



МАГИСТЕРСКАЯ ДИССЕРТАЦИЯ

**Optimal Flow Matching:  
новый подход к генеративному моделированию и  
оптимальному транспорту с прямыми траекториями после  
одной процедуры минимизации**

Магистерская образовательная программа:  
09.04.01 Информатика и вычислительная техника  
Интеллектуальный анализ данных

Студент: \_\_\_\_\_ Никита Корнилов  
*подпись*

Научный руководитель: \_\_\_\_\_ Александр Гасников  
*подпись*  
д.ф.-м.н., профессор

Со-руководитель \_\_\_\_\_ Александр Коротин  
*подпись*  
к.ф.-м.н., научный  
сотрудник

Москва 2025

Авторское право 2025. Все права защищены.

**Optimal Flow Matching:  
новый подход к генеративному моделированию и оптимальному  
транспорту с прямыми траекториями после одной процедуры  
минимизации**  
Никита Корнилов

Представлено в Московский физико-технический институт  
Июнь 19

### **Аннотация**

В последние несколько лет в области генеративного моделирования активно развиваются Flow Matching (FM) методы. Одно из интригующих свойств этих методов, это возможность обучать потоки с прямыми траекториями, реализующими оптимальные транспортные (Optimal Transport, OT) перемещения. Прямолинейность траекторий критически важна для быстрого и качественного семплирования из моделей.

Однако большинство существующих методов спрямления траекторий основаны на нетривиальных итеративных процедурах, которые накапливают ошибку в процессе обучения, или используют эвристики, опирающиеся на мини-батч OT.

Чтобы исправить эти недостатки, я разработал и теоретически обосновал новый подход — Optimal Flow Matching (OFM), который позволяет восстановить решение оптимального транспорта для квадратичной функции потерь всего за один шаг FM минимизации. Основная идея этого подхода заключается в использовании прямых векторных полей, параметризованных выпуклыми функциями. Все теоретические свойства и практическая эффективность метода были подтверждены в большом количестве экспериментов на реальных прикладных задачах.

# Оглавление

<b>1 Введение</b>	<b>4</b>
<b>2 Вклад Автора</b>	<b>5</b>
<b>3 Публикации</b>	<b>6</b>
<b>4 Обзор Литературы</b>	<b>7</b>
Динамический оптимальный транспорт . . . . .	7
Статический оптимальный транспорт . . . . .	7
Flow Matching (FM) . . . . .	8
Action Matching . . . . .	9
Optimal Transport Conditional Flow Matching (OT-CFM) . . . . .	9
Rectified Flow (RF) . . . . .	10
Заключение . . . . .	11
<b>5 Новый подход к генеративному моделированию</b>	<b>12</b>
5.1 Теория: Вывод функции потерь . . . . .	12
5.2 Свойства OFM . . . . .	13
5.3 Детали реализации на практике . . . . .	14
<b>6 Численные эксперименты</b>	<b>16</b>
6.1 Основные эксперименты . . . . .	16
2D Иллюстрации . . . . .	16
ОТ Бенчмарк . . . . .	17
Непарный перевод изображений . . . . .	18
Амортизация . . . . .	19
Вычислительные ресурсы и время . . . . .	20
<b>7 Обсуждение и заключение</b>	<b>22</b>
7.1 Сравнение с релевантными работами . . . . .	22
7.2 Ограничения OFM . . . . .	22
7.3 Заключение . . . . .	23
<b>Литература</b>	<b>24</b>
<b>Приложение</b>	<b>28</b>
7.4 Доказательства . . . . .	28
7.5 Action Matching . . . . .	33
7.6 Технические детали экспериментов . . . . .	35
Реализация OFM . . . . .	35
Детали бенчмарка . . . . .	36

# Глава 1

## Введение

Последние успехи в генеративном моделировании [1, 2, 3] в основном связаны с моделями Flow Matching (FM) [4]. Эти модели преобразуют начальное вероятностное распределение в целевое с помощью обычных дифференциальных уравнений (ОДУ), описывающих последовательный процесс изменения распределения. Однако такие процессы обычно имеют криволинейные траектории, что приводит к трудоемкому и длительному интегрированию ОДУ при сэмплировании из получившегося распределения. Чтобы решить эту проблему, были разработаны улучшения FM [5, 6, 7], направленные на выпрямление траекторий.

Метод Rectified Flow (RF) [5, 6] итеративно несколько раз минимизирует FM задачу, постепенно корректируя и спрямляя траектории. Однако с каждой минимизацией RF накапливает ошибку (см. [6, §2.2] и [5, §6]), что может ухудшить качество работы метода. Другое популярное направление для спрямления траекторий основано на связи прямых путей с оптимальным транспортом (ОТ) [8]. Основная цель ОТ — найти способ преобразования одного распределения в другое с минимальными затратами на перемещение вероятностных масс. Такие ОТ перемещения обычно описываются ОДУ с прямолинейными траекториями.

В методе OT Conditional Flow Matching (OT-CFM) [7, 9] предлагается применять FM к ОТ-решению между батчами из рассматриваемых распределений. Однако эта эвристика не гарантирует прямых траекторий из-за конечного размера мини-батчей (см., например, [9, Рис. 1, справа] для наглядной иллюстрации).

**Положения, выносимые на защиту.** В моей магистерской диссертации я устранило указанные выше проблемы методов спрямления траекторий. Я предлагаю новый подход **Optimal Flow Matching** (OFM), который уже после **одной** процедуры минимизации обеспечивает прямолинейные траектории, не требующие решения ОДУ для сэмплирования. Метод восстанавливает решение задачи оптимального транспорта с квадратичной функцией стоимости, то есть решает задачу Бенаму–Бренье.

**Научная новизна.** Основная идея OFM заключается в рассмотрении во время минимизации FM только определенных векторных полей, которые *по построению* дают прямые траектории. Эти векторные поля являются градиентами выпуклых функций, параметризуемых на практике с помощью Input Convex Neural Networks [10]. Эти поля являются характерными решениями для задачи ОТ. В OFM можно опционально использовать minibatch ОТ или любой другой транспортный план — и это имеет *полное теоретическое обоснование*.

## **Глава 2**

# **Вклад Автора**

В рамках данного диссертационного исследования мой вклад состоит в следующем:

- **Концепт и теоретическое обоснование нового подхода в генеративном моделировании:**
  - Разработка основной идеи метода и его практической реализации,
  - Полное теоретическое обоснование, включая все математические доказательства,
- **Экспериментальная часть:**
  - Экспериментальная проверка эффективности предложенного метода,
  - Планирование и проведение расширенных сравнительных экспериментов (совместно с П. Мокровым для оптимизации временных затрат и перекрёстной проверки результатов),
- **Подготовка рукописи:**
  - Написание основного текста диссертации (за исключением некоторых технических деталей экспериментов, предоставленных П. Мокровым).

Научное руководство, рецензирование текста, а также генерацию новых исследовательских идей и направлений осуществляли А. В. Гасников и А. А. Коротин.

## **Глава 3**

# **Публикации**

На основе этой магистерской работы были опубликована следующая статья на  $A^*$  конференции NeuralPS 2024:

1. Kornilov, Nikita, et al. "Optimal flow matching: Learning straight trajectories in just one step." Advances in Neural Information Processing Systems 37 (2024): 104180-104204.

# Глава 4

## Обзор Литературы

Сначала я расскажу теоретические основы оптимального транспорта, необходимые для дальнейших доказательств.

### **Динамический оптимальный транспорт**

Ещё до того как генеративные модели приобрели свою нынешнюю популярность, исследователи изучали связанную с ними задачу динамического оптимального транспорта (Dynamic OT) [11]. Основная цель Dynamic OT - найти векторное поле  $u$ , которое преобразует распределение  $p_0$  в распределение  $p_1$  с минимальными затратами. Такие отображения обычно описываются ОДУ с прямолинейными траекториями. Промежуточное распределение, генерируемое полем  $u$  в момент времени  $t \in [0, 1]$ , обозначается как  $p_t^u$ . Формально, Dynamic OT представляет собой следующую задачу минимизации:

$$\begin{aligned} \mathbb{W}_2^2(p_0, p_1) = \min_u & \quad \int_0^1 \int_{\mathbb{R}^D} \frac{|u_t(x_t)|_2^2}{2} p_t^u(x_t) dx dt, \\ \text{s.t.} & \quad p_1^u = p_1. \end{aligned} \tag{4.1}$$

В (4.1) ищутся векторные поля  $u$ , определяющие потоки, которые начинаются в  $p_0$  и заканчиваются в  $p_1$ . Среди таких потоков выбирается поле с минимальной кинетической энергией на всём временном интервале. На практике задача OT обычно решается без учета динамики, используя статическую постановку.

### **Статический оптимальный транспорт**

**Формулировки Монжа и Канторовича.** Формулировка задачи оптимального транспорта по Монжу имеет вид:

$$\inf_{T \# p_0 = p_1} \int_{\mathbb{R}^D} c(x_0, T(x_0)) p_0(x_0) dx_0, \tag{4.2}$$

где инфимум берется по измеримым функциям  $T : \mathbb{R}^D \rightarrow \mathbb{R}^D$ , удовлетворяющим условию сохранения массы  $T \# p_0 = p_1$ . Такие функции называются транспортными отображениями. Если существует транспортное отображение  $T^*$ , достигающее инфимума, оно называется оптимальным транспортным отображением.

Поскольку оптимальное транспортное отображение  $T^*$  в формулировке Монжа может не существовать, то была предложена релаксация Канторовича для задачи (4.2). Рассмотрим множество транспортных планов  $\Pi(p_0, p_1)$ , т.е. множество совместных распределений на  $\mathbb{R}^D \times \mathbb{R}^D$ , чьи маргины равны  $p_0$  и  $p_1$  соответственно. Формулировка Канторовича имеет вид:

$$\inf_{\pi \in \Pi(p_0, p_1)} \int_{\mathbb{R}^D \times \mathbb{R}^D} c(x_0, x_1) \pi(x_0, x_1) dx_0 dx_1. \tag{4.3}$$

При слабых предположениях на  $p_0, p_1$  инфимум всегда достигается (возможно, не единственным образом). Оптимальный план  $\pi^* \in \Pi(p_0, p_1)$  называется оптимальным транспортным

планом. Если оптимальный  $\pi^* = [\text{id}, T^*] \# p_0$ , то  $T^*$  является решением формулировки Монжа (4.2).

**Квадратичная функция стоимости.** В случае квадратичной функции стоимости транспорта  $c(x_0, x_1) = \frac{|x_0 - x_1|^2}{2}$  инфимумы в обеих формулировках ОТ (Монжа и Канторовича) всегда достигаются единственным образом [8, Теорема Бренье 2.12]. Они связаны соотношением  $\pi^* = [\text{id}, T^*] \# p_0$ . Более того, оптимальные значения (4.2) и (4.3) равны друг другу. Квадратный корень из оптимального значения называется расстоянием Вассерштейна  $\mathbb{W}_2(p_0, p_1)$  между распределениями  $p_0$  и  $p_1$ , т.е.:

$$\begin{aligned} \mathbb{W}_2^2(p_0, p_1) &:= \min_{\pi \in \Pi(p_0, p_1)} \int_{\mathbb{R}^D \times \mathbb{R}^D} \frac{|x_1 - x_0|^2}{2} \pi(x_0, x_1) dx_0 dx_1 \\ &= \min_{T \# p_0 = p_1} \int_{\mathbb{R}^D} \frac{|x_0 - T(x_0)|^2}{2} p_0(x_0) dx_0. \end{aligned} \quad (4.4)$$

Задача (4.4) имеет эквивалентную двойственную форму [8]:

$$\mathbb{W}_2^2(p_0, p_1) = \text{Const}(p_0, p_1) - \min_{\text{выпуклые } \Psi} \underbrace{\left[ \int_{\mathbb{R}^D} \Psi(x_0) p_0(x_0) dx_0 + \int_{\mathbb{R}^D} \bar{\Psi}(x_1) p_1(x_1) dx_1 \right]}_{=: \mathcal{L}_{OT}(\Psi)} \quad (4.5)$$

где минимум берется по выпуклым функциям  $\Psi(x) : \mathbb{R}^D \rightarrow \mathbb{R}$ .

Здесь  $\bar{\Psi}(x_1) := \sup_{x_0 \in \mathbb{R}^D} [\langle x_0, x_1 \rangle - \Psi(x_0)]$  - выпуклая сопряженная (по Фенхелю) функция к  $\Psi$ . Слагаемое  $\text{Const}(p_0, p_1)$  не зависит от  $\Psi$ . Следовательно, минимизация (4.4) по транспортным планам  $\pi$  эквивалентна минимизации  $\mathcal{L}_{OT}(\Psi)$  из (4.5) по выпуклым функциям  $\Psi$ . Более того, оптимальное транспортное отображение  $T^*$  может быть выражено через оптимальную  $\Psi^*$  (так называемый *потенциал Бренье* [8]), а именно:

$$T^* = \nabla \Psi^*. \quad (4.6)$$

Решение  $u^*$  для динамической задачи ОТ (4.1) строится так, чтобы генерировать прямолинейные траектории, соединяющие точки  $x$  и  $\nabla \Psi^*(x)$  для всех  $x \in \mathbb{R}^D$ .

**Солверы ОТ.** Существует множество солверов для решения непрерывной задачи ОТ [12, 13, 14, 15, 16, 17, 18, 19, 20], описание которых можно найти в обзоре [21]. Солверы на основе ICNN [13, 14, 20] непосредственно минимизируют целевую функцию  $\mathcal{L}_{OT}$  из (4.5), параметризую класс выпуклых функций с помощью выпуклых по входу нейронных сетей (ICNNs) [10]. Детали реализации солверов могут различаться, но основная идея остается неизменной. Для вычисления сопряженной функции  $\bar{\Psi}(x_1)$  в точке  $x_1$  они решают задачу выпуклой оптимизации из определения сопряженной функции.

## Flow Matching (FM)

Метод Flow Matching [4] стал первым подходом, успешно применившим идеи диффузионных моделей к произвольным распределениям. Авторы предлагают и теоретически и практически обосновывают новую FM-функцию потерь для устойчивого и эффективного обучения. Они задают линейный интерполянт  $x_t = (1 - t)x_0 + tx_1$  для любого момента времени  $t \in [0, 1]$ , где  $x_0$  и  $x_1$  берутся из совместного распределения  $\pi$ , чьи маргиналы равны  $p_0$  и  $p_1$  соответственно. Распределение  $\pi$  называется транспортным планом. Далее они выучивают векторное поле этого линейного интерполянта в точке  $x_t$ , используя несмещенную оценку

$x_1 - x_0$ . Это достигается путем решения:

$$\min_u \int_0^1 \int_{\mathbb{R}^D \times \mathbb{R}^D} |u_t(x_t) - (x_1 - x_0)|^2 \pi(x_0, x_1) dx_0 dx_1 dt,$$

$$x_t = (1-t)x_0 + tx_1.$$

Множество траекторий, генерируемых решением FM, обладает полезным свойством: итоговое распределение всегда равно  $p_1$  для любого начального транспортного плана  $\pi$ . Более того, маргинальные распределения решения и линейного интерполянта совпадают для любого момента времени  $t \in [0, 1]$ . Это свойство называется свойством сохранения маргиналов.

Для перемещения точки  $x_0$  согласно обученному векторному полю  $u$  требуется численное интегрирование ОДУ  $dx_t = u_t(x_t)dt$ . Векторные поля с прямолинейными (или почти прямолинейными) траекториями дают меньшую ошибку временной дискретизации и повышают вычислительную эффективность, что важно для практических применений.

Обычно векторное поле  $u$  параметризуется U-Net образными нейронными сетями. Функция потерь оптимизируется стохастическими методами. Для интеграции ОДУ можно использовать методы численного интегрирования, в частности, методы Рунге-Кутты.

**Недостатки.** FM унаследовал основной недостаток диффузионных моделей: его решения обладают криволинейные траектории, что приводит к трудоемкому интегрированию ОДУ при сэмплировании и большому количеству вычислений значения функции векторного поля. Исследователи заметили, что некоторые начальные планы  $\pi$  могут давать более прямые траектории после FM по сравнению со стандартным независимым планом  $p_0 \times p_1$ . Две наиболее популярные модификации - Optimal Transport Conditional Flow Matching [7] и Rectified Flow [6, 22, 5].

## Action Matching

В работе [23] авторы предлагают новый метод Action Matching для изучения широкого класса преобразований распределений, используя только независимые сэмплы из временной эволюции системы. Этот подход предоставляет удобную целевую функцию потерь, которая избегает явных предположений о базовой динамике и исключает необходимость backpropagation через дифференциальные уравнения или солверы оптимального транспорта.

Опираясь на связи с оптимальным транспортом, Action Matching может быть расширен на стохастические дифференциальные уравнения и динамики с созданием/уничтожением вероятностной массы.

В отличие от FM, где эволюция системы задается вручную заранее через линейный интерполянт, данный подход изучает произвольную установившуюся динамику системы между двумя распределениями через выборки из промежуточных распределений.

## Optimal Transport Conditional Flow Matching (OT-CFM)

Первое направление исследований [7] посвящено включению свойств решений задачи ОТ в FM. Если в качестве начального плана для FM использовать динамический ОТ-план  $\pi^*$ , то решением будет являться векторное поле  $u^*$ , которое порождает прямые траектории. Однако, как правило, истинный ОТ-план  $\pi^*$  неизвестен. В таком случае, для достижения некоторой степени прямолинейности обучаемых траекторий, естественной идеей является выбор начального плана  $\pi$ , близкого к оптимальному  $\pi^*$ . Вдохновленные этим, авторы OT-CFM используют преимущества приближенного ОТ-плана на мини-батчах и достигают значительно

лучших практических результатов по сравнению со стандартным FM. Сначала они независимо выбирают батчи из распределений  $p_0$  и  $p_1$ . Затем объединяют эти батчи в соответствии с дискретным OT-планом между ними. Полученный объединенный батч используется в FM. Дискретная задача OT является выпуклой оптимизационной задачей, которая возникла гораздо раньше, чем её непрерывный аналог. Её можно эффективно решить: современные алгоритмы требуют  $O(b^3)$  операций, где  $b$  — размер батча.

**Недостатки.** Главный недостаток OT-CFM заключается в том, что он восстанавливает лишь решение с непрямыми траекториями. Более того, этот подход был предложен как эвристика без должного теоретического обоснования. Для сходимости к истинному оптимальному транспортному плану размер батча должен быть достаточно большим, однако с увеличением размера батча вычислительное время также резко возрастает. На практике размеры батчей, обеспечивающие хорошее приближение решения непрерывного OT, оказываются невычислимыми.

## Rectified Flow (RF)

Другое направление исследований [6, 5, 22] основано на свойстве выпрямления траекторий при минимизации FM, которое ещё авторы оригинальной работы отметили как интересный феномен. Авторы [6] первыми развили эту идею до полностью работоспособного метода. Они предложили итеративный подход для улучшения плана  $\pi$ , постепенно выпрямляя траектории с каждой итерацией. Эта идея итеративного уточнения оказала значительное влияние из-за своей эффективности. Например, последние модели Stable Diffusion используют её для уменьшения времени сэмплирования.

**RF.** Можно итеративно применять Flow Matching (FM) к начальному транспортному плану (например, независимому плану), постепенно его корректируя. А именно, алгоритм Rectified Flow на  $K$ -й итерации имеет вид:

$$\phi^{K+1} = \text{FM}(\pi^K), \quad \pi^{K+1} = [\text{id}, \phi^{K+1}] \# p_0, \quad (4.7)$$

где  $\phi^K, \pi^K$  обозначают отображение потока и транспортный план на  $K$ -й итерации соответственно. Траектории  $\{\{z_t\}_{t \in [0,1]}\}^K$ , полученные после  $K$  итераций Rectified Flow, становятся теоретически всё более прямыми, то есть ошибка аппроксимации  $z_t^K \approx (1-t)z_0^K + tz_1^K, \forall t \in [0, 1]$  уменьшается с ростом  $K$ . Авторы также утверждают, что для любой выпуклой функции стоимости  $c$  отображение потока  $\phi_1^\pi$ , полученное из Flow Matching, даёт транспортную стоимость не выше, чем начальный транспортный план  $\pi$ :

$$\int \mathbb{R}^D c(x_0, \phi_1^\pi(x_0)) p_0(x_0) dx_0 \leq \int_{\mathbb{R}^D \times \mathbb{R}^D} c(x_0, x_1) \pi(x_0, x_1) dx_0 dx_1.$$

Интуитивно, транспортные стоимости гарантированно уменьшаются, потому что траектории FM, как решения ОДУ, не пересекаются, даже если начальные прямые, соединяющие  $x_0$  и  $x_1$ , могут пересекаться.

**c-RF.** С каждой итерацией RF (4.7) транспортные стоимости для всех выпуклых функций стоимости не возрастают, но для заданной функции стоимости сходимость к соответствующему OT-плану (транспортировке с минимальными затратами относительно  $c$ ) не гарантируется. В [5] авторы решают эту проблему и для любой заданной выпуклой функции стоимости  $c$  модифицируют RF так, чтобы обеспечить сходимость к OT решению для  $c$ . В этой модификации, называемой  $c$ -Rectified Flow ( $c$ -RF), авторы немного изменяют целевую функцию FM и ограничивают область оптимизации только потенциальными векторными полями вида  $u_t(\cdot) = \nabla \bar{c}(\nabla f_t(\cdot))$ , где  $f_t(\cdot) : \mathbb{R}^D \rightarrow \mathbb{R}$  — произвольная зависящая от времени скалярная функция, а  $\bar{c}$  — выпуклая сопряжённая функция к  $c$ . Для получения динамическо-

го ОТ-решения (квадратичная функция стоимости) функция потерь должна оставаться неизменной, а векторное поле  $u_t$  должно быть простым градиентом  $\nabla f_t(\cdot)$  скалярной функции  $f_t$ .

**PeRFlow.** В [22] авторы предлагают piecewise Rectified Flow (PeRFlow), который делит траектории потока на несколько временных интервалов и выполняет FM в каждом из них. Решая ОДУ в укороченных временных интервалах, PeRFlow избегает моделирования всей траектории ОДУ для сэмплирования данных. Это значительно сокращает время сэмплирования, позволяя проводить интегрирование в реальном времени параллельно с обучением.

**Недостатки.** На практике RF с каждой итерацией накапливает ошибку, вызванную неточностью предыдущих итераций. Эта проблема упоминается в [5, §6, пункт 3]. Из-за нейросетевых аппроксимаций невозможно получить точное решение FM (например,  $\phi_1^K \# p_0 \neq p_1$ ), и эта неточность только растёт с итерациями. Кроме того, обучение (*c*-)RF перестаёт быть свободным от интегрирования ОДУ после первой итерации, так как для вычисления плана  $\pi^{K+1} = [\text{id}, \phi^{K+1}] \# p_0$  необходимо интегрировать ОДУ. Это резко увеличивает время обучения по сравнению с исходным FM. PeRFlow ослабляет этот эффект, но он не исчезает полностью, так как основа метода остаётся неизменной. Авторы демонстрируют в экспериментах, что RF может не справиться с точным воспроизведением целевого распределения. Хотя в теории количество итераций RF должно стремиться к бесконечности для получения прямых траекторий, на практике прямолинейность траекторий и полученное векторное поле перестают изменяться уже после 2 – 3 итераций.

## Заключение

Метод Flow Matching [4] оказал значительное влияние на область генеративного моделирования, предоставив надежный и простой способ преобразования произвольных распределений друг в друга. Однако задача получения прямолинейных траекторий по-прежнему остается актуальной. Ряд исследований посвящен модификациям метода для спрямления траекторий.

Направление Rectified Flow [5, 6, 22] использует теоретически обоснованный эффект выпрямления траекторий в FM. Однако из-за итеративной природы на практике RF требует значительных вычислительных ресурсов и накапливает ошибку, что может приводить к неточному воспроизведению целевого распределения. Другое направление, OT-CFM [7], использует векторные поля оптимального транспорта, которые обычно являются прямолинейными. Предложенные эвристики улучшают траектории, но не обеспечивают их полной прямолинейности ни в теории, ни на практике.

Проблема получения прямолинейных траекторий остается открытой и востребованной в приложениях. Существующие модификации имеют фундаментальные теоретические ограничения такие как необходимость последовательных минимизаций в RF или неточное мини-батч решение в OT-CFM. Таким образом, требуется новое решение, основанное на иных принципах.

Стоит отметить, что даже стандартный Rectified Flow на практике демонстрирует хорошие и надежные результаты. Однако прогресс не стоит на месте, как и требования к генерации. Фундаментальные ограничения текущих методов могут стать непреодолимыми при достижении определенного уровня точности. Поэтому наличие альтернативных решений без таких ограничений в будущем может оказаться крайне полезным.

# Глава 5

## Новый подход к генеративному моделированию

В этой главе я подробно представляю свой новый метод под названием Optimal Flow Matching (OFM), разработанный для устранения недостатков существующих методов на основе FM. Я привожу теоретическое обоснование предложенного алгоритма, детали практической реализации и сравнение с предыдущими подходами.

### 5.1 Теория: Вывод функции потерь

**Оптимальные векторные поля.** Векторное поле  $u^\Psi$  называется оптимальным если оно порождает прямые траектории  $\{\{z_t\}_{t \in [0,1]}\}$ , такие что существует выпуклая функция  $\Psi : \mathbb{R}^D \rightarrow \mathbb{R}$ , что для любой траектории  $\{z_t\}_{t \in [0,1]}$  начальная точка  $z_0$  переходит в конечную  $z_1 = \nabla\Psi(z_0)$ :

$$z_t = (1-t)z_0 + t\nabla\Psi(z_0), \quad t \in [0, 1].$$

Функция  $\Psi$  определяет ОДУ

$$dz_t = (\nabla\Psi(z_0) - z_0)dt, \quad z_t|_{t=0} = z_0. \quad (5.1)$$

Уравнение (5.1) не даёт явной формулы для  $u^\Psi$ , так как оно зависит  $z_0$ . Явная формула строится следующим образом: для момента времени  $t \in [0, 1]$  и точки  $x_t$  ищется траектория  $\{z_t\}_{t \in [0,1]}$  такая что

$$x_t = z_t = (1-t)z_0 + t\nabla\Psi(z_0). \quad (5.2)$$

**Функция потерь.** Optimal Flow Matching (OFM) строится следующим образом: Область оптимизации функции потерь FM с фиксированным планом  $\pi$  ограничивается только оптимальными векторными полями. Функция потерь для OFM выводятся путем подстановки выражения для векторного поля  $u_\Psi$  в функцию потерь FM:

$$\begin{aligned} \mathcal{L}_{OFT}^\pi(\Psi) &:= \mathcal{L}_{FM}^\pi(u^\Psi) = \int_0^1 \left\{ \int_{\mathbb{R}^D \times \mathbb{R}^D} \|u_t^\Psi(x_t) - (x_1 - x_0)\|^2 \pi(x_0, x_1) dx_0 dx_1 \right\} dt, \quad (5.3) \\ x_t &= (1-t)x_0 + tx_1. \end{aligned}$$

Теорема 5.1 утверждает, что OFM решает задачу динамического ОТ за одну процедуру минимизации для любого начального плана  $\pi$ .

**Теорема 5.1 (Связь OFM и ОТ)** Рассмотрим два вероятностных распределения  $p_0, p_1 \in \mathcal{P}_{ac,2}(\mathbb{R}^D)$  и любой транспортный план  $\pi \in \Pi(p_0, p_1)$  между ними. Тогда функция потерь динамического OT  $\mathcal{L}_{OT}$  и функция потерь Optimal Flow Matching  $\mathcal{L}_{OFT}^\pi$  имеют одни и те

*же минимумы, т.е.,*

$$\arg \min_{\text{выпуклая } \Psi} \mathcal{L}_{OFM}^\pi(\Psi) = \arg \min_{\text{выпуклая } \Psi} \mathcal{L}_{OT}(\Psi).$$

Для любой выпуклой функции  $\Psi$ , верно следующее равенство

$$\underbrace{\mathbb{E}_{x_0, x_1 \sim \pi} \int_0^1 \|u_t^\Psi(x_t) - (x_1 - x_0)\|^2 dt}_{=\mathcal{L}_{OFM}^\pi(\Psi)} = 2 \cdot \underbrace{\mathbb{E}_{x_0, x_1 \sim \pi} [\Psi(x_0) + \bar{\Psi}(x_1)]}_{=\mathcal{L}_{OT}(\Psi)} + \text{Const}'(\pi), \quad (5.4)$$

где  $\text{Const}'(\pi) := -2\mathbb{E}_{x_0, x_1 \sim \pi} [\langle x_0, x_1 \rangle]$  не зависит от  $\Psi$ .

Основной технический результат, используемый для доказательства свойств OFM, представлен в Лемме 4.

**Лемма 1 (Главная Лемма об интегрировании)** Для любых двух точек  $x_0, x_1 \in \mathbb{R}^D$  и выпуклой функции  $\Psi$  верно следующее равенство

$$\int_0^1 \|u_t^\Psi(x_t) - (x_1 - x_0)\|^2 dt = 2 \cdot [\Psi(x_0) + \bar{\Psi}(x_1) - \langle x_0, x_1 \rangle]. \quad (5.5)$$

Доказательство возможности аналитического вычисления интеграла по времени в случае линейной интерполяции для произвольных  $x_0, x_1$  и  $\Psi$  требует интегрирования вдоль параметризованных кривых, использования свойств выпуклого сопряжения и применения нетривиальной замены переменных.

## 5.2 Свойства OFM

В данном подразделе представлены теоретические свойства OFM, которые дают понимание об его основных принципах работы.

**Предположение 1 (Упрощенная функция потерь OFM)** Функция потерь (5.3) может быть упрощена до более удобной формы:

$$\mathcal{L}_{OFM}^\pi(\Psi) = \int_0^1 \left\{ \int_{\mathbb{R}^D \times \mathbb{R}^D} \left\| \frac{(\phi_t^\Psi)^{-1}(x_t) - x_0}{t} \right\|^2 \pi(x_0, x_1) dx_0 dx_1 \right\} dt, \quad x_t = (1-t)x_0 + tx_1. \quad (5.6)$$

Упрощенная форма (5.6) показывает, что функция потерь OFM фактически измеряет, насколько хорошо  $\Psi$  восстанавливает начальные точки  $x_0$  линейных интерполянтов зависимости от будущей точки  $x_t$  и времени  $t$ .

**Генеративные свойства OFM.** Основная цель OFM - построить векторное поле  $u$ , максимально близкое к динамическому ОТ-полю  $u^*$ . Для измерения расстояния между ними можно использовать метод наименьших квадратов:

$$\text{dist}(u, u^*) := \int_0^1 \int_{\mathbb{R}^D} |u_t(x_t) - u_t^*(x_t)|^2 \underbrace{\phi_t^* \# p_0(x_t)}_{:= p_t^*(x_t)} dx_t dt. \quad (5.7)$$

**Предположение 2 (Невычислимое расстояние)** *Расстояние  $\text{dist}(u, u^*)$  между произвольным векторным полем  $u$  и ОТ-полем  $u^*$  равно функции потерь FM из (4.7) с оптимальным планом  $\pi^*$ , т.е.*

$$\text{dist}(u, u^*) = \mathcal{L}_{FM}^{\pi^*}(u) - \underbrace{\mathcal{L}_{FM}^{\pi^*}(u^*)}_{=0}.$$

Невычислимое расстояние  $\text{dist}(u, u^*)$  невозможно минимизировать, поскольку оптимальный план  $\pi^*$  неизвестен. В OT-CFM [9] авторы эвристически аппроксимируют  $\pi^*$  в  $\mathcal{L}_{FM}^{\pi^*}(u)$ , но получают смещенное решение. Удивительно, но для *оптимальных* векторных полей это расстояние может быть вычислено явно через **любой** известный план  $\pi$ .

**Предположение 3 (Вычислимое расстояние для OFM)** *Расстояние  $\text{dist}(u^\Psi, u^{\Psi^*})$  между **оптимальным** векторным полем  $u^\Psi$ , порожденным выпуклой функцией  $\Psi$ , и векторным полем  $u^{\Psi^*}$  с потенциалом Бренье  $\Psi^*$  может быть непосредственно вычислено через функцию потерь OFM (5.3) и **любой** план  $\pi$ :*

$$\text{dist}(u^\Psi, u^{\Psi^*}) = \mathcal{L}_{FM}^\pi(u^\Psi) - \mathcal{L}_{FM}^\pi(u^{\Psi^*}) = \mathcal{L}_{OFM}^\pi(\Psi) - \mathcal{L}_{OFM}^\pi(\Psi^*). \quad (5.8)$$

В (5.8) первое слагаемое представляет собой вычислимую функцию потерь OFM, а второе слагаемое не зависит от  $\Psi$ . Следовательно, в процессе минимизации в OFM расстояние (5.7) между текущим векторным полем и динамическим ОТ-полем постепенно уменьшается вплоть до полного совпадения.

**Вассерштейн приближение OFM.** В [21, Теорема A.3] авторы демонстрируют, что минимизация функции потерь  $L_{OT}(\Psi)$  естественным образом приводит к минимизации расстояния Вассерштейна между распределением  $p_1$  и распределением  $\nabla\Psi\#p_0$ . Как прямое следствие Теоремы 5.1, данное генеративное свойство сохраняется и для нашего функционала  $\mathcal{L}_{OFM}^\pi(\Psi)$ .

**Лемма 2 (Вассерштейн приближение OFM)** *Пусть выпуклая функция  $\Psi$  достигает точности  $\varepsilon$  в функции потерь OFM, то есть:*

$$\mathcal{L}_{OFM}^\pi(\Psi) - \mathcal{L}_{OFM}^\pi(\Psi^*) \leq \varepsilon \quad (5.9)$$

*Если при этом  $\Psi$  является  $L$ -гладкой [24], тогда отображение  $\phi_{\Psi,1}$  трансформирует  $p_0$  в  $\varepsilon$ -близкое к  $p_1$  распределение:*

$$\mathbb{W}_2^2(\phi_1^\Psi\#p_0, p_1) \leq L \cdot \varepsilon \quad (5.10)$$

## 5.3 Детали реализации на практике

В этом подразделе объясняются детали оптимизации функции потерь Optimal Flow Matching (5.3) на практике.

**Параметризация  $\Psi$ .** На практике класс выпуклых функций параметризуется с помощью Input Convex Neural Networks (ICNNs) [10]  $\Psi_\theta$  с параметрами  $\theta$ . Это нейронные сети со одномерным выходом, спроектированные таким образом, чтобы сеть была выпуклой по своему входу. Они состоят из полно связных или сверточных блоков, некоторые веса которых сделаны неотрицательными для сохранения выпуклости. Кроме того, функции активации должны быть неубывающими и выпуклыми по каждому входному направлению. Эти сети поддерживают большинство популярных методов обучения (например, оптимизацию градиентным спуском, dropout, skip-connection и т.д.).

**Вычисление градиента функции потерь OFM.** Для градиента функции потерь OFM (5.3) существует явная формула, удобная для реализации в современных библиотеках глубокого обучения.

**Теорема 5.2 (Явная формула для вычисления градиента)** Градиент  $\mathcal{L}_{OFM}^\pi$  может быть вычислен как

$$\frac{d\mathcal{L}_{OFM}^\pi}{d\theta} := \frac{d}{d\theta} \mathbb{E}_{t;x_0,x_1 \sim \pi} \left\langle \text{NO-GRAD} \left\{ 2 \left( t \nabla^2 \Psi_\theta(z_0) + (1-t)I \right)^{-1} \frac{(x_0 - z_0)}{t} \right\}, \nabla \Psi_\theta(z_0) \right\rangle,$$

где переменные под NO-GRAD и  $z_0$  предполагаются константами при дифференцировании.

**Решение выпуклой подзадачи.** Нахождение начальной точки  $z_0$  можно сформулировать как задачу минимизации:

$$\begin{aligned} x_t &= (1-t)z_0 + t\nabla\Psi(z_0), \\ 0 &= \nabla \left( \frac{(1-t)}{2} \|\cdot\|^2 + t\Psi(\cdot) - \langle x_t, \cdot \rangle \right)(z_0), \\ z_0 &= \arg \min_{z_0 \in \mathbb{R}^D} \left[ \frac{(1-t)}{2} \|z_0\|^2 + t\Psi(z_0) - \langle x_t, z_0 \rangle \right]. \end{aligned} \quad (5.11)$$

Задача оптимизации (5.11) является как минимум **(1-t)-сильно выпуклой** и может быть эффективно решена для любой заданной точки  $x_t$  (в отличие от стандартных невыпуклых задач оптимизации).

**Алгоритм.** Псевдокод метода Optimal Flow Matching представлен в листинге 1. Математическое ожидание по плану  $\pi$  и времени  $t$  с равномерным распределением на  $[0, 1]$  оценивается методом Монте-Карло.

---

### Algorithm 1 Optimal Flow Matching

---

**Input:** Начальный транспортный план  $\pi \in \Pi(p_0, p_1)$ , количество итераций  $K$ , размер батча  $B$ , оптимизатор  $Opt$ , оптимизатор для подзадачи  $SubOpt$ , ICNN  $\Psi_\theta$

- 1: **for**  $k = 0, \dots, K-1$  **do**
- 2:   Семплирует батч  $\{(x_0^i, x_1^i)\}_{i=1}^B$  размера  $B$  из плана  $\pi$ ;
- 3:   Семплирует батч  $\{t^i\}_{i=1}^B$  размера  $B$  из  $U[0, 1]$ ;
- 4:   Вычисляет линейный интерполянт  $x_{t^i}^i = (1-t^i)x_0^i + t^i x_1^i$  для всех  $i \in \overline{1, B}$ ;
- 5:   Находит начальные точки  $z_0^i$  через решение выпуклой задачи посредством  $SubOpt$ :

$$z_0^i = \text{NO-GRAD} \left\{ \arg \min_{z_0^i} \left[ \frac{(1-t^i)}{2} \|z_0^i\|^2 + t^i \Psi_\theta(z_0^i) - \langle x_{t^i}^i, z_0^i \rangle \right] \right\};$$

- 6:   Вычисляет лосс  $\hat{\mathcal{L}}_{OFM}$
  - 7:    $\hat{\mathcal{L}}_{OFM} = \frac{1}{B} \sum_{i=1}^B \left\langle \text{NO-GRAD} \left\{ 2 \left( t^i \nabla^2 \Psi_\theta(z_0^i) + (1-t^i)I \right)^{-1} \frac{(x_0^i - z_0^i)}{t^i} \right\}, \nabla \Psi_\theta(z_0^i) \right\rangle;$
  - 8:   Обновляет параметры  $\theta$  посредством шага оптимизатора  $Opt$  с градиентами  $\frac{d\hat{\mathcal{L}}_{OFM}}{d\theta}$ ;
  - 9: **end for**
-

# Глава 6

## Численные эксперименты

### 6.1 Основные эксперименты

В данной главе представлена оценка эффективности предложенного Optimal Flow Matching на практике. Сначала рассматривается демонстрационный двумерный случай (§6.1), наглядно иллюстрирующий ключевые характеристики метода. Затем проводится тестирование на бенчмарке Wasserstein-2 [21] (§6.1). Метод также применяется для решения задачи высокоразмерного перевода изображений из двух разных распределений друг в друга в латентном пространстве предобученного автоэнкодера ALAE (§6.1).

Исходный код реализации OFM и всех проведенных экспериментов доступен в репозитории: <https://github.com/Jhomanik/Optimal-Flow-Matching>.

### 2D Иллюстрации

В данном подразделе я демонстрирую proof-of-concept предложенного OFM на двумерной задаче и показываю, что решения OFM не зависят от начального транспортного плана  $\pi$ . Алгоритм 1 запускается между стандартным гауссовым распределением  $p_0 = \mathcal{N}(0, I)$  и смесью восьми гауссовых распределений  $p_1$ , показанной на Рисунке 6.2a. Рассматриваются три различных стохастических плана  $\pi$  (Рисунок 6.2b): независимый план  $p_0 \times p_1$  (idp), минибатчевый (mb) и антиминибатчевый (anti-mb) (Рисунки 6.2c, 6.2d) дискретный оптимальный транспорт (с квадратичной стоимостью) с размером батча  $B_{mb} = 64$ . В антиминибатчевом случае пары исходных и целевых точек составляются путем решения задачи дискретного оптимального транспорта с отрицательной квадратичной стоимостью  $-\|x - y\|_2^2$ . Полученные OFM траектории представлены на Рисунке 6.2. Можно эмпирически наблюдать, что OFM находит одинаковое решение для всех начальных планов  $\pi$ . Эти планы также применяются к оригинальному FM (4.7) на Рисунке 6.1. В сравнении с OFM, результирующие траектории, полученные FM, существенно зависят от плана.

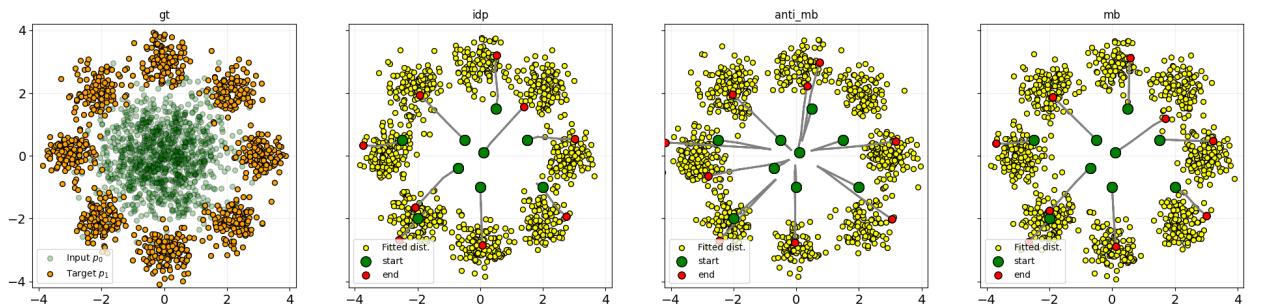


Рис. 6.1: Результаты Flow Matching на Гауссиана $\rightarrow$ 8 Гауссиан.

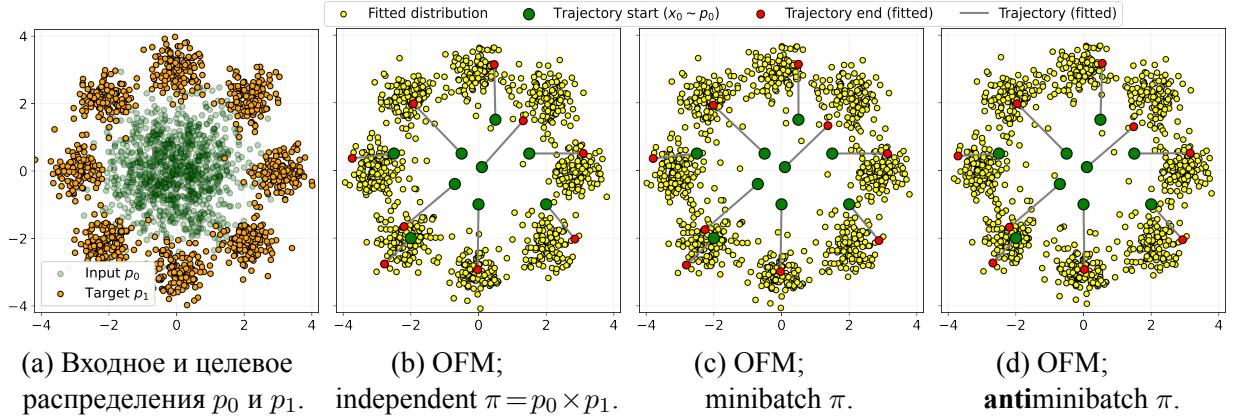


Рис. 6.2: Результаты Optimal Flow Matching на Гауссиана→8 Гауссиан.

## OT Бенчмарк

В данном подразделе проводится сравнение OFM с другими методами в способности решать задачу оптимального транспорта (OT). Методы OFM, FM-based и OT-решатели тестируются на бенчмарке OT Benchmark [21]. Авторы бенчмарка предоставляют высокоразмерные непрерывные распределения  $p_0, p_1$ , для которых истинное OT-отображение  $T^*$  для квадратичной стоимости известно по построению.

**Метрики.** Следуя авторам бенчмарка [21], для оценки качества восстановленного транспортного отображения  $T$  между  $p_0$  и  $p_1$  используется *процент необъясненной дисперсии* (UVP):  $\mathcal{L}^2\text{-UVP}(T) := 100 \cdot \|T - T^*\|_{\mathcal{L}^2(p_0)}^2 / \text{Var}(p_1)\%$ . Значения  $\mathcal{L}^2\text{-UVP}(T) \approx 0\%$  означают, что  $T$  хорошо аппроксимирует  $T^*$ , тогда как значения  $\geq 100\%$  свидетельствуют о значительном отклонении от оптимальности. Также вычисляется *косинусная схожесть* между истинными направлениями  $T^* - \text{id}$  и полученными направлениями  $T - \text{id}$ :

$$\cos(T - \text{id}, T^* - \text{id}) = \frac{\langle T - \text{id}, T^* - \text{id} \rangle_{\mathcal{L}^2(p_0)}}{\|T - \text{id}\|_{\mathcal{L}^2(p_0)} \cdot \|T^* - \text{id}\|_{\mathcal{L}^2(p_0)}} \in [-1, 1].$$

Для хороших аппроксимаций метрика косинуса стремится к 1. Оценка метрик  $\mathcal{L}^2\text{-UVP}$  и  $\cos$  проводится на выборках размера  $2^{14}$  из  $p_0$ .

**Соперники.** В экспериментах сравниваются: Conditional Flow Matching (OT-CFM), Rectified Flow (RF), *c*-Rectified Flow (*c*-RF), OT-солвер MMv-1 [13] и его амортизированная версия из [20]. В работах [13] и [20] авторы непосредственно минимизируют двойственную формулировку  $\mathcal{L}_{OT}$  (4.5), параметризуя  $\Psi$  с помощью ICNN и вычисляя  $\bar{\Psi}(x_1)$  через решение выпуклой оптимизационной подзадачи, что аналогично нашей инверсии (5.11). В [20] дополнительно представлены результаты с параметризацией  $\Psi$  MLP-сетями. В соответствии с [21], приводятся также результаты для линейного OT-отображения (бейзлайн), которое преобразует средние значения и дисперсии распределений. Для OFM рассматриваются два начальных плана: независимый план (Ind) и минибатчевый OT (MB) с размером батча  $B_{mb} = 64$ .

**Результаты.** Основные результаты представлены в Таблице 6.1.

Солвер	Тип солвера	$D = 2$	$D = 4$	$D = 8$	$D = 16$	$D = 32$	$D = 64$	$D = 128$	$D = 256$
MMv1* [13]		0.2	1.0	1.8	1.4	6.9	8.1	2.2	2.6
Amortization, ICNN** [20]	Dual OT solver	0.26	0.78	1.6	1.1	1.9	4.2	1.6	2.0
Amortization, MLP** [20]		0.03	0.22	0.6	0.8	2.0	2.1	0.67	0.59
Linear* [21]	Baseline	14.1	14.9	27.3	41.6	55.3	63.9	63.6	67.4
OT-CFM [9]		0.16	0.73	2.27	4.33	7.9	11.4	12.1	27.5
RF [6]		8.58	49.46	51.25	63.33	63.52	85.13	84.49	83.13
<i>c</i> -RF [5]	Flow Matching	1.56	13.11	17.87	35.39	48.46	66.52	68.08	76.48
OFM Ind		0.19	0.61	1.4	1.1	1.47	8.35	1.96	3.96
OFM MB		<b>0.15</b>	<b>0.52</b>	<b>1.2</b>	<b>1.0</b>	<b>1.2</b>	<b>7.2</b>	<b>1.5</b>	<b>2.9</b>

Таблица 6.1:  $\mathcal{L}^2\text{-UVP}$  значения солверов на размерностях  $D = 2, 4, 8, 16, 32, 64, 128, 256$ .

Лучшие метрики среди *Flow Matching* методов выделены **жирным**. \* Метрики взяты из [21]. \*\* Метрики взяты из [20].

Результаты для косинусной метрики  $\cos$  представлены в Таблице 6.2.

Солвер	Тип солвера	$D = 2$	$D = 4$	$D = 8$	$D = 16$	$D = 32$	$D = 64$	$D = 128$	$D = 256$
MMv1* [13]	Dual OT solver	0.99	0.99	0.99	0.99	0.98	0.99	0.99	0.99
Linear*	Baseline	0.75	0.80	0.73	0.73	0.76	0.75	0.77	0.77
OT-CFM MB [9]		0.999	0.985	0.978	0.968	0.975	0.96	0.949	0.915
RF [6]		0.87	0.75	0.65	0.67	0.72	0.70	0.70	0.70
c-RF [5]	Flow Matching	0.989	0.83	0.83	0.78	0.778	0.762	0.748	0.73
OFM Ind		0.999	0.993	0.993	0.993	0.999	0.966	0.992	0.981
OFM MB		<b>0.999</b>	<b>0.994</b>	<b>0.995</b>	<b>0.994</b>	<b>0.999</b>	<b>0.970</b>	<b>0.994</b>	<b>0.986</b>

Таблица 6.2:  $\cos$  значения солверов на размерностях  $D = 2, 4, 8, 16, 32, 64, 128, 256$ .

Лучшие метрики среди *Flow Matching* методов выделены **жирным**. \* Метрики взяты из [21]. \*\* Метрики взяты из [20].

**Обсуждение результатов.** Среди методов на основе Flow Matching, OFM демонстрирует наилучшие результаты независимо от выбранного плана. Для всех рассмотренных планов OFM сходится к близким конечным решениям с сопоставимыми метриками. Минибатчевый план показывает несколько лучшие результаты, особенно в задачах высокой размерности. Теоретически, результаты OFM должны быть одинаковыми для любого плана  $\pi$ , однако при стохастической оптимизации планы с высокой дисперсией приводят к сходимости к менее точным решениям.

ОТ-солвер на основе MLP, как правило, превосходит наш OFM, поскольку MLP не имеют практических ограничений, свойственных ICNN. Однако использование MLP является эмпирическим приемом без строгого теоретического обоснования. При замене ICNN на MLP в OFM метод теряет сходимость.

Rectified Flow (RF) показывает результаты хуже даже линейного базового метода, что ожидаемо, так как он не предназначен для решения задачи  $\mathbb{W}_2$  оптимального транспорта. Модификация c-RF работает лучше, но ее качество быстро ухудшается с ростом размерности. OT-CFM демонстрирует наилучшие результаты среди базовых методов на основе Flow Matching, но в задачах высокой размерности уступает OFM.

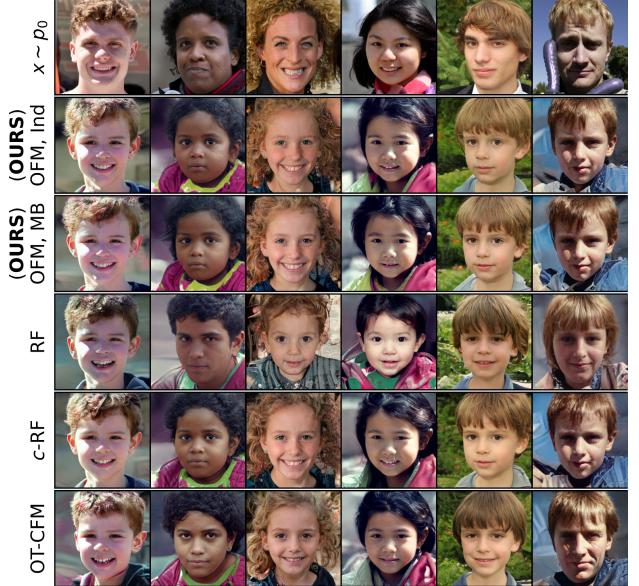


Рис. 6.3: Непарный перевод *Adult*  $\rightarrow$  *Child* в латентном пространстве ALAE  $1024 \times 1024$  FFHQ.

## Непарный перевод изображений

Еще одной задачей, связанной с преобразованием распределений, является непарный перевод изображений [25]. В работе используется подход из [26], где преобразование вычисляется в 512-мерном латентном пространстве предобученного автоэнкодера ALAE [27] на наборе данных FFHQ  $1024 \times 1024$  [28]. В частности, обучающие выборки FFHQ (60K лиц) разделяются на подмножества *children* (дети) и *adults* (взрослые), а соответствующие латентные коды ALAE рассматриваются как исходное ( $p_0$ ) и целевое ( $p_1$ ) распределения.

На этапе инференса берется новое (не встречавшееся при обучении) изображение взрослого из тестовой выборки FFHQ, извлекается его латентный код, который затем обрабатывается обученной моделью и декодируется обратно в пространство изображений. Размер батча для методов с минибатчевым оптимальным транспортом ( $[OFM, MB]$ ,  $[OT-CFM]$ ) составляет  $B_{mb} = 128$ . Качественные результаты и метрика FID [29] представлены на Рисунке 6.3 и в Таблице 6.3 соответственно.

Method	OFM, Ind Fitted	OFM, MB Fitted	RF	$c$ -RF	OT-CFM
FID	11.8	<b>11.0</b>	21.0	13.5	12.9

Таблица 6.3: FID метрика на задаче Adult→Child.

**Обсуждение.** OFM сходится к практически одинаковым решениям как для независимого плана, так и для МВ плана, генерируя качественно правдоподобные изображения. Наиболее близкие к OFM результаты показывает метод [ $c$ -RF]. Аналогично OFM, этот метод (в пределе шагов RF) также восстанавливает квадратичное ОТ-отображение.

**Технические детали.** Для проведения экспериментов по высокоразмерному преобразованию изображений с использованием предобученного автоэнкодера ALAE был адаптирован публично доступный код:

<https://github.com/SKholkin/LightSB-Matching>.

Дополнительные результаты для OFM представлены на Рисунке 6.4.



Рис. 6.4: Непарный перевод *Adult*→*Child* в латентном пространстве ALAE  $1024 \times 1024$  FFHQ.  
Дополнительные результаты.

## Амортизация

Для обучения OFM требуется эффективное решение подзадачи (5.11). В качестве более продвинутой альтернативы LBFGS рассматривается техника амортизации из работы [20]. А именно, приближенное решение (5.11) в точке  $x_t$  для момента времени  $t$  можно найти с помощью дополнительной MLP-сети  $A_\phi(\cdot, \cdot) : \mathbb{R}^D \times [0, 1] \rightarrow \mathbb{R}^D$ :

$$A_\phi(x_t, t) \approx \arg \min_{z_0 \in \mathbb{R}^D} \left[ \frac{(1-t)}{2} |z_0|^2 + t \Psi(z_0) - \langle x_t, z_0 \rangle \right], \quad (6.1)$$

после чего запускать солвер подзадачи (LBFGS), инициализированный значением  $A_\phi(x_t, t)$ , до достижения сходимости. В модифицированном пайплайне обучения параметры  $\phi$  сети  $A_\phi$  включаются в процесс оптимизации, как показано в Алгоритме 2.

---

**Algorithm 2** Optimal Flow Matching

---

**Input:** Начальный транспортный план  $\pi \in \Pi(p_0, p_1)$ , количество итераций  $K$ , размер батча  $B$ , оптимизатор  $Opt$ , оптимизатор для подзадачи  $SubOpt$ , ICNN  $\Psi_\theta$ , солвер для амортизации  $AmorOpt$ , MLP  $A_\phi$

- 1: **for**  $k = 0, \dots, K - 1$  **do**
- 2:   Семплирует батч  $\{(x_0^i, x_1^i)\}_{i=1}^B$  размера  $B$  из плана  $\pi$ ;
- 3:   Семплирует батч  $\{t^i\}_{i=1}^B$  размера  $B$  из  $U[0, 1]$ ;
- 4:   Вычисляет линейный интерполянт  $x_{t^i}^i = (1 - t^i)x_0^i + t^i x_1^i$  для всех  $i \in \overline{1, B}$ ;
- 5:   Вычисляет инициализацию  $z_{init}^i = A_\phi(x_{t^i}^i, t^i)$  для всех  $i \in \overline{1, B}$ ;
- 6:   Находит  $z_0^i$  в (5.11) посредством  $SubOpt$ , инициализированного  $z_{init}^i$  для всех  $i \in \overline{1, B}$ ;
- 7:   Вычисляет функцию потерь OFM  $\hat{\mathcal{L}}_{OFM}$

$$\hat{\mathcal{L}}_{OFM} = \frac{1}{B} \sum_{i=1}^B \left\langle \text{NO-GRAD} \left\{ 2 \left( t^i \nabla^2 \Psi_\theta(z_0^i) + (1 - t^i) I \right)^{-1} \frac{(x_0^i - z_0^i)}{t^i} \right\}, \nabla \Psi_\theta(z_0^i) \right\rangle;$$

- 8:   Обновляет параметры  $\theta$  через оптимизатор  $Opt$  с градиентом  $\frac{d\hat{\mathcal{L}}_{OFM}}{d\theta}$ ;
- 9:   Вычисляет лосс амортизации  $\mathcal{L}_{Amor}$

$$\mathcal{L}_{Amor} = \frac{1}{B} \sum_{i=1}^B \|z_{init}^i - z_0^i\|^2;$$

- 10:   Обновляет параметры  $\phi$  через оптимизатор  $AmorOpt$  с градиентом  $\frac{d\mathcal{L}_{Amor}}{d\phi}$ ;
  - 11: **end for**
- 

**Обсуждение.** В ходе экспериментов не было выявлено улучшения итоговых метрик по сравнению с базовой версией OFM при одинаковых гиперпараметрах. Однако предложенная модификация потенциально может сократить общее время обучения. В процессе тренировки модель  $A_\phi(\cdot, \cdot)$  постепенно учится предсказывать всё более точные начальные приближения  $z_{init}^i$ , что позволяет сократить количество затратных итераций  $SubOpt$ .

## Вычислительные ресурсы и время

Далее приведены ориентировочные временные затраты на обучение OFM и других FM-методов в различных экспериментах.

В демонстрационном 2D-эксперименте обучение занимает  $\approx 1.5$  часа на одной видеокарте 1080 Ti. Для бенчмарка Wasserstein-2 время вычислений зависит от размерности  $D = 2, 4, \dots, 256$ . Полный цикл экспериментов (для независимого и минибатчевого плана  $\pi$ ) занимает  $\approx 3$  дня на трёх GPU A100. В эксперименте с ALAE обучение длится  $\approx 5$  часов на одной 1080 Ti.

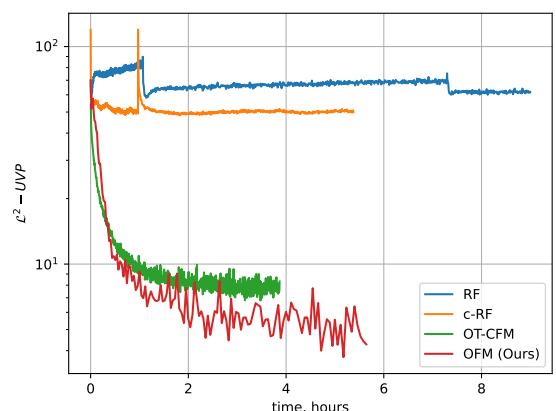


Рис. 6.5: График  $\mathcal{L}^2$ –UVP метрики с течением времени обучения для  $D = 32$ .

Для лучшего понимания динамики методов во времени на рис. 6.5 показана зависимость метрики  $\mathcal{L}^2$ –UVP от времени обучения для размерности  $D = 32$ . Несмотря на высокую вычислительную стоимость одной итерации OFM, метод достигает наилучших результатов за меньшее количество шагов.

# Глава 7

## Обсуждение и заключение

### 7.1 Сравнение с релевантными работами

В данном разделе проводится сравнение Optimal Flow Matching (OFM) с существующими подходами к спрямлению траекторий. Ключевая особенность OFM заключается в том, что метод работает исключительно с потоками, которые по своей конструкции имеют прямолинейные траектории и не требуют интегрирования ОДУ для переноса точек. В отличие от этого, другие методы могут давать искривленные траектории в процессе обучения и все равно требуют решения ОДУ даже при почти прямолинейных путях.

**Солверы задач оптимального транспорта** [13, 14, 20]. Как показано в Теореме 5.1, OFM и двойственные методы решения ОТ по сути минимизируют одну и ту же функцию потерь оптимального транспорта. Однако OFM активно использует временную компоненту динамического процесса, создавая тем самым новую теоретическую связь между оптимальным транспортом и Flow Matching. Такая прямая взаимосвязь открывает возможности для объединения преимуществ обоих методов и их более глубокого понимания.

**Сравнение с OT-CFM подходом** [7]. В отличие от OFM, метод OT-CFM дает неточное решение задачи оптимального транспорта и не гарантирует получения прямолинейных траекторий. В OT-CFM план оптимального транспорта для мини-батчей используется как эвристика для улучшения траекторий на практике. В то же время, OFM теоретически обосновывает использование любого начального транспортного плана  $\pi$  (Теорема 5.1).

**Сравнение с методами Rectified Flow** [6, 5, 22]. В Rectified Flow авторы применяют итеративный процесс Flow Matching для улучшения траекторий. Однако на каждой итерации происходит накопление ошибки из-за нейросетевой аппроксимации. Кроме того, RF не гарантирует сходимости к ОТ-плану для квадратичной функции стоимости. Модификация  $c$ -Rectified Flow [5] может сходиться к ОТ-плану для произвольной функции стоимости  $c$ , но остается итеративной. Оба метода требуют интегрирования ОДУ после первой итерации для продолжения обучения. В отличие от них, OFM рассматривает только квадратичную функцию стоимости, но находит соответствующее ОТ-решение всего за одну итерацию FM без необходимости симуляции траекторий.

### 7.2 Ограничения OFM

Несмотря на преимущества, метод Optimal Flow Matching имеет следующие ограничения:

**Решение выпуклой подзадачи.** В процессе обучения требуется решение сильно выпуклой подзадачи (5.11) для вычисления начальной точки  $z_0$ . На практике для этого можно использовать стандартный градиентный спуск (с оптимизатором LBFGS), однако существуют более эффективные специализированные методы решения задач сопряжения, как в области оптимизации [30, 31], так и в теории оптимального транспорта [20, 14], что открывает широкие возможности для улучшений.

**Ограничения ICNN.** Известно, что выпуклые нейросети (ICNN) могут уступать обычным нейросетям [21, 32], что потенциально ограничивает эффективность OFM. Вместе с

тем, методы улучшения ICNN активно исследуются [33, 34, 35, 36] благодаря их рас- тущей популярности в различных приложениях [37, 38, 39].

**Вычислительная сложность.** Расчет градиента функции потерь OFM (Теорема 5.2) требует обращения матрицы Гессе  $\nabla^2\Psi(\cdot)$ , что является вычислительно сложной операцией. Это представляет собой существенное ограничение для масштабирования метода.

## 7.3 Заключение

Optimal Flow Matching представляет собой новую модификацию метода Flow Matching, которая после одной итерации FM обеспечивает прямолинейные траектории, не требующие решения ОДУ для сэплирования. Более того, метод восстанавливает решение оптимального транспорта для квадратичной функции стоимости при любом начальном плане. Ключевая идея OFM заключается в рассмотрении специальных векторных полей - градиентов выпуклых функций, которые по построению дают прямолинейные траектории.

В отличие от методов OT-CFM (теоретически необоснованная прямота траекторий) и RF (накопление ошибок), OFM не имеет фундаментальных ограничений. Проблемы метода связаны в основном с процедурой оптимизации, а не с его концепцией.

Новые теоретические результаты обладают значительным потенциалом для усовершенствования современных методов на основе Flow Matching и могут стимулировать дальнейшие исследования в этом направлении. Это особенно актуально в свете растущего интереса к методам Flow Matching в современных генеративных моделях [22, 1, 2].

# Литература

- [1] X. Liu, X. Zhang, J. Ma, J. Peng, and qiang liu, “Instaflow: One step is enough for high-quality diffusion-based text-to-image generation,” in *The Twelfth International Conference on Learning Representations*, 2024.
- [2] P. Esser, S. Kulal, A. Blattmann, R. Entezari, J. Müller, H. Saini, Y. Levi, D. Lorenz, A. Sauer, F. Boesel, D. Podell, T. Dockhorn, Z. English, and R. Rombach, “Scaling rectified flow transformers for high-resolution image synthesis,” in *Forty-first International Conference on Machine Learning*, 2024.
- [3] H. Cao, C. Tan, Z. Gao, Y. Xu, G. Chen, P.-A. Heng, and S. Z. Li, “A survey on generative diffusion models,” *IEEE Transactions on Knowledge and Data Engineering*, 2024.
- [4] Y. Lipman, R. T. Q. Chen, H. Ben-Hamu, M. Nickel, and M. Le, “Flow matching for generative modeling,” in *The Eleventh International Conference on Learning Representations*, 2023.
- [5] Q. Liu, “Rectified flow: A marginal preserving approach to optimal transport,” *arXiv preprint arXiv:2209.14577*, 2022.
- [6] X. Liu, C. Gong, and qiang liu, “Flow straight and fast: Learning to generate and transfer data with rectified flow,” in *The Eleventh International Conference on Learning Representations*, 2023.
- [7] A.-A. Pooladian, H. Ben-Hamu, C. Domingo-Enrich, B. Amos, Y. Lipman, and R. T. Chen, “Multisample flow matching: Straightening flows with minibatch couplings,” in *International Conference on Machine Learning*, pp. 28100–28127, PMLR, 2023.
- [8] C. Villani, *Topics in optimal transportation*, vol. 58. American Mathematical Soc., 2021.
- [9] A. Tong, K. FATRAS, N. Malkin, G. Huguet, Y. Zhang, J. Rector-Brooks, G. Wolf, and Y. Bengio, “Improving and generalizing flow-based generative models with minibatch optimal transport,” *Transactions on Machine Learning Research*, 2024. Expert Certification.
- [10] B. Amos, L. Xu, and J. Z. Kolter, “Input convex neural networks,” in *International Conference on Machine Learning*, pp. 146–155, PMLR, 2017.
- [11] J.-D. Benamou and Y. Brenier, “A computational fluid mechanics solution to the monge-kantorovich mass transfer problem,” *Numerische Mathematik*, vol. 84, no. 3, pp. 375–393, 2000.
- [12] A. Genevay, M. Cuturi, G. Peyré, and F. Bach, “Stochastic optimization for large-scale optimal transport,” *Advances in neural information processing systems*, vol. 29, 2016.
- [13] A. Taghvaei and A. Jalali, “2-wasserstein approximation via restricted convex potentials with application to improved training for gans,” *arXiv preprint arXiv:1902.07197*, 2019.

- [14] A. Makkouva, A. Taghvaei, S. Oh, and J. Lee, “Optimal transport mapping via input convex neural networks,” in *International Conference on Machine Learning*, pp. 6672–6681, PMLR, 2020.
- [15] M. Daniels, T. Maunu, and P. Hand, “Score-based generative neural networks for large-scale optimal transport,” *Advances in neural information processing systems*, vol. 34, pp. 12955–12965, 2021.
- [16] V. De Bortoli, J. Thornton, J. Heng, and A. Doucet, “Diffusion schrödinger bridge with applications to score-based generative modeling,” *Advances in Neural Information Processing Systems*, vol. 34, pp. 17695–17709, 2021.
- [17] A. Korotin, L. Li, J. Solomon, and E. Burnaev, “Continuous wasserstein-2 barycenter estimation without minimax optimization,” in *International Conference on Learning Representations*, 2021.
- [18] J. Fan, S. Liu, S. Ma, H.-M. Zhou, and Y. Chen, “Neural monge map estimation and its applications,” *Transactions on Machine Learning Research*, 2023. Featured Certification.
- [19] T. Uscidda and M. Cuturi, “The monge gap: A regularizer to learn all transport maps,” in *International Conference on Machine Learning*, pp. 34709–34733, PMLR, 2023.
- [20] B. Amos, “On amortizing convex conjugates for optimal transport,” in *The Eleventh International Conference on Learning Representations*, 2023.
- [21] A. Korotin, L. Li, A. Genevay, J. M. Solomon, A. Filippov, and E. Burnaev, “Do neural optimal transport solvers work? a continuous wasserstein-2 benchmark,” *Advances in neural information processing systems*, vol. 34, pp. 14593–14605, 2021.
- [22] H. Yan, X. Liu, J. Pan, J. H. Liew, Q. Liu, and J. Feng, “Perflow: Piecewise rectified flow as universal plug-and-play accelerator,” 2024.
- [23] K. Neklyudov, R. Brekelmans, D. Severo, and A. Makhzani, “Action matching: Learning stochastic dynamics from samples,” in *International conference on machine learning*, pp. 25858–25889, PMLR, 2023.
- [24] S. Shalev-Shwartz, A. Tewari, *et al.*, “On the duality of strong convexity and strong smoothness: Learning applications and matrix regularization,”
- [25] J.-Y. Zhu, T. Park, P. Isola, and A. A. Efros, “Unpaired image-to-image translation using cycle-consistent adversarial networks,” in *Proceedings of the IEEE international conference on computer vision*, pp. 2223–2232, 2017.
- [26] A. Korotin, N. Gushchin, and E. Burnaev, “Light schrödinger bridge,” in *The Twelfth International Conference on Learning Representations*, 2023.
- [27] S. Pidhorskyi, D. A. Adjeroh, and G. Doretto, “Adversarial latent autoencoders,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 14104–14113, 2020.
- [28] T. Karras, S. Laine, and T. Aila, “A style-based generator architecture for generative adversarial networks,” in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 4401–4410, 2019.

- [29] M. Heusel, H. Ramsauer, T. Unterthiner, B. Nessler, and S. Hochreiter, “Gans trained by a two time-scale update rule converge to a local nash equilibrium,” *Advances in neural information processing systems*, vol. 30, 2017.
- [30] B. Van Scy, R. A. Freeman, and K. M. Lynch, “The fastest known globally convergent first-order method for minimizing strongly convex functions,” *IEEE Control Systems Letters*, vol. 2, no. 1, pp. 49–54, 2017.
- [31] J. Hiriart-Urruty and Y. Lucet, “Parametric computation of the legendre-fenchel conjugate with application to the computation of the moreau envelope,” *Journal of Convex Analysis*, vol. 14, no. 3, p. 657, 2007.
- [32] A. Korotin, V. Egiazarian, L. Li, and E. Burnaev, “Wasserstein iterative networks for barycenter estimation,” *Advances in Neural Information Processing Systems*, vol. 35, pp. 15672–15686, 2022.
- [33] S. Chaudhari, S. Pranav, and J. M. Moura, “Gradient networks,” *arXiv preprint arXiv:2404.07361*, 2024.
- [34] C. Bunne, A. Krause, and M. Cuturi, “Supervised training of conditional monge maps,” *Advances in Neural Information Processing Systems*, vol. 35, pp. 6859–6872, 2022.
- [35] J. Richter-Powell, J. Lorraine, and B. Amos, “Input convex gradient networks,” *arXiv preprint arXiv:2111.12187*, 2021.
- [36] P.-J. Hoedt and G. Klambauer, “Principled weight initialisation for input-convex neural networks,” *Advances in Neural Information Processing Systems*, vol. 36, 2024.
- [37] S. Yang and B. W. Bequette, “Optimization-based control using input convex neural networks,” *Computers & Chemical Engineering*, vol. 144, p. 107143, 2021.
- [38] M. Ławryńczuk, “Input convex neural networks in nonlinear predictive control: A multi-model approach,” *Neurocomputing*, vol. 513, pp. 273–293, 2022.
- [39] Y. Chen, Y. Shi, and B. Zhang, “Optimal control via neural networks: A convex approach,” *arXiv preprint arXiv:1805.11835*, 2018.
- [40] R. T. Rockafellar, “Convex analysis,” 2015.
- [41] E. Polovinkin and M. Balashov, “Elements of convex and strongly convex analysis,” 2007.
- [42] S. Boyd and L. Vandenberghe, *Convex optimization*. Cambridge university press, 2004.
- [43] A. Korotin, V. Egiazarian, A. Asadulaev, A. Safin, and E. Burnaev, “Wasserstein-2 generative networks,” in *International Conference on Learning Representations*, 2021.
- [44] C.-W. Huang, R. T. Q. Chen, C. Tsirigotis, and A. Courville, “Convex potential flows: Universal probability distributions with optimal transport and convex optimization,” in *International Conference on Learning Representations*, 2021.
- [45] D. Morales-Brottons, T. Vogels, and H. Hendrikx, “Exponential moving average of weights in deep learning: Dynamics and benefits,” *Transactions on Machine Learning Research*, 2024.
- [46] K. He, H. Fan, Y. Wu, S. Xie, and R. Girshick, “Momentum contrast for unsupervised visual representation learning,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020.

- [47] J. R. Dormand and P. J. Prince, “A family of embedded runge-kutta formulae,” *Journal of computational and applied mathematics*, vol. 6, no. 1, pp. 19–26, 1980.

# Приложение

## 7.4 Доказательства

В этом разделе представлены доказательства всех результатов из основного текста работы, а также некоторых вспомогательных результатов. Доказательства Предположения 5.2 и Предположения 1 переставлены местами, поскольку первое основано на втором.

Отметим, что во всех теоретических выкладках, если не указано иное, предполагается дифференцируемость выпуклого потенциала  $\Psi$  в заданных точках  $z_0, x_t$ . Это предположение сделано для упрощения и не нарушает теорию. Известно, что выпуклые функции дифференцируемы почти всюду по мере Лебега [40]. Следовательно, поскольку рассматриваются абсолютно непрерывные распределения  $p_0, p_1$  (см. §4), дифференцируемость  $\Psi$  в рассматриваемых точках также выполняется почти наверное.

**Доказательства Предположения 1** (Упрощенная функция потерь OFM)

**Доказательство:** По определению  $\mathcal{L}_{OFM}^\pi(\Psi)$  равняется

$$\mathcal{L}_{OFM}^\pi(\Psi) := \int_0^1 \left\{ \int_{\mathbb{R}^D \times \mathbb{R}^D} \|u_t^\Psi(x_t) - (x_1 - x_0)\|^2 \pi(x_0, x_1) dx_0 dx_1 \right\} dt, x_t = (1-t)x_0 + tx_1. \quad (7.1)$$

Для фиксированных точек  $x_0, x_1$  и времени  $t$  находится точка  $z_0 = (\phi_t^\Psi)^{-1}(x_t)$ , такая что для момента  $t \in [0, 1]$  верно  $x_t = (1-t)x_0 + tx_1$ . Эта точка  $z_0$  удовлетворяет равенству

$$x_t = t\nabla\Psi(z_0) + (1-t)z_0.$$

Векторное поле  $u_t^\Psi$  определено как

$$u_t^\Psi(x_t) = \nabla\Psi(z_0) - z_0 = \frac{x_t - z_0}{t}.$$

Подставляя  $u_t^\Psi(x_t)$  в подынтегральное выражение (7.1), получается его упрощенная версия

$$\begin{aligned} \|x_1 - x_0 - u_t^\Psi(x_t)\|^2 &= \left\| x_1 - x_0 - \left( \frac{x_t - z_0}{t} \right) \right\|^2 \\ &= \frac{1}{t^2} \|tx_1 - tx_0 - ((1-t)x_0 + tx_1) + z_0\|^2 \\ &= \frac{1}{t^2} \|z_0 - x_0\|^2 = \left\| \frac{(\phi_t^\Psi)^{-1}(x_t) - x_0}{t} \right\|^2. \end{aligned}$$

□

**Доказательство Предположения 5.2** (Явная формула градиента функции потерь)

**Доказательство:** Точка  $z_0 = (\phi_t^{\Psi_\theta})^{-1}(x_t)$  зависит от параметров  $\theta$ . Дифференцирую подынтегральное выражение для упрощенной функции потерь (5.6) для фиксированных точек  $x_0, x_1$

и времени  $t$ , получается выражение

$$d \left( \frac{1}{t^2} \|z_0 - x_0\|^2 \right) = 2 \left\langle \frac{z_0 - x_0}{t^2}, \frac{dz_0}{d\theta} d\theta \right\rangle. \quad (7.2)$$

Для точки  $z_0$ , уравнение (7.7) также верно по построению

$$x_t = (1-t)z_0 + t\nabla\Psi_\theta(z_0). \quad (7.3)$$

Дифференцируя (7.3) по  $\theta$ , получается

$$\begin{aligned} 0 &= (1-t) \frac{dz_0}{d\theta} + t \nabla^2 \Psi_\theta(z_0) \frac{dz_0}{d\theta} + t \frac{\partial \nabla \Psi_\theta}{\partial \theta}(z_0) \Rightarrow \\ \frac{dz_0}{d\theta} &= - (t \nabla^2 \Psi_\theta(z_0) + (1-t)I)^{-1} \cdot t \frac{\partial \nabla \Psi_\theta}{\partial \theta}(z_0). \end{aligned}$$

Следовательно, выполняется равенство

$$\begin{aligned} (7.2) &= \left\langle 2 \frac{x_0 - z_0}{t}, (t \nabla^2 \Psi_\theta(z_0) + (1-t)I)^{-1} \frac{\partial \nabla \Psi_\theta}{\partial \theta}(z_0) d\theta \right\rangle \\ &= \left\langle 2 (t \nabla^2 \Psi_\theta(z_0) + (1-t)I)^{-1} \frac{(x_0 - z_0)}{t}, \frac{\partial \nabla \Psi_\theta}{\partial \theta}(z_0) d\theta \right\rangle. \end{aligned} \quad (7.4)$$

Теперь дифференцирование по  $\theta$  содержится только в правой части (7.4) в терме  $\frac{\partial \nabla \Psi_\theta}{\partial \theta}$ . Следовательно, точка  $z_0$  и левая часть (7.4) могут рассматриваться как константы при дифференцировании. Чтобы получить градиент OFM-лосса, необходимо также взять математическое ожидание по плану  $\pi$  и времени  $t$ .  $\square$

Следующие две леммы необходимы, чтобы доказать главную Теорему 5.1.

**Лемма 3 (Свойства выпуклых функций и их сопряжений)** Пусть функция  $\Psi : \mathbb{R}^D \rightarrow \mathbb{R}$  является выпуклой, а точки  $x_0, x_1 \in \mathbb{R}^D$ . Пусть  $\Psi$  и  $\bar{\Psi}$  дифференцируемы в точках  $x_0$  и  $x_1$  соответственно. Тогда следующие утверждения эквивалентны:

1.  $x_1 = \nabla\Psi(x_0)$ ;
2.  $x_0 = \arg \max_{z \in \mathbb{R}^D} \{ \langle x_1, z \rangle - \Psi(z) \}$ ;
3. Равенство Фенхеля-Янга:  $\Psi(x_0) + \bar{\Psi}(x_1) = \langle x_1, x_0 \rangle$ ;
4.  $x_0 = \nabla\bar{\Psi}(x_1)$ ;
5.  $x_1 = \arg \max_{z \in \mathbb{R}^D} \{ \langle z, x_0 \rangle - \bar{\Psi}(z) \}$ ;

**Доказательство:** Эта лемма является упрощенной версией [41, Теоремы 1.16.4]. Также доказательство можно получить, комбинируя факты из [42, §3.3].  $\square$

**Лемма 4 (Главная Лемма об интегрировании)** Для двух точек  $x_0, x_1 \in \mathbb{R}^D$  и выпуклой функции  $\Psi$  следующие равенства верны

$$\int_0^1 \|u_t^\Psi(x_t) - (x_1 - x_0)\|^2 dt = 2 \cdot [\Psi(x_0) + \bar{\Psi}(x_1) - \langle x_0, x_1 \rangle], \quad (7.5)$$

$$\text{т.е. } x_t = tx_0 + (1-t)x_1.$$

**Доказательство:** Следуя Предположению 1, используется упрощенная форма функции потерь

$$\|u_t^\Psi(x_t) - (x_1 - x_0)\|^2 = \frac{1}{t^2} \|z_0 - x_0\|^2, \quad (7.6)$$

где точка  $z_0 = z_0(t) = (\phi_t^\Psi)^{-1}(x_t)$  удовлетворяет:

$$x_t = t\nabla\Psi(z_0) + (1-t)z_0. \quad (7.7)$$

Далее выражение (7.6) подставляется в правую часть (7.5):

$$\int_0^1 \|u_t^\Psi(x_t) - (x_1 - x_0)\|^2 dt = \int_0^1 \frac{1}{t^2} \|z_0 - x_0\|^2 dt. \quad (7.8)$$

Для дальнейшего упрощения (7.8) необходимы предварительные результаты. Используя (7.7), становятся верными равенства:

$$\begin{aligned} x_t &= t\nabla\Psi(z_0) + (1-t)z_0 &= (1-t)x_0 + tx_1 \Rightarrow \\ t(\nabla\Psi(z_0) - x_1) &= (1-t)(x_0 - z_0) \Rightarrow \\ (\nabla\Psi(z_0) - x_1) &= \left(\frac{1-t}{t}\right)(x_0 - z_0) \Rightarrow \end{aligned} \quad (7.9)$$

$$\|\nabla\Psi(z_0) - x_1\|^2 = \frac{(1-t)^2}{t^2} \|z_0 - x_0\|^2. \quad (7.10)$$

Заменяя в (7.8) переменную времени  $t$  на  $s = \frac{t}{1-t}$ ,  $ds = \frac{dt}{(1-t)^2}$ , из (7.10) получается:

$$\int_0^1 \frac{1}{t^2} \|z_0(t) - x_0\|^2 dt = \int_0^1 \frac{(1-t)^2}{t^2} \|z_0(t) - x_0\|^2 \frac{dt}{(1-t)^2} = \int_0^\infty \|\nabla\Psi(z_0(s)) - x_1\|^2 ds. \quad (7.11)$$

Точки  $z_0(s(t)) = (\phi_t^\Psi)^{-1}(x_t)$ ,  $t \in (0, 1)$  формируют кривую в пространстве  $\mathbb{R}^D$  с параметризацией через  $t$  (или  $s(t)$ ). В формуле (7.11) заменяется интегрирование по переменной  $s$  на интегрирование вдоль кривой. Для этого необходимы следующие две вещи

1. Пределы интегрирования. Пределы интегрирования по кривой  $z_0(t)$ :

$$\begin{aligned} z_0(t)|_{t=0} &= x_0, \\ z_0(t)|_{t=1} &= (\nabla\Psi)^{-1}(x_1) \stackrel{\text{Лемма 3; 1.4}}{=} \nabla\bar{\Psi}(x_1). \end{aligned} \quad (7.12)$$

2. Замена дифференциала  $dz_0$ . Начиная с (7.9), получается:

$$\begin{aligned} (7.9) \Rightarrow s(\nabla\Psi(z_0) - x_1) &= (x_0 - z_0) \Rightarrow \\ d[s(\nabla\Psi(z_0) - x_1)] &= d[x_0 - z_0] \Rightarrow \\ s\nabla^2\Psi(z_0)dz_0 + (\nabla\Psi(z_0) - x_1)ds &= -dz_0 \Rightarrow \\ (\nabla\Psi(z_0) - x_1)ds &= -(s\nabla^2\Psi(z_0) + I)dz_0. \end{aligned} \quad (7.13)$$

Продолжая с (7.11), выводятся:

$$\begin{aligned}
 (7.11) &= \int_0^\infty \langle \nabla \Psi(z_0) - x_1, \nabla \Psi(z_0) - x_1 \rangle ds \\
 &\stackrel{(7.13)}{=} \int_{z_0} \langle x_1 - \nabla \Psi(z_0), (s \nabla^2 \Psi(z_0) + I) dz_0 \rangle \\
 &= \int_{z_0} \langle x_1 - \nabla \Psi(z_0), dz_0 \rangle + \int_{z_0} \langle s(x_1 - \nabla \Psi(z_0)), \nabla^2 \Psi(z_0) dz_0 \rangle \\
 &\stackrel{(7.9)}{=} \int_{z_0} \langle x_1 - \nabla \Psi(z_0), dz_0 \rangle + \int_{z_0} \langle z_0 - x_0, \nabla^2 \Psi(z_0) dz_0 \rangle. \tag{7.14}
 \end{aligned}$$

Далее будут использованы равенства

$$\begin{aligned}
 d\langle z_0, \nabla \Psi(z_0) \rangle &= \langle z_0, \nabla^2 \Psi(z_0) dz_0 \rangle + \langle dz_0, \nabla \Psi(z_0) \rangle \Rightarrow \\
 \langle z_0, \nabla^2 \Psi(z_0) dz_0 \rangle &= d\langle z_0, \nabla \Psi(z_0) \rangle - \langle \nabla \Psi(z_0), dz_0 \rangle.
 \end{aligned}$$

И как следствие, можно заменить (7.14):

$$\begin{aligned}
 (7.14) &= \int_{z_0} \langle x_1 - \nabla \Psi(z_0), dz_0 \rangle + \int_{z_0} \langle z_0 - x_0, \nabla^2 \Psi(z_0) dz_0 \rangle \\
 &= \int_{z_0} \langle x_1, dz_0 \rangle - \int_{z_0} \langle \nabla \Psi(z_0), dz_0 \rangle \\
 &\quad + \int_{z_0} d\langle z_0, \nabla \Psi(z_0) \rangle - \int_{z_0} \langle \nabla \Psi(z_0), dz_0 \rangle - \int_{z_0} \langle x_0, \nabla^2 \Psi(z_0) dz_0 \rangle \\
 &= \int_{z_0} \langle x_1, dz_0 \rangle - 2 \int_{z_0} \langle \nabla \Psi(z_0), dz_0 \rangle + \int_{z_0} d\langle z_0, \nabla \Psi(z_0) \rangle - \int_{z_0} \langle x_0, \nabla^2 \Psi(z_0) dz_0 \rangle \tag{7.15}
 \end{aligned}$$

Все подынтегральные выражения имеют замкнутые дифференциалы

$$\begin{aligned}
 \langle x_1, dz_0 \rangle &= d\langle x_1, z_0 \rangle, \\
 \langle \nabla \Psi(z_0), dz_0 \rangle &= d\Psi(z_0), \\
 \langle x_0, \nabla^2 \Psi(z_0) dz_0 \rangle &= d\langle x_0, \nabla \Psi(z_0) \rangle.
 \end{aligned}$$

Интегрируя их с начальной точки  $x_0$  и до конечной  $\nabla \bar{\Psi}(x_1)$  согласно пределам (7.12), получается

$$\begin{aligned}
 (7.15) &= \int_{z_0} d\langle x_1, z_0 \rangle - 2 \int_{z_0} d\Psi(z_0) + \int_{z_0} d\langle z_0, \nabla \Psi(z_0) \rangle - \int_{z_0} d\langle x_0, \nabla \Psi(z_0) \rangle \\
 &= \langle x_1, \nabla \bar{\Psi}(x_1) \rangle - \langle x_1, x_0 \rangle + 2(\Psi(x_0) - \Psi(\nabla \bar{\Psi}(x_1))) + \langle (\nabla \bar{\Psi}(x_1), \nabla \Psi(\nabla \bar{\Psi}(x_1))) \rangle \\
 &\quad - \langle x_0, \nabla \Psi(x_0) \rangle + \langle x_0, \nabla \Psi(x_0) \rangle - \langle x_0, \nabla \Psi(\nabla \bar{\Psi}(x_1)) \rangle. \tag{7.16}
 \end{aligned}$$

Далее используются свойства сопряженных функций (Лемма (3)):

$$\begin{aligned}
 \Psi(\nabla \bar{\Psi}(x_1)) &\stackrel{4+3}{=} \langle \nabla \bar{\Psi}(x_1), x_1 \rangle - \bar{\Psi}(x_1), \\
 \nabla \Psi(\nabla \bar{\Psi}(x_1)) &\stackrel{4+1}{=} x_1.
 \end{aligned}$$

Это позволяет упростить (7.16):

$$\begin{aligned}
(7.16) &= \langle x_1, \nabla \bar{\Psi}(x_1) \rangle - \langle x_1, x_0 \rangle + 2(\Psi(x_0) + \bar{\Psi}(x_1) - \langle \nabla \bar{\Psi}(x_1), x_1 \rangle) + \langle (\nabla \bar{\Psi}(x_1), x_1) \\
&\quad - \langle x_0, \nabla \Psi(x_0) \rangle + \langle x_0, \nabla \Psi(x_0) \rangle - \langle x_0, x_1 \rangle \\
&= 2[\Psi(x_0) + \bar{\Psi}(x_1) - \langle x_0, x_1 \rangle].
\end{aligned}$$

□

Интегрирование равенства (7.5) по транспортному плану  $\pi$  и учет формул функций потерь (4.5) и (5.3) позволяют доказать Теорему 5.1.

### Доказательство Теоремы 5.1

**Доказательство:** Главная Лемма об интегрировании 4 утверждает, что для любых фиксированных точек  $x_0, x_1$  верно

$$\int_0^1 \|x_1 - x_0 - u_t^\Psi(x_t)\|^2 dt = 2[\Psi(x_0) + \bar{\Psi}(x_1) - \langle x_0, x_1 \rangle].$$

Навешивания мат ожидания по плану  $\pi$  (интегрирования по точкам  $x_0, x_1 \sim \pi$ ) даёт

$$\underbrace{\mathbb{E}_{x_0, x_1 \sim \pi} \int_0^1 \|u_t^\Psi(x_t) - (x_1 - x_0)\|^2 dt}_{=\mathcal{L}_{OFM}^\pi(\Psi)} = 2 \cdot \underbrace{\mathbb{E}_{x_0, x_1 \sim \pi} [\Psi(x_0) + \bar{\Psi}(x_1)]}_{=\mathcal{L}_{OT}(\Psi)} - \underbrace{2 \cdot \mathbb{E}_{x_0, x_1 \sim \pi} [\langle x_0, x_1 \rangle]}_{=: \text{Const}'(\pi)},$$

где  $\text{Const}'(\pi)$  не зависит от  $\Psi$ . Поэтому минимумы обеих функций потерь  $\mathcal{L}_{OFM}^\pi(\Psi)$  и  $\mathcal{L}_{OT}(\Psi)$  достигаются на одних и тех же функциях  $\Psi$ . □

### Доказательство Предположения 2 (Невычислимое расстояние)

**Доказательство:** По определениям  $\text{dist}(u, u^*)$  (5.7) и  $\mathcal{L}_{FM}^{\pi^*}(u)$  (4.7) имеется:

$$\begin{aligned}
\text{dist}(u, u^*) &= \int_0^1 \int_{\mathbb{R}^D} \|u_t(x_t) - u_t^*(x_t)\|^2 \underbrace{\phi_t^* \# p_0(x_t)}_{=p_t^*(x_t)} dx_t dt, \\
\mathcal{L}_{FM}^{\pi^*}(u) &= \int_0^1 \left\{ \int_{\mathbb{R}^D \times \mathbb{R}^D} \|u_t(x_t) - (x_1 - x_0)\|^2 \pi^*(x_0, x_1) dx_0 dx_1 \right\}, \quad x_t = (1-t)x_0 + tx_1.
\end{aligned}$$

Для оптимального плана  $\pi^*$ , каждая точка  $x_0$  почти наверное отправляется в  $\nabla \Psi^*(x_0)$ . Поэтому в функции потерь FM можно интегрировать только по  $x_0$  с заменой  $x_1 = \nabla \Psi^*(x_0)$  для фиксированного  $t$ :

$$\begin{aligned}
\int_{\mathbb{R}^D \times \mathbb{R}^D} \|u_t(x_t) - (x_1 - x_0)\|^2 \pi^*(x_0, x_1) dx_0 dx_1 &= \int_{\mathbb{R}^D} \|u_t(x_t) - (\nabla \Psi^*(x_0) - x_0)\|^2 p_0(x_0) dx_0, \\
x_t &= (1-t)x_0 + t\nabla \Psi^*(x_0).
\end{aligned} \tag{7.17}$$

Поле динамического ОТ  $u^* = u^{\Psi^*}$  с оптимальным потенциалом  $\Psi^*$ . Более того, для любой точки  $x_t = (1-t)x_0 + t\nabla \Psi^*(x_0)$ , порожденной  $u^*$ , верно то, что  $u_t^*(x_t) = u_t^{\Psi^*}(x_t) = \nabla \Psi^*(x_0) -$

$x_0$ . Это выражение совпадает с (7.17):

$$\begin{aligned} (7.17) &= \int_{\mathbb{R}^D} \|u_t(x_t) - (\nabla\Psi^*(x_0) - x_0)\|^2 p_0(x_0) dx_0 \\ &= \int_{\mathbb{R}^D} \|u_t(x_t) - u_t^*(x_t)\|^2 p_0(x_0) dx_0, \quad x_t = (1-t)x_0 + t\nabla\Psi^*(x_0). \end{aligned}$$

Заменяя переменную  $x_0$  на  $x_t = \phi_t^*(x_0)$ , и вероятность  $p_0(x_0)dx_0 = \phi_t^*\#p_0(x_t)dx_t = p_t^*(x_t)dx_t$ , получается

$$\int_{\mathbb{R}^D \times \mathbb{R}^D} \|u_t(x_t) - (x_1 - x_0)\|^2 \pi^*(x_0, x_1) dx_0 dx_1 = \int_{\mathbb{R}^D} \|u_t(x_t) - u_t^*(x_t)\|^2 p_t^*(x_t) dx_t.$$

Поэтому интегрирование по времени  $t$  даёт желаемый результат

$$\text{dist}(u, u^*) = \mathcal{L}_{FM}^{\pi^*}(u),$$

с  $\mathcal{L}_{FM}^{\pi^*}(u^*) = \text{dist}(u^*, u^*) = 0$ .  $\square$

**Доказательство Предположения 3** (Вычислимое расстояние для OFM)

**Доказательство:** Для векторного поля  $u^\Psi$  применение формулы из Предположения 2 даёт

$$\text{dist}(u^\Psi, u^{\Psi^*}) = \mathcal{L}_{FM}^{\pi^*}(u^\Psi) - \mathcal{L}_{FM}^{\pi^*}(u^{\Psi^*}) \stackrel{(5.3)}{=} \mathcal{L}_{OFM}^{\pi^*}(\Psi) - \mathcal{L}_{OFM}^{\pi^*}(\Psi^*).$$

Согласно Лемме 4 для любого плана  $\pi$  и выпуклой функции  $\Psi$  верно следующее равенство

$$\underbrace{\mathbb{E}_{x_0, x_1 \sim \pi} \int_0^1 \|u_t^\Psi(x_t) - (x_1 - x_0)\|^2 dt}_{=\mathcal{L}_{OFM}^{\pi}(\Psi)} = 2 \cdot \underbrace{\mathbb{E}_{x_0, x_1 \sim \pi} [\Psi(x_0) + \bar{\Psi}(x_1)]}_{=\mathcal{L}_{OT}(\Psi)} - 2 \cdot \underbrace{\mathbb{E}_{x_0, x_1 \sim \pi} [\langle x_0, x_1 \rangle]}_{=: \text{Const}'(\pi)}.$$

Поскольку значение  $\text{Const}'(\pi)$  не зависит от  $\Psi$ , то это значения совпадают для  $\Psi, \Psi^*$  и можно вывести

$$\begin{cases} \mathcal{L}_{OFM}^{\pi}(\Psi) = 2 \cdot \mathcal{L}_{OT}(\Psi) - \text{Const}'(\pi), \\ \mathcal{L}_{OFM}^{\pi}(\Psi^*) = 2 \cdot \mathcal{L}_{OT}(\Psi^*) - \text{Const}'(\pi) \end{cases} \Downarrow \mathcal{L}_{OFM}^{\pi}(\Psi) - \mathcal{L}_{OFM}^{\pi}(\Psi^*) = 2 \cdot \mathcal{L}_{OT}(\Psi) - 2 \cdot \mathcal{L}_{OT}(\Psi^*). \quad (7.18)$$

Правая часть (7.18) не зависит от плана  $\pi$ , и поэтому левая часть является инвариантом для любого плана, в том числе и оптимального  $\pi^*$ :

$$\mathcal{L}_{OFM}^{\pi}(\Psi) - \mathcal{L}_{OFM}^{\pi}(\Psi^*) = \mathcal{L}_{OFM}^{\pi^*}(\Psi) - \mathcal{L}_{OFM}^{\pi^*}(\Psi^*) = \text{dist}(u^\Psi, u^{\Psi^*}).$$

$\square$

## 7.5 Action Matching

В этом разделе показывается, что идеи OFM могут быть применены в рамках метода Action Matching (AM) [23]. В данной постановке доступен для сэмплирования стохастический про-

цесс, описываемый промежуточными распределениями  $p_t$ . Цель состоит в нахождении зависящей от времени функции  $s_t : [0, 1] \times \mathbb{R}^D \rightarrow \mathbb{R}$ , такой, что векторное поле  $u_t = \nabla s_t$  порождает рассматриваемый процесс. Функция  $s_t$  может быть найдена путем минимизации следующей функции потерь АМ:

$$\begin{aligned}\mathcal{L}_{AM}(s) &:= \int_{\mathbb{R}^D} s_0(x_0)p_0(x_0)dx_0 - \int_{\mathbb{R}^D} s_1(x_1)p_1(x_1)dx_1 \\ &+ \int_0^1 \int_{\mathbb{R}^D} \left[ \frac{1}{2} \|\nabla s_t(x_t)\|^2 + \frac{\partial s_t}{\partial t}(x_t) \right] p_t(x_t)dx_t dt.\end{aligned}\quad (7.19)$$

В дальнейшем будут рассматриваться оптимальные векторные поля  $u^\Psi$ , заданные выпуклой функцией  $\Psi$ . Для нахождения явной формулы  $s^\Psi$ , градиент которой равен  $u^\Psi$ , т.е.,  $u_t^\Psi \equiv \nabla s_t^\Psi$ , необходимо использовать факт, что для  $x_t \in \mathbb{R}^D$  точка  $z_0 \in \mathbb{R}^D$  удовлетворяет

$$\begin{aligned}x_t &= t\nabla\Psi(z_0) + (1-t)z_0, \\ x_t &= \nabla \left( t\Psi(\cdot) + \frac{(1-t)}{2} \|\cdot\|^2 \right) (z_0) := \nabla\varphi_t(z_0), \\ z_0 &= \nabla\overline{\varphi}_t(x_t).\end{aligned}$$

Векторное поле  $u_t^\Psi(x_t)$  можно выразить как:

$$\begin{aligned}u_t^\Psi(x_t) &= \nabla\Psi(z_0) - z_0 = \frac{x_t - z_0}{t} = \nabla \left( \underbrace{\frac{\|\cdot\|^2}{2t}}_{=:s_t} - \frac{\overline{\varphi}_t}{t} \right) (x_t), \\ s_t(x_t) &= \frac{\|x_t\|^2}{2t} - \frac{\overline{\varphi}_t(x_t)}{t}.\end{aligned}\quad (7.20)$$

Границные случаи  $t = 0$  и  $t = 1$  равны:

$$\begin{aligned}s_1(x_1) &= \frac{\|x_1\|^2}{2} - \overline{\Psi}(x_1), \\ s_0(x_0) &= \Psi(x_0) - \frac{\|x_0\|^2}{2}.\end{aligned}$$

Для времени  $t \in (0, 1)$  можно заметить, что

$$\frac{1}{2} \|\nabla s_t(x_t)\|^2 = \frac{1}{2} \frac{\|x_t - z_0\|^2}{t^2}.$$

Если взять производную  $s_t$  по  $t$ , то слагаемое  $\|\cdot\|^2/2t$  из (7.20) можно не рассматривать. Поэтому остаётся только слагаемое

$$\frac{\overline{\varphi}_t(x_t)}{t} = \frac{1}{t} \max_{z \in \mathbb{R}^D} \left\{ \langle x_t, z \rangle - t\Psi(z) - \frac{(1-t)}{2} \|z\|^2 \right\} = \max_{z \in \mathbb{R}^D} \left\{ \frac{\langle x_t, z \rangle}{t} - \Psi(z) - \frac{(1-t)}{2t} \|z\|^2 \right\}.$$

Более того, максимум достигается в точке  $z_0$ . Согласно теореме об огибающей, для взятия производной от максимума по времени  $t$  необходимо продифференцировать максимизируемую функцию и затем подставить точку  $z_0$ , в которой достигается максимум. Математически это

выражается как:

$$\frac{\partial(\overline{\varphi_t}/t)}{\partial t}(x_t) = -\frac{\langle x_t, z_0 \rangle}{t^2} + \frac{\|z_0\|^2}{2t^2}. \quad (7.21)$$

Подставляя (7.21) в (7.20), можно получить

$$\frac{\partial s_t}{\partial t}(x_t) = -\frac{\|x_t\|^2}{2t^2} + \frac{\langle x_t, z_0 \rangle}{t^2} - \frac{\|z_0\|^2}{2t^2} = -\frac{1}{2} \frac{\|x_t - z_0\|^2}{t^2}.$$

Поэтому в случае оптимальных векторных полей для любых  $t \in [0, 1]$  и  $x_t \in \mathbb{R}^D$  верно

$$\left[ \frac{1}{2} \|\nabla s_t(x_t)\|^2 + \frac{\partial s_t}{\partial t}(x_t) \right] = \frac{1}{2} \frac{\|x_t - z_0\|^2}{t^2} - \frac{1}{2} \frac{\|x_t - z_0\|^2}{t^2} \equiv 0.$$

В заключение, функция потерь АМ полностью эквивалентна двойственной форме функции потерь ОТ:

$$\begin{aligned} \mathcal{L}_{AM}(s^\Psi) &= \int_{\mathbb{R}^D} \Psi(x_0) p_0(x_0) dx_0 + \int_{\mathbb{R}^D} \bar{\Psi}(x_1) p_1(x_1) dx_1 \\ &- \int_{\mathbb{R}^D} \frac{\|x_0\|^2}{2} p_0(x_0) dx_0 - \int_{\mathbb{R}^D} \frac{\|x_1\|^2}{2} p_1(x_1) dx_1. \end{aligned}$$

## 7.6 Технические детали экспериментов

### Реализация OFM

Для практической реализации подхода OFM используются полносвязанные архитектуры ICNN, предложенные в [43, Приложение B2] (W2GN\_ICNN) и [44, Приложение E1] (CPF\_ICNN). Чтобы гарантировать выпуклость, обе архитектуры накладывают определённые ограничения на веса нейронной сети и используемые функции активации (подробности см. в соответствующих статьях). Реализации взяты из их официальных репозиториев:

<https://github.com/iamalexkorotin/Wasserstein2Benchmark>;  
<https://github.com/CW-Huang/CP-Flow>.

Основное различие между W2GN\_ICNN и CPF\_ICNN заключается в том, что W2GN\_ICNN использует функцию активации *CELU* и выпуклые квадратичные skip-connection, тогда как CPF\_ICNN применяет функцию активации *Softplus*.

Гиперпараметры Алгоритма 1 и используемых ICNN для различных экспериментов собраны в Таблице 7.1. Во всех экспериментах в качестве оптимизатора *SubOpt* используется LBFGS (`torch.optim.LBFGS`) с  $K_{\text{sub}}$  шагами оптимизации и критерием ранней остановки, основанным на норме градиента. Что касается последнего, параметр `tolerance_grad` в `torch.optim.LBFGS` устанавливается равным  $gtol$ . Для нахождения начальной точки  $z_0^i$  (Шаг 5 Алгоритма 1), *SubOpt* инициализируется значением  $x_{ti}^i$ .

Интересно отметить, что на практике OFM метод устойчив к качеству восстановленных начальных точек  $z_0^i$  (Шаг 5 Алгоритма 1). В частности, значений  $K_{\text{sub}} \approx 5 - 10$  и  $gtol \approx 0.01$  оказалось достаточно для большинства экспериментов.

В качестве оптимизатора *Opt* используется Adam с learning rate  $lr$  и остальными гиперпараметрами по умолчанию.

Эксперимент	ICNN архитектура $\Psi_\theta$	$K$	$B$	$lr$	$K_{\text{sub}}$
Иллюстрации 2D	$\text{CPF\_ICNN}, \mathbb{R}^2 \rightarrow \mathbb{R}$ , Softplus, [1024, 1024]	30K	1024	$10^{-2}$	5
W2 бенчмарк, раз. $D$	$\text{W2GN\_ICNN}, \mathbb{R}^D \rightarrow \mathbb{R}$ , CELU, [128, 128, 64]	30K	1024	$10^{-3}$	50
ALAE	$\text{W2GN\_ICNN}, \mathbb{R}^{512} \rightarrow \mathbb{R}$ , CELU, [1024, 1024]	10K	128	$10^{-3}$	10

Таблица 7.1: Гиперпараметры OFM для различных экспериментов.

## Детали бенчмарка

В экспериментах используется экспоненциальное скользящее среднее (EMA, Exponential Moving Average) [45, 46] весов обученной модели. EMA создает сглаженную копию модели, веса которой обновляются на каждой новой итерации обучения  $t + 1$  по формуле:

$$\theta_{t+1}^{\text{ema}} = \alpha \theta_t^{\text{ema}} + (1 - \alpha) \theta_{t+1},$$

где  $\theta_{t+1}$  — обновленные веса модели. Финальные метрики рассчитываются при  $\alpha = 0.999$ .

**Детали реализации солверов.** Архитектуры нейронных сетей конкурирующих методов Flow Matching и их параметры, использованные в сравнительных экспериментах, представлены в Таблице 7.2. В таблице обозначение "FC" соответствует полностью соединенным слоям (fully-connected).

Солвер	Архитектура	Функция активации	Скрытые слои	Оптимизатор	Размер батча	Learning rate	Итер. за раунд * раунд
OT CFM [9]	FC NN $\mathbb{R}^D \times [0, 1] \rightarrow \mathbb{R}^D$	ReLU	[128, 128, 64]	RMSprop	1024	$10^{-3}$	200.000
RF [6]	FC NN $\mathbb{R}^D \times [0, 1] \rightarrow \mathbb{R}^D$	ReLU	[128, 128, 64]	RMSProp	1024	$10^{-4}$	65.000 * 3
c-RF [5]	FC NN $\mathbb{R}^D \times [0, 1] \rightarrow \mathbb{R}$	ReLU	[128, 128, 64]	RMSProp	1024	$10^{-5}$	100.000 * 2

Таблица 7.2: Параметры моделей, обученных на бенчмарке с размерностями  $D = 2, 4, 8, 16, 32, 64, 128, 256$ .

Временная переменная  $t$  в архитектурах ( $c-$ )RF и OT-CFM добавляется как дополнительный слой на входе без специальной предобработки. В RF и  $c$ -RF обыкновенные дифференциальные уравнения (ОДУ) решаются явным методом Рунге-Кутты порядка 5(4) [47] с абсолютной погрешностью  $10^{-4} - 10^{-6}$ . В OFM и  $c$ -RF градиенты по входу вычисляются через автоматическое дифференцирование PyTorch.

Следуя авторам RF [6], в экспериментах выполняются только 2 – 3 раунда минимизации в RF. В последующих раундах прямолинейность траекторий и метрики изменяются незначительно, в то время как ошибка обучения целевому распределению продолжает накапливаться.

Реализации методов OT-CFM [9] и RF [6] основаны на официальных репозиториях:

<https://github.com/atong01/conditional-flow-matching>  
<https://github.com/gnobitab/RectifiedFlow>

Реализация  $c$ -RF следует framework'у RF с модификацией архитектуры оптимизируемой нейросети. Вместо сети  $\mathbb{R}^D \times [0, 1] \rightarrow \mathbb{R}^D$  используется параметризация временно-зависимой модели со скалярным выходом  $\mathbb{R}^D \times [0, 1] \rightarrow \mathbb{R}$ , градиенты которой задают векторное поле.