоха
Hopper	No-Gen	~1600 (к ~50-й эпохе)	~6.5–6.6
	Flow-based	~1700–1750	6.3–6.5
	Diff-NoVG	~1700–1750	6.3–6.5
	Diff+PET+VG	~1800 (после ~150-й эпохи)	~6.2
Walker2d	No-Gen	<3200	>6.8 → ~6.5
	Flow-based	3300–3350	6.4–6.5
	Diff-NoVG	3300–3350	6.4–6.5
	Diff+PET+VG	>3500	~6.2
HalfCheetah	No-Gen	~2300	~6.5–6.6
	Flow-based	2400–2450	~6.5–6.6

	Diff-NoVG	2400–2450	~6.4
	Diff+PET+VG	>2500	~6.2

- No-Gen демонстрирует наиболее скромные показатели накопленной награды; кроме того, время на GPU остаётся на уровне ~6.5–6.6 сек/эпоха в финале обучения.
- Flow-based и Diff-NoVG оба дают более высокие значения награды по сравнению с No-Gen, а также немного меньшую среднюю загрузку GPU (~6.3–6.5 сек/эпоха).
- Diff+PET+VG показывает самую высокую итоговую награду в каждой из трёх сред и одновременно минимальное время вычислений (~6.2 сек/эпоха), подчёркивая эффективность параметрически-эффективной донастройки (PET) и ценностного руководства (VG).

Таким образом, комплексные показатели «накопленная награда / GPU Time» оказываются наилучшими у Diff+PET+VG, что подтверждает важность сочетания диффузионной генерации, PET и VG.

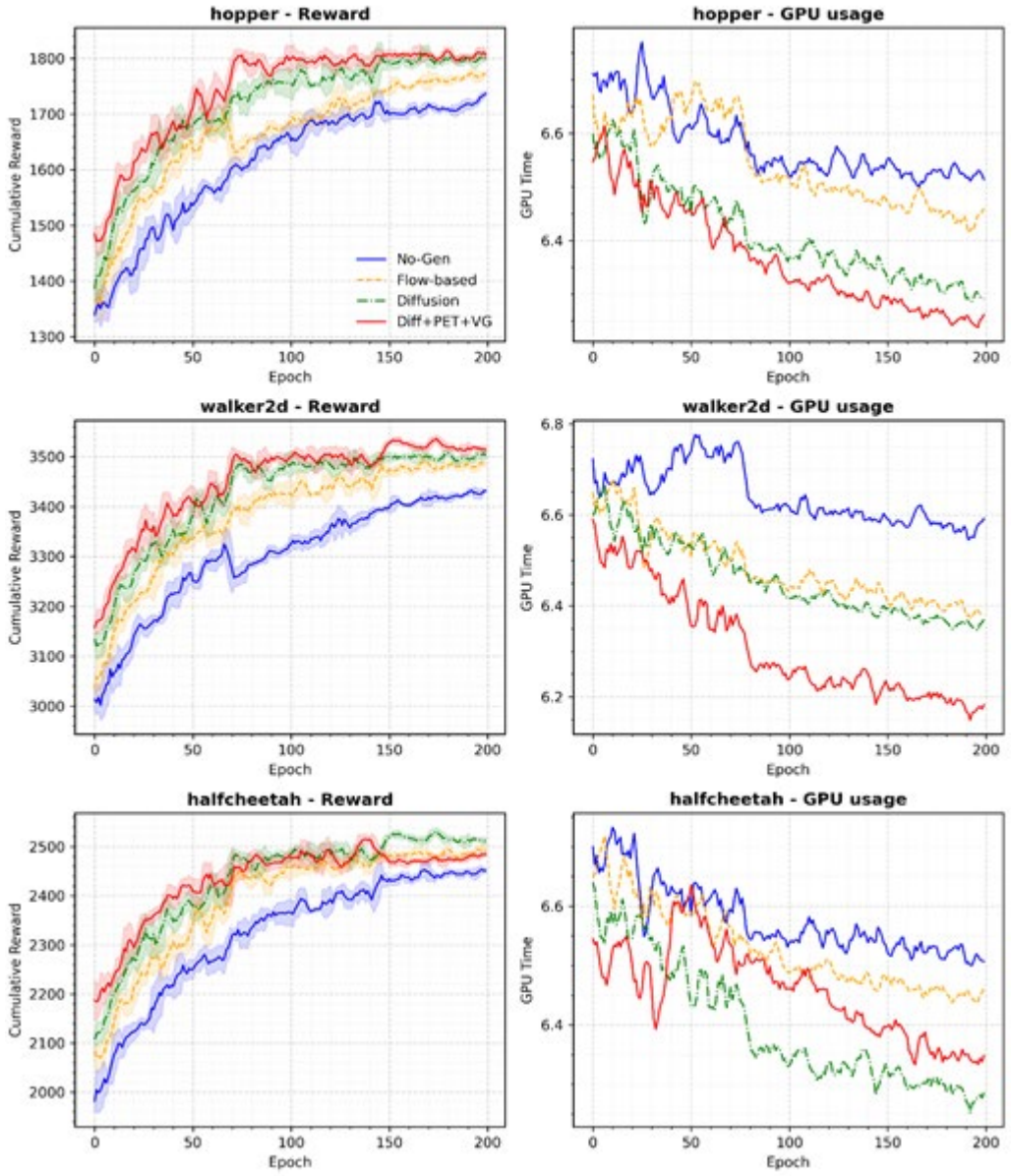

Рис. 4. – Накопленная награда (слева) и время вычислений на GPU (справа) в трёх средах *(Hopper, Walker2d, HalfCheetah; сравнение No-Gen, Flow-based, Diff-NoVG, Diff+PET+VG)*

C. t-SNE-визуализация и анализ ценности

Чтобы глубже понять различия в распределениях (s, a), был применён метод t-SNE (см. рис.5). Каждая точка соответствует конкретному состоянию s и действию a, а цвет указывает на оценку $Q(s, a)$ по Critic-сети. При этом:

- Offline (круги) – исходные офлайн-демонстрации;

- Diff-NoVG (треугольники) – данные, сгенерированные диффузионной моделью без учёта Q-оценки;
- Diff-VG (квадраты) – выборка, полученная при активном ценностном руководстве (VG).

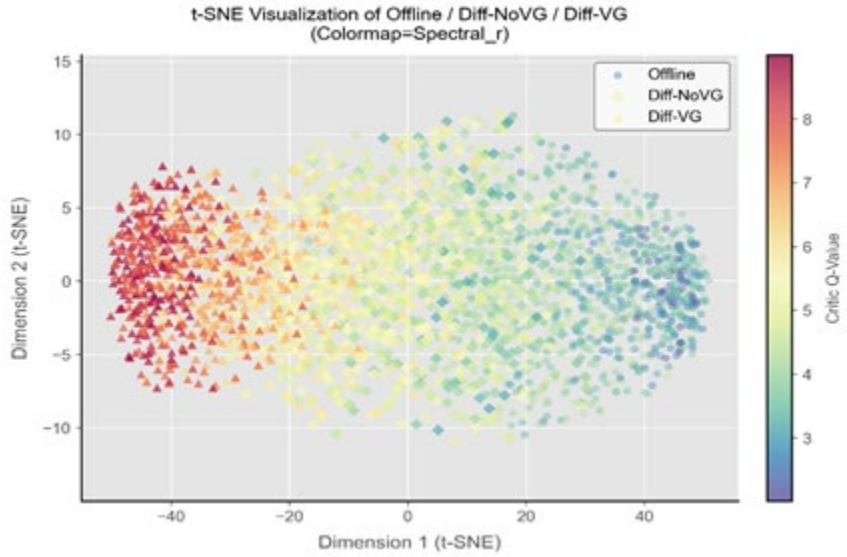

Рис. 5. – t-SNE-визуализация (цвет = Q, форма = источник данных) для сравнения Offline, Diff-NoVG и Diff-VG

Видно, что Offline-пробы располагаются в достаточно компактной зоне с умеренными значениями Q. Вариант Diff-NoVG шире охватывает пространство, однако нередко пропускает области с наибольшим Q. Напротив, Diff-VG явно сдвигается к более «выгодным» регионам (цвет ближе к жёлтому), что говорит о целенаправленном формировании траекторий в зонах с высокими оценками награды. Тем самым VG выступает важным механизмом фильтрации и генерации данных для ускоренного обучения.

Таким образом, VG целенаправленно сдвигает генерируемые траектории в «полезные» регионы. При этом PET (не отражён напрямую на рис.5) помогает дообучать модель без «катастрофического забывания» и снижать вычислительные затраты.

D. Обсуждение и выводы

1. Сочетание Diffusion с PET и VG

Эксперименты показывают, что совместное использование диффузионной генерации (способной моделировать многомодальные действия), механизма PET (сокращающего вычислительные затраты на онлайн-этапе) и VG (ценностного руководства) даёт наилучшие итоги как по награде, так и по времени на GPU.

2. Сравнение с Flow-based

Flow-модели приносят определённый выигрыш по сравнению с полным отсутствием генерации (No-Gen), но для более сложных распределений диффузионный подход показывает более высокие результаты — как по итоговым наградам, так и по эффективному охвату состояний и действий.

3. Анализ распределения и устойчивость

t-SNE (рис. 5) демонстрирует, что VG-сэмплы предпочитают области с более высокими оценками Q. Рост средней награды на рис. 4 согласуется с этим сдвигом. Кроме того, PET позволяет уменьшить обновляемые параметры, снижая риск дестабилизации модели.

4. Ограничения и направления будущих исследований

- Для более сложных задач (например, многозадачные сценарии или реальные роботизированные системы) потребуется расширенный офлайн-набор и гибрид датчиков (визуальных, тактильных и т.п.).
- Тонкая настройка критериев фильтрации при VG может ещё сильнее повысить качество генерируемых данных.
- В условиях реального мира необходима дополнительная проверка безопасности: хотя PET и VG снижают риск неудачных действий, практическая эксплуатация требует протоколов безопасного обучения на аппаратном роботе.

Итог: метод Diff+PET+VG даёт наилучшее сочетание «накопленная награда / вычислительная эффективность» в каждой из тестируемых сред, что доказывает релевантность предложенных механизмов (диффузия, PET, VG) для обучения по демонстрациям с ограниченной онлайн-фазой.

Заключение

В данной работе мы предложили двухэтапный метод обучения робота по демонстрациям, сочетающий диффузионную генеративную модель и онлайн-обучение PPO. В офлайн-фазе с помощью диффузии удаётся многомодально расширять исходный набор демонстраций, формируя синтетические «псевдодемонстрации», тогда как онлайн-дообучение методом PPO с учётом уже предобученной стратегии позволяет эффективно повышать качество управления при ограниченном количестве взаимодействий со средой. Дополнительно мы проиллюстрировали пользу параметрически-эффективной донастройки (PET), уменьшающей вычислительные затраты, а также ценностного руководства (VG), фокусирующего модель на состояниях и действиях с более высокими оценками Q.

Основные итоги включают:

1. Повышение качества стратегии: синтетические демонстрации, сгенерированные диффузионной моделью, охватывают многомодальные варианты действий, дополняя исходную офлайн-выборку. Это помогает избежать узкой специализации и улучшает итоговую политику.

2. Экономия онлайн-ресурсов: благодаря тому, что робот уже обладает «экспертным» предобучением, в онлайн-фазе ему нужно меньше проб и ошибок, а значит, снижаются риски небезопасных действий и затраты времени.

3. Стабильность и эффективность вычислений: механизмы PET и VG обеспечивают одновременно более высокую итоговую награду и меньшую нагрузку на GPU; параметрическая донастройка сокращает число

обновляемых весов, а ценностная фильтрация повышает целевое качество генерируемых данных.

4. Применимость в реальных сценариях: предложенный подход может быть адаптирован к роботизированным системам с различной конфигурацией сенсоров и большим разнообразием задач, где важно уметь быстро обучаться по небольшому набору экспертных демонстраций.

В ходе экспериментов мы продемонстрировали превосходство предлагаемого решения (Diff+PET+VG) над альтернативными вариантами (Flow-based, Diffusion без ценностного руководства и отсутствие генерации), что выражается в более высокой накопленной награде и меньших вычислительных затратах. Анализ t-SNE подтвердил, что ценностное руководство помогает «перенастроить» модель в области состояний-действий с высокими Q-значениями, повышая результативность обучения.

Перспективы развития затрагивают несколько направлений:

- Апробация метода на сложных реальных роботах с мультимодальными датчиками (включая зрение и тактильную информацию), где расширение демонстраций особенно критично.
- Улучшение фильтрации при генерации с учётом дополнительных критериев безопасности и успешности, чтобы ещё точнее выбирать «перспективные» сэмплы.
- Исследование гибридных схем обучения, объединяющих диффузию, имитационное обучение и другие методы (например, CQL, BCQ, IQL) в более сложных средах.

Таким образом, результаты подтверждают, что комбинация диффузионного моделирования и ограниченной онлайн-фазы RL может существенно расширить возможности обучения по демонстрациям, обеспечивая многомодальный охват, высокую итоговую награду и экономию ресурсов. Это делает предложенный подход одним из перспективных

решений для робототехнических систем, где критичны безопасность, адаптивность и ограниченное количество доступных демонстраций.

Литература (References)

1. Argall B.D., Chernova S., Veloso M., Browning B. A survey of robot learning from demonstration // Robotics and autonomous systems. 2009. V. 57(5). pp. 469-483.
2. Schaal S., Ijspeert A., Billard A. Computational approaches to motor learning by imitation // Philosophical Transactions of the Royal Society of London. Series B: Biological Sciences. 2003. V. 358(1431). pp. 537-547.
3. Zhu Z., Lin K., Jain A.K., Zhou J. Transfer learning in deep reinforcement learning: A survey // IEEE Transactions on Pattern Analysis and Machine Intelligence. 2023. V. 45(11). pp. 13344 – 13362.
4. Taniguchi T., Mochihashi D., Nagai T., Uchida S., Inoue N., Kobayashi I., Inamura T. Survey on frontiers of language and robotics // Advanced Robotics. 2019. V.33 (15-16). pp. 700-730.
5. Urain J., Mandlekar A., Du Y., Shafiullah M., Xu D., Fragkiadaki K., Chalvatzaki G., Peters J. Deep generative models in robotics: A survey on learning from multimodal demonstrations. URL: arxiv.org/abs/2408.04380.
6. Houlsby N., Giurgiu A., Jastrzebski S., Morrone B., De Laroussilhe Q., Gesmundo A., Gelly S. Parameter-efficient transfer learning for NLP // International conference on machine learning. 2019, May. pp. 2790-2799.
7. Nakamoto M., Mees O., Kumar A., Levine S. Steering your generalists: Improving robotic foundation models via value guidance. URL: arxiv.org/abs/2410.13816.
8. Ding N., Qin Y., Yang G., Wei F., Yang Z., Su Y., Sun M. Parameter-efficient fine-tuning of large-scale pre-trained language models // Nature Machine Intelligence. 2023. V.5 (3). pp. 220-235.

9. Silver D., Huang A., Maddison C.J., Guez A., Sifre L., Van Den Driessche G., Hassabis D. Mastering the game of Go with deep neural networks and tree search // Nature. 2016. V.529 (7587). pp. 484-489.
10. Kingma D.P. Auto-encoding variational bayes. URL: arxiv.org/abs/1312.6114.
11. Schulman J., Wolski F., Dhariwal P., Radford A., Klimov O. Proximal policy optimization algorithms. URL: arxiv.org/abs/1707.06347.
12. Nichol A.Q., Dhariwal P. Improved denoising diffusion probabilistic models // International conference on machine learning. 2021, July. pp. 8162-8171.
13. Pathak D., Agrawal P., Efros A.A., Darrell T. Curiosity-driven exploration by self-supervised prediction // International conference on machine learning. 2017, July. pp. 2778-2787.
14. Mnih V., Kavukcuoglu K., Silver D., Rusu A.A., Veness J., Bellemare M.G., Hassabis D. Human-level control through deep reinforcement learning // Nature. 2015. V.518 (7540). pp. 529-533.
15. Ho J., Jain A., Abbeel P. Denoising diffusion probabilistic models // Advances in neural information processing systems. 2020. V. 33. pp. 6840-6851.
16. Sutton R.S. Reinforcement learning: An introduction. Canbridge MA: The MIT Press, 2018. 552 p.

Дата поступления: 19.01.2025
Дата публикации: 10.03.2025