ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ
УЧРЕЖДЕНИЕ НАУКИ
САНКТ-ПЕТЕРБУРГСКОЕ ОТДЕЛЕНИЕ
МАТЕМАТИЧЕСКОГО ИНСТИТУТА ИМ. В. А. СТЕКЛОВА
РОССИЙСКОЙ АКАДЕМИИ НАУК

На правах рукописи

Кноп Александр Анатольевич

Сложность эвристических вычислений и интерактивных протоколов

01.01.06 — математическая логика, алгебра и теория чисел

Диссертация
на соискание ученой степени
кандидата физико-математических наук

Научный руководитель:
д.ф.-м.н. Э. А. Гирш

Санкт-Петербург
2016

Оглавление

Введение

Актуальность темы

В классической теории сложности рассматривается время работы алгоритма (интерактивного протокола) в наихудшем случае. Однако для приложений, как правило, интереснее среднее время работы. В связи с этим в 1986 году Левин начал исследования в на тот момент новом разделе теории сложности — теории сложности в среднем.

По аналогии с классической теорией сложности (или теорией сложности в наихудшем случае), в теории сложности в среднем наибольший интерес представляет связь между различными вычислительными ресурсами. В данной работе нас будут более всего интересовать следующие вопросы:

1) насколько ресурс «время работы» существенен, правда ли, что за большее время можно решить большее число задач, и если да, то насколько;

2) насколько использование подсказки, зависящей от длины входа, увеличивает вычислительные возможности.

В терминах структурной сложности эти вопросы можно сформулировать следующим образом:

1) для каких моделей вычислений $\mathfrak{C}$ верно, что для любого k выполняется $\mathfrak{C}\mathbf{P} \not\subseteq \mathfrak{C}\mathbf{Time}[n^k]$, то есть полиномиальные по времени вычисления не моделируются за время n^k;

2) для каких моделей вычислений $\mathfrak{C}$ верно, что для любого k выполняется $\mathfrak{C}\mathbf{P} \not\subseteq \mathbf{Size}[n^k]$, то есть полиномиальные по времени вычисления не моделируются вычислениями при помощи булевых схем размера n^k.

В классическом случае оба этих вопроса решены не полностью. Исследование первого из них началось в 1967 году с работы Хартманиса и Стернса [1], в которой они доказали, что $\mathbf{P} \not\subseteq \mathbf{DTime}[n^k]$ для любого k. Для доказательства этого результата ими была использована техника диагонализации. Однако этот метод невозможно перенести на классы, не замкнутые относительно отрицания, такие как **NP**. Для этих классов вопрос о существовании иерархии оставался открытым, пока в 1973 году Кук [2] не доказал, что $\mathbf{NP} \not\subseteq \mathbf{NTime}[n^k]$ для всех k. Его доказательство базировалось на существовании **NP**-полной задачи и на иерархии для детерминированных вычислений. К сожалению, доказательство было сложным и не переносилось на другие классы. Десять лет спустя, в 1983 году, Зак [3] предложил новое, более простое, доказательство, основанное на отложенной диагонализации. Это доказательство позволило доказать иерархию для всех синтаксических моделей вычислений, но для семантических же моделей, таких как **BPP**, вопрос до сих пор открыт. В 2004 году Фортноу и Сантанам [4] совершили прорыв и доказали иерархию по времени для *эвристических* вероятностных алгоритмов с ограниченной ошибкой (они доказали, что $\mathrm{Heur}_\delta\mathbf{BPP} \not\subseteq \mathrm{Heur}_\delta\mathbf{BPTime}[n^k]$), для доказательства этого факта они воспользовались существованием оптимального алгоритма для **PSpace**-полного языка, их доказательство было технически сложным и не переносилось на другие модели вычислений. В 2007 году Первышев [5] улучшил этот результат, доказав, что $\mathrm{Heur}_\delta\mathbf{BPP} \not\subseteq \mathrm{Heur}_{\frac{1}{2}-\delta}\mathbf{BPTime}[n^k]$ для всех k, и разработав технику, позволяющую доказать теоремы об иерархии для других семантических моделей вычислений (таких как интерактивные протоколы). В то же время вопрос об увеличении параметра ошибки выше $\frac{1}{2} - \delta$ все еще открыт.

Исследования второго вопроса начались в 1982 году с работы Канна-

на [6], в которой он доказал, что $\Sigma_2\mathbf{P} \not\subseteq \mathbf{Size}[n^k]$ для всех k. Доказательство базировалось на теореме Карпа–Липтона [7]. В 2001 Кай [8] заметил, что теорему Карпа–Липтона можно усилить и тем самым доказать, что $\mathbf{S}_2\mathbf{P} \not\subseteq \mathbf{Size}[n^k]$ для любого k. К сожалению, успехи в классическом случае на этом закончились. Однако в 2009 Сантанам [9] доказал, что $\mathbf{MA}/1 \not\subseteq \mathbf{Size}[n^k]$, тем самым улучшив предыдущие результаты.

Во всех вышеупомянутых результатах среднее время работы считалось по равномерному распределению, но естественно было бы рассматривать и другие распределения. Однако, если не ограничивать никак класс распределений, то становятся верны несколько парадоксальные утверждения: так, в 1992 году Ли и Витани [10] доказали, что существует такое распределение D, что $(L, D) \in \mathrm{Heur}_{\frac{1}{n^3}}\mathbf{P}$ тогда и только тогда, когда $L \in \mathbf{P}$. В связи с этим, как правило, рассматривают распределения из какого-нибудь естественного класса. Своеобразным аналогом вопроса об иерархии по времени является следующий вопрос: может ли усложнение распределения увеличить сложность языка и, наоборот, можно ли уменьшить сложность языка, увеличив сложность распределения. В 1987 году Гуревич и Шелах [11] доказали, что существует алгоритм, который проверяет граф на гамильтоновость за линейное время в среднем на равномерном распределении, что косвенно указывает на возможность положительного ответа на этот вопрос.

В следующих секциях мы детально рассмотрим каждый из трех вопросов.

Степень разработанности темы

Нижние оценки на схемную сложность

Широко известно доказательство подсчетом того, что существуют булевы функции суперполиномиальной схемной сложности. Однако все попытки доказать суперполиномиальную нижнюю оценку для явной функции (функции из класса $\mathbf{NP}$) до сих пор не увенчались успехом...

Существует три основных направления, с которых пытаются подступиться к решению данной проблемы. Самый очевидный подход заключается в попытках доказать какие-нибудь оценки на функции из **NP**, но на данный момент лучшим результатом в этой области является $3.01n - o(n)$ [12] (оценку можно улучшить до $5n - o(n)$, если рассматривать схемы в базисе де Моргана [13]). Другим вариантом является исследование ограниченных классов схем. Это направление оказывается более успешным, известна экспоненциальная нижняя оценка на монотонную схемную сложность, а также на схемы с ограниченной глубиной. Однако и это направление не привело к успехам в общем случае.

Последнее направление — это попытки доказать нижние оценки для функций из все меньших и меньших классов. Экспоненциальная нижняя оценка, полученная подсчетом, требует дважды экспоненциального времени. Бурман и другие [14] показали, что данную оценку можно усилить и найти функцию экспоненциальной сложности в классе $\mathbf{MA_{Exp}}$. Менее амбициозной целью является доказательство нижних оценок вида n^k (для всех k). Это направление исследований было начато Кананом [6], который показал, что для каждого k существует язык из $\Sigma_2\mathbf{P} \cap \Pi_2\mathbf{P}$, не имеющий схем размера n^k. Данный результат был усилен до языков из класса $\mathbf{S}_2\mathbf{P}$ [8]. При этом попытки доказать существование такого языка в классе **MA** не привели ни к чему лучше задач из $Promise\mathbf{MA}$ и языков из $\mathbf{MA}/1$ [9] (утверждение было доказано при помощи техники, разработанной в работах Барака, Фортноу и Сантанама [15, 4]).

Препятствие на пути доказательства нижних оценок на языки из **MA** типично для результатов структурной теории сложности (таких как иерархии по времени, существование полной задачи) для семантических классов. В конструкции Сантанама протокол не на всех входах имеет ограниченную вероятность ошибки, требующуюся в определении класса **MA**. Аналогичную проблему решил Первышев для теоремы об иерархии по времени в классе эвристических вероятностных алгоритмов с ограниченной ошибкой; ее же решил Ицыксон при построении Avg**BPP**-полной задачи (вопрос существования Avg**MA**-полной задачи

все еще открыт). Они решили эту проблему, сопоставив каждому входу вероятность, начиная с которой мы примем данный вход.

Вопрос 1. Можно ли доказать нижнюю оценку на эвристическую схемную сложность задачи из эвристического аналога класса **MA**?

Теоремы об иерархиях по времени

Теорема об иерархии по времени для некоторой модели вычислений утверждает, что в данной модели вычислений за большее время можно решить строго большее множество задач. Для детерминированных машин Тьюринга подобная теорема была доказана Хартманисом и Стернсом [1] при помощи диагонализации. Для того чтобы показать, что существует язык, разрешимый за $O(n^3)$ шагов, но не разрешимый за n^2 шагов, можно рассмотреть язык, содержащий строку x тогда и только тогда, когда машина Тьюринга M_x отвергает x за n^2 шагов. Иерархии по времени известны для всех синтаксических моделей вычислений. При этом стандартная диагонализация не работает, если класс не замкнут относительно дополнения (как, например, **NP**). Однако отложенная диагонализация, предложенная Заком [3], работает для всех синтаксических моделей.

Вычислительная модель называется семантической, если невозможно эффективно перечислить все корректные машины. Например, **BPTime**, **RTime** и **ZPTime** являются семантическими. На текущий момент неизвестно никаких точных теорем об иерархии по времени для семантических моделей вычислений. Например, лучший результат об иерархии по времени для вероятностных вычислений с ограниченной ошибкой имеет суперполиномиальный зазор: $\mathbf{BPTime}[n^{\log n}] \subsetneq \mathbf{BPTime}[2^{n^\epsilon}]$ [16].

Первым продвижением в данной теме была теорема об иерархии по времени для вероятностных вычислений с несколькими битами неравномерной подсказки [15, 4], позже была доказана иерархия для всех семантических классов с одним битом подсказки: $\mathbf{BPTime}/1$ [4],

$\mathbf{ZPTime}/1, \mathbf{MATime}/1$ и т.д. [17]. Доказательство иерархии в работах [15, 4] основано на существовании оптимального алгоритма для некоторого **PSpace**-полного языка. Доказательство в работе [17] основано на усложненной версии отложенной диагонализации.

Фортноу и Сантанам также доказали иерархию по времени для эвристических вероятностных алгоритмов с ограниченной ошибкой (такие алгоритмы могут на «малой» доле входов выдавать неправильный ответ или выдавать ответ с неправильной вероятностью). Точнее, они показали, что существует язык L, такой, что (L, U) принадлежит $\mathrm{Heur}_{\frac{1}{n^c}}\mathbf{BPP}$, но (L, U) не принадлежит $\mathrm{Heur}_{\frac{1}{n^c}}\mathbf{BPTime}[n^a]$. Доказательство этого факта также базируется на существовании оптимального алгоритма для **PSpace**-полного языка. Первышев [5] упростил и усилил данную теорему об иерархии для эвристической версии **BPTime**, он доказал, что существует язык L, такой, что $(L, U) \in \mathrm{Heur}_{\epsilon}\mathbf{BPP}$, но $(L, U) \notin \mathrm{Heur}_{\frac{1}{2}-\epsilon}\mathbf{BPTime}[n^a]$. Первышев использовал отложенную диагонализацию против всех вероятностных машин. Отложенная диагонализация требует возможности промоделировать машину на входах с длиной, большей на единицу. Проблема заключается в том, что вероятностная машина может принять вход с произвольной вероятностью, поэтому промоделировать такую машину невозможно при помощи машин с ограниченной ошибкой. Пусть M — это вероятностная машина Тьюринга, которая не соблюдает условие ограниченной ошибки, но нам необходимо промоделировать ее на входе x. Первышев придумал метод, как промоделировать машину эвристически: для каждого входа x мы рассматриваем множество строк $\{y_1, y_2, \ldots, y_N\}$, где N достаточно большое. Для каждой строки y_i запускаем $M(x)$ много раз и вычисляем долю единиц μ_i среди ответов $M(x)$. Алгоритм принимает y_i, если μ_i больше θ_{y_i}, где $\theta_{y_i} = \frac{2}{5} + \frac{i}{5N}$. Заметим, что если $M(x)$ выполняет условие ограниченной ошибки, то с высокой вероятностью ответы для всех y_i будет одинаковыми. А если $M(x)$ не соблюдает условие ограниченной ошибки, то наш алгоритм не соблюдает его только на малой доле строк y_i, точнее, на таких y_i, что θ_{y_i} очень близка к $\Pr[M(x) = 1]$.

Вопрос 2. Можно ли доказать иерархию по времени для эвристических вычислений с параметром ошибки $1 - \epsilon$ вместо $\frac{1}{2} - \epsilon$?

Несложно заметить, что данное доказательство можно переделать в доказательство того, что существует такое полиномиально сэмплируемое распределение, что любое распределение, сэмплируемое за время n^a, имеет статистическое расстояние с ним не меньше $\frac{1}{2} - \epsilon$. В 2013 году Ватсон [18] улучшил этот результат и доказал, что для любых констант a и k существует такое $D \in \mathbf{PSamp}$, что носитель D лежит в $[k]$, и для любого $R \in \mathbf{DSamp}[n^a]$ для бесконечно многих n статистическое расстояние между D_n и R_n не меньше $1 - \frac{1}{k} - \epsilon$.

Вопрос 3. Можно ли формализовать связь между теоремой Ватсона и иерархиями по времени для эвристических распределений?

Иерархии относительно сложности распределений

Как уже было сказано ранее, в теории сложности в среднем рассматриваются задачи в паре с распределением на входах. Задача называется простой в среднем, если она может быть эффективно решена на большой относительно этого распределения доле входов.

Соответственно, вопрос о том, как связана сложность проблемы с распределением на входах, естественен. Известно, что многие трудные задачи можно решить за полиномиальное в среднем время на равномерном распределении на входах: такое доказано для проверки графа на гамильтоновость [11] (заметим, что в классическом случае данная задача **NP**-полна), проверки графов на изоморфизм [19] (в тоже время существуют распределения, для которых неизвестно полиномиального алгоритма [20]).

Также в работе [10] было построено распределение, такое, что любой язык прост в среднем на этом распределении тогда и только тогда, когда он прост в наихудшем случае. Из-за этого в теории сложности вычислений, как правило, рассматривают распределения из каких-то естественных классов распределений, а не произвольные.

Двумя наиболее распространенными такими классами являются класс полиномиально сэмплируемых распределений (распределения являющиеся распределениями результата выполнения какого-то полиномиального вероятностного алгоритма) и класс полиномиально вычислимых распределений (распределений, чья функция распределения вычислима за полиномиальное время). Известно, что первый класс содержит второй, но неизвестно, равны они или нет. При этом доказано, что если существует односторонняя функция, то они не равны [21].

Вопрос 4. Верно ли, что любой «не очень сложный» язык можно упростить «не слишком сильно», усложнив распределение?

Вопрос 5. Верно ли, что существует «не слишком сложное» распределение, которое «сильно» усложняет язык?

Цели, полученные результаты и структура диссертации

Цели работы.

1) Доказать нижние оценки на эвристическую схемную сложность эвристической версии класса **MA**.

2) Найти более простое доказательство эвристической иерархии для вероятностных вычислений с ограниченной ошибкой.

3) Найти более простое доказательство эвристической иерархии для недетерминированных вычислений.

4) Усилить известные условные теоремы об иерархии для вероятностных вычислений с ограниченной ошибкой.

5) Исследовать возможность улучшения параметра ошибки в теоремах об иерархии по времени для эвристических вычислений.

6) Доказать теорему об иерархии для эвристических вероятностных вычислений с ограниченной ошибкой на всех «простых» распределениях.

7) Построить такой язык, что он решается на «простом» распределении за полиномиальное время, но ни на каком «не очень сложном» распределении он не решается «быстрее».

8) Построить такое «не очень сложное» распределение и язык, что этот язык не решается за полиномиальное время на этом распределении, но решается на «простых» распределениях.

Научная новизна. Все результаты диссертации являются новыми.
Теоретическая и практическая ценность. Работа носит теоретический характер. Ее результаты могут быть использованы в классической структурной теории сложности и теории сложности в среднем.
Методы исследований. В работе используются методы теории сложности вычислений.
Положения, выносимые на защиту.

1) Доказана нижняя оценка на эвристическую схемную сложность эвристических полиномиальных протоколов Мерлин–Артур: доказано, что для любого k выполняется $\mathrm{Heur}\mathbf{MA} \not\subseteq \mathrm{Heur}_{1-\delta}\mathbf{Size}[n^k]$.

2) Получено новое, более простое, доказательство того, что $\mathrm{Heur}_{\delta}\mathbf{BPP} \not\subseteq \mathrm{Heur}_{\frac{1}{2}-\epsilon}\mathbf{BPTime}[n^k]$.

3) Получено новое, более простое, доказательство того, что $\mathbf{NP} \not\subseteq \mathrm{Heur}_{\frac{1}{2}-\epsilon}\mathbf{NTime}[n^k]$.

4) Доказано, что если существует односторонняя функция, то $\mathbf{P} \not\subseteq \mathrm{Heur}_{\frac{1}{2}-\epsilon}\mathbf{BPTime}[n^k]$.

5) Доказано, что для любых a, k, δ и ϵ существует $f : \{0,1\}^* \to [a]$, такая, что $(f, U) \in \mathrm{Heur}_{\delta}\mathbf{FBPP}$, но $(f, U) \notin \mathrm{Heur}_{1-\frac{1}{a}-\epsilon}\mathbf{FBPTime}[n^k]$.

6) Доказано, что для любых a, δ и ϵ существует язык L, такой, что $(L, U) \in \mathrm{Heur}_{\delta}\mathbf{BPP}$, но $(L, R) \notin \mathrm{Heur}_{\frac{1}{2}-\epsilon}\mathbf{BPTime}[n^k]$ для любого $R \in \mathbf{DSamp}[n^k]$.

7) Доказано, что для любых $\epsilon > 0$ и $c > 0$ существуют такой язык L и распределение $D \in \mathbf{DSamp}[n^{\log^{\epsilon}(n)}]$, что $(L, D) \notin \mathrm{Heur}_{1-\frac{1}{2^{(\log\log\log n)^c}}}\mathbf{P}$ и для любого $R \in \mathbf{PSamp}$ верно, что $(L, R) \in \mathrm{Heur}_{\frac{1}{2^{(\log\log\log n)^c}}}\mathbf{DTime}[n]$.

8) Доказано, что для любого $a > 0$ существуют такой язык L и распределение $D \in \mathbf{PSamp}$, что $(L, D) \notin \mathrm{Heur}_{\frac{1}{n^a}}\mathbf{P}$ и для любого $R \in \mathbf{DSamp}[n^a]$ верно, что $(L, R) \in \mathrm{Heur}_{O(\frac{1}{n^a})}\mathbf{DTime}[n]$.

Апробация работы. Результаты диссертационной работы были изложены на следующих конференциях и семинарах.

1) Международная конференция “Eighth International Conference on Computability, Complexity and Randomness” (Москва, CCR 2013).

2) Международная конференция “The 10th International Computer Science Symposium in Russia” (Иркутск, CSR 2015).

3) Семинар лаборатории “Exploring limits of computations” (Токио, 2015).

4) Международная конференция “The 26th International Symposium on Algorithms and Computation” (Нагоя, ISAAC 2015).

5) Международная конференция “Problems in Theoretical Computer Science” (Москва, 2015).

6) Международный семинар “Special Semester Program on Complexity Theory” (Санкт-Петербург, 2016).

7) Международная конференция “The 27th International Symposium on Algorithms and Computation” (Сидней, ISAAC 2016).

Публикации. Основные результаты диссертации опубликованы в рецензируемых научных изданиях [22, 23, 24], входящих в список рекомендованных ВАК.

Работы [23] и [24] написаны в соавторстве. В работе [24] идея применения теоремы об иерархии по времени моделирования распределения для доказательства теорем об иерархии для эвристических вычислений была придумана в неразделимом соавторстве с Д.М. Ицыксоном. При этом техническая реализации всех доказательств теорем об иерархии для эвристических вычислений в классах **BPP**, **NP** и **FBPP** принадлежит диссертанту. Условная теорема об иерархии по времени для вероятностных вычислений при условии существования односторонних функций была доказана в неразделимом соавторстве с Д.М. Ицыксоном и Д.О. Соколовым. В работе [24] диссертанту принадлежит доказательство иерархии для слабого варианта трудности задачи в среднем и и доказательство теоремы об иерархии по времени моделирования распределения для бесконечно малых статистических расстояний, при этом постановка задачи принадлежит Д.М. Ицыксону. Неупомянутые результаты работ принадлежат соавторам.

Также результаты диссертации опубликованы в нерецензируемых изданиях [25, 26, 27].

Структура и объем работы. Диссертация состоит из введения, четырех глав и списка литературы. Общий объем диссертации 66 страниц. Список литературы включает 40 наименований на 5 страницах.

В главе 1 вводятся основные обозначения и определяются основные понятия.

В главе 2 доказывается нижняя оценка на схемную сложность эвристического класса Мерлин–Артур.

В главе 3 доказывается связь иерархии по времени для задачи сэмплирования и иерархии по времени для эвристических вероятностных вычислений с ограниченной ошибкой; связь иерархии по времени для задачи недетерминированного сэмплирования и иерархии по времени для эвристических недетерминированного вычислений; доказывается иерар-

хия по времени для вероятностных вычислений с ограниченной ошибкой при условии существования односторонней функции; доказывается иерархия по времени вычисления функций для эвристических вероятностных вычислений с ограниченной ошибкой.

В главе 4 доказывается иерархия по времени сэмплирования распределения для строгой сложности для квазиполиномиально сэмплируемых распределений; доказывается слабая иерархия по времени сэмплирования распределения для слабой сложности для полиномиально сэмплируемых распределений; доказывается иерархия по времени вычисления распределений для полиномиально сэмплируемых распределений.

Глава 1

Основные понятия

В данной работе используются следующие обозначения:

- если L — это язык, то обозначим $L^{=n}$ множество $L \cap \{0,1\}^n$;
- по аналогии с обозначением $O(n)$ будем обозначать $\mathrm{poly}(n)$ множество таких функций $f(n)$, что существует $k > 0$, такое, что $f(n) = O(n^k)$;
- будем называть D ансамблем распределений, если D — это семейство распределений $\{D_n\}_{n=1}^{\infty}$ на битовых строках;
- распределенной задачей будем называть пару (L, D), состоящую из языка L и ансамбля распределений D;
- равномерное распределение на строках длины n обозначим U_n;
- для двух множеств $S_1, S_2 \subseteq \{0,1\}^n$ определим расстояние между ними
$$\Delta(S_1, S_2) = \frac{|(S_1 \cup S_2) \setminus (S_1 \cap S_2)|}{2^n};$$
- для случайных величин χ_1, χ_2, принимающих значение из множества K, определим статистическое расстояние как
$$\Delta(\chi_1, \chi_2) = \max_{S \subseteq K} |\Pr[\chi_1 \in S] - \Pr[\chi_2 \in S]|.$$

1.1 Классы эвристических вычислений

Теперь определим классы распределенных задач, которые будем изучать. Сначала определим эвристические аналоги класса **P**.

Определение 1.1 ([28]). 1) Класс $\mathrm{Heur}_{\delta(n)}\mathbf{DTime}[f(n)]$ состоит из распределенных задач (L, D), таких, что существует детерминированный алгоритм $A(x)$, работающий $O(f(n))$ шагов, и для любого n верно, что $\Pr\limits_{x \leftarrow D_n}[A(x) = L(x)] \geq 1 - \delta(n)$. Также определим $\mathrm{Heur}_{\delta(n)}\mathbf{P} = \bigcup\limits_{k \geq 0} \mathrm{Heur}_{\delta(n)}\mathbf{DTime}[n^k]$.

2) Класс $\mathrm{Avg}_{\delta(n)}\mathbf{DTime}[f(n)]$ состоит из распределенных задач (L, D), таких, что существует детерминированный алгоритм $A(x)$, работающий $O(f(n))$ шагов, для любого n верно, что $\Pr\limits_{x \leftarrow D_n}[A(x) = \perp] \leq \delta(n)$, и для любого x верно, что $A(x) \in \{L(x), \perp\}$. Также определим $\mathrm{Avg}_{\delta(n)}\mathbf{P} = \bigcup\limits_{k \geq 0} \mathrm{Avg}_{\delta(n)}\mathbf{DTime}[n^k]$.

Определение вероятностных классов потребует больших усилий, так как мы также должны разрешить не соблюдать ограничения на вероятность на малой доле входов.

Определение 1.2. 1) Класс $\mathrm{Heur}_{\delta(n)}\mathbf{BPTime}[f(n)]$ состоит из распределенных задач (L, D), таких, что существует вероятностный алгоритм $A(x)$, работающий $O(f(n))$ шагов, и для любого n верно, что $\Pr\limits_{x \leftarrow D_n}[\Pr[A(x) = L(x)] \geq \frac{3}{4}] \geq 1 - \delta(n)$. Также определим $\mathrm{Heur}_{\delta(n)}\mathbf{BPP} = \bigcup\limits_{k \geq 0} \mathrm{Heur}_{\delta(n)}\mathbf{BPTime}[n^k]$.

2) Класс $\mathrm{Avg}_{\delta(n)}\mathbf{BPTime}[f(n)]$ состоит из распределенных задач (L, D), таких, что существует вероятностный алгоритм $A(x)$, работающий $O(f(n))$ шагов, для любого n верно, что $\Pr\limits_{x \leftarrow D_n}[\Pr[A(x) = \perp] \geq \frac{1}{8}] \leq \delta(n)$ и для любого x верно, что $\Pr[A(x) \notin \{L(x), \perp\}] \leq \frac{1}{8}$. Также определим $\mathrm{Avg}_{\delta(n)}\mathbf{BPP} = \bigcup\limits_{k \geq 0} \mathrm{Avg}_{\delta(n)}\mathbf{BPTime}[n^k]$.

Теперь определим равномерные по ошибке классы (определения даются по аналогии с определениями [29]). В этих классах искомая ошибка передается на вход алгоритму.

Определение 1.3. 1) Класс Heur$\mathbf{DTime}[f(n)]$ состоит из распределенных задач (L, D), таких, что существует детерминированный алгоритм $A(x, \delta)$, работающий $O(f(\frac{n}{\delta}))$ шагов, и для любых n и δ верно, что $\Pr\limits_{x \leftarrow D_n}[A(x, \delta) = L(x)] \geq 1 - \delta$. Также определим Heur$\mathbf{P} = \bigcup\limits_{k \geq 0}$ Heur$\mathbf{DTime}[n^k]$.

2) Класс Avg$\mathbf{DTime}[f(n)]$ состоит из распределенных задач (L, D), таких, что существует детерминированный алгоритм A, работающий $O(f(\frac{n}{\delta}))$ шагов, для любых n и δ верно, что $\Pr\limits_{x \leftarrow D_n}[A(x, \delta) = \perp] \leq \delta$ и для любого x верно, что $A(x, \delta) \in \{L(x), \perp\}$. Также определим Avg$\mathbf{P} = \bigcup\limits_{k \geq 0}$ Avg$\mathbf{DTime}[n^k]$.

Аналогично можно определить эвристические аналоги любого классического класса. Однако в связи с тем, что мы в следующей главе докажем нижние оценки на эвристическую схемную сложность полиномиальных в среднем протоколов Мерлин–Артур, определим еще несколько классов.

Определение 1.4. 1) Будем называть булевой схемой от переменных $x_1, \ldots, x_n$ помеченный ориентированный ациклический граф с входящей степенью вершин 0 или 2, в вершинах входящей степени 0 которого стоят переменные $x_1, \ldots, x_n$, а каждая внутренняя вершина помечена бинарной булевой функцией. Значение булевой схемы на подстановке значений $v_1, \ldots, v_n$ переменным $x_1, \ldots, x_n$ — это строка значений, вычесленных в вершинах исходящей степени 0. Значение вычисленное в вершине, помеченной переменной x_i — это v_i, а значение, вычисленное во внутренней вершине — это значение функции, которой помечена вершина, на значениях, вычисленных в двух ее предках. Размером схемы будем называть размер графа. Размером

схемы C будем называть количество вершин в графе и обозначать его $|C|$.

2) Аналогично будем называть вероятностной булевой схемой от переменных $x_1, \ldots, x_n$ со случайными битами $r_1, \ldots, r_m$ булеву схему от переменных $x_1, \ldots, x_n, r_1, \ldots, r_m$.

3) Язык L принадлежит классу $\mathbf{Size}[f(n)]$ тогда и только тогда, когда существует семейство булевых схем C_n, такое, что $|C_n| < f(n)$ и для любого $x \in \{0,1\}^*$ выполняется равенство $C_{|x|}(x) = L(x)$.

4) Язык L принадлежит классу $\mathbf{BPSize}[f(n)]$ тогда и только тогда, когда существует семейство вероятностных булевых схем C_n, такое, что $|C_n| < f(n)$ и для любого $x \in \{0,1\}^*$ выполняется $\Pr[C_{|x|}(x) = L(x)] > \frac{3}{4}$ (вероятность берется по случайным битам схемы).

5) Распределенная задача (L, D) принадлежит классу $\mathrm{Heur}_{\delta(n)}\mathbf{Size}[f(n)]$ тогда и только тогда, когда существует семейство булевых схем C_n, такое, что $|C_n| < f(n)$ и $\Pr\limits_{x \leftarrow D_n}[C_{|x|}(x) = L(x)] \geq 1 - \delta(n)$.

6) Распределенная задача (L, D) принадлежит классу $\mathrm{Heur}_{\delta(n)}\mathbf{BPSize}[f(n)]$ тогда и только тогда, когда существует семейство булевых схем C_n, такое, что $|C_n| < f(n)$ и $\Pr\limits_{x \leftarrow D_n}[\Pr[C_{|x|}(x) = L(x)] > \frac{3}{4}] \geq 1 - \delta(n)$ (вложенная вероятность берется по случайным битам схемы).

Определение 1.5. Распределенная задача (L, D) имеет эвристический протокол Мерлин-Артур (будем обозначать это как $(L, D) \in \mathrm{Heur}\mathbf{MA}$) тогда и только тогда, когда существует вероятностный алгоритм $A(x, y, \delta)$ (здесь x — вход, y — доказательство Мерлина и δ — параметр уверенности), а также семейство множеств $\{S_{n,\delta} \subseteq \{0,1\}^n\}_{\delta \in \mathbb{Q}_+, n \in \mathbb{N}}$ (большие множества, где протокол работает корректно), такие, что для всех n и δ

- $D_n(S_{n,\delta}) \geq 1 - \delta$,
- $A(x, y, \delta)$ работает время $\text{poly}(\frac{n}{\delta})$ и
- для любого x из $S_{n,\delta}$:

$$x \in L \Rightarrow \exists y \ \Pr[A(x, y, \delta) = 1] > \frac{2}{3},$$
$$x \notin L \Rightarrow \forall y \ \Pr[A(x, y, \delta) = 0] > \frac{2}{3}.$$

Распределенная задача (L, D) имеет полиномиальный в среднем протокол Мерлин-Артур (будем обозначать это как $(L, D) \in \text{Avg}\mathbf{MA}$), если в дополнение к предыдущим требованиям выполняется условие, что для всех x не из $S_{n,\delta}$ протокол возвращает с высокой вероятностью либо «отказ», либо верный ответ:

$$x \in L \Rightarrow \exists y \ \Pr[A(x, y, \delta) = 1] > \frac{2}{3} \vee \Pr[A(x, y, \delta) = \perp] > \frac{1}{8},$$
$$x \notin L \Rightarrow \forall y \ \Pr[A(x, y, \delta) = 0] > \frac{2}{3} \vee \Pr[A(x, y, \delta) = \perp] > \frac{1}{8}.$$

Также в главе 3 мы, в том числе, докажем иерархию на эвристический аналог класса **NP**, определим поэтому соответствующий эвристический класс.

Определение 1.6. Класс $\text{Heur}_{\delta(n)}\mathbf{NTime}[f(n)]$ состоит из распределенных задач (L, D), таких, что существует недетерминированный алгоритм $A(x)$, работающий $O(f(n))$ шагов, и для любого n верно, что $\Pr\limits_{x \leftarrow D_n}[A(x) = L(x)] \geq 1 - \delta(n)$. Также определим $\text{Heur}_{\delta(n)}\mathbf{NP} = \bigcup\limits_{k \geq 0} \text{Heur}_{\delta(n)}\mathbf{NTime}[n^k]$.

1.2 Классы распределений

В дальнейшем мы будем считать, что во всех ансамблях распределений D распределение D_n имеет носителем подмножество $\{0, 1\}^n$.

В главе 4 мы будем исследовать связь сложности задачи со сложностью распределения. Для этого необходимо определить классы сложности распределений. Следующие два класса сложности используются чаще всего.

Определение 1.7. 1) Будем называть ансамбль распределений D сэмплируемым за время $t(n)$, если существует вероятностный алгоритм S, такой, что на входе 1^n он работает $O(t(n))$ шагов, и $S(1^n)$ одинаково распределено с D_n. Множество всех ансамблей, сэмплируемых за время $t(n)$, обозначим $\mathbf{DSamp}[t(n)]$. Также рассмотрим множество $\mathbf{PSamp} = \bigcup\limits_{c>0} \mathbf{DSamp}[n^c]$ всех ансамблей, сэмплируемых за полиномиальное время.

2) Будем называть ансамбль распределений D вычислимым за время $t(n)$, если для всех n вероятности всех строк (относительно D_n) — рациональные числа, и существует алгоритм $A(x, n)$, работающий время $O(t(|x|))$ и вычисляющий функцию распределения D_n (т.е. $\sum\limits_{y \le x} D_n(x)$, где $\le$ — это лексикографический порядок). Множество всех ансамблей, вычислимых за время $t(n)$, обозначим $\mathbf{Comp}[t(n)]$. Как и в прошлый раз, определим множество $\mathbf{PComp} = \bigcup\limits_{c>0} \mathbf{Comp}[n^c]$ всех полиномиально вычислимых распределений.

Глава 2

Схемная сложность Avg**MA**

В данной главе доказывается нижняя оценка на схемную сложность языков, принимаемых за полиномиальное в среднем время протоколами Мерлина-Артура. Именно, докажем, что существует положительная константа a, такая, что для любого k существует задача из Avg**MA**, не принадлежащая классу $\mathrm{Heur}_{\frac{1}{n^a}}\mathbf{Size}[n^k]$.

Основная идея доказательства заключается в том, чтобы взять сложный, самоисправляемый и контролируемый язык, а затем превратить его в язык из Avg**MA** так, чтобы он остался достаточно сложным в среднем. Свойство самопроверяемости нужно для того, чтобы превратить сложный в наихудшем случае язык в сложный в среднем язык, а свойство проверяемости ответов — для построения протоколов Мерлин–Артур.

В секции 2.1 даются формальные определения необходимых свойств и излагаются необходимые результаты, доказаные ранее. В секции 2.2 дается формальное определение эвристических сведений и доказываются различные их свойства. В секции 2.3 доказываются нижняя оценка на схемную сложность и верхняя оценка на время работы протоколов. В секции 2.4 рассматриваются возможности для усиления полученых результатов и демонстрируются препятствия на пути этих усилений.

Результаты этой главы опубликованы в [22].

2.1 Определения

Определение 2.1 ([30])**.** Пусть b — это рациональное положительное число. Язык L является b-самоисправляемым, если существует вероятностный оракульный алгоритм A (корректор для L), такой, что для всех языков L', если $\Delta(L^{=n}, L'^{=n}) \leq \frac{1}{n^b}$, то $\forall x \in \{0,1\}^n$, $\Pr[A^{L'^{=n}}(x) = L(x)] > \frac{3}{4}$. Мы будем называть язык самоисправляемым, если он b-самоисправляемый для какой-то константы b.

Неформально это определение говорит, что если у нас есть оракульный доступ к языку, достаточно близкому к L, то мы можем вероятностно разрешить язык L за полиномиальное время.

Определение 2.2 ([30])**.** Язык L является f-контролируемым, если существует полиномиальный по времени вероятностный оракульный алгоритм M (контролер L), такой, что для всех $x \in \{0,1\}^n$ верно:

- если $x \in L$, то $\Pr[M^{L^{=f(n)}}(x) = 1] = 1$ *(абсолютная полнота)*;
- если $x \notin L$, то для любого L' выполняется $\Pr[M^{L'^{=f(n)}}(x) = 1] < \frac{1}{2^n}$ *(корректность)*.

Также нам потребуется следующий несложный факт про схемы, являющийся простейшим усилением теоремы Адлемана ($\mathbf{BPP} \subseteq \mathbf{P}/\mathrm{poly}$).

Лемма 2.1 ([31])**.** Для любых функций $\delta\colon \mathbb{N} \to [0;1]$ и $t\colon \mathbb{N} \to \mathbb{N}$,

$$\mathrm{Heur}_{\delta(n)}\mathbf{BPSize}[t(n)] \subseteq \mathrm{Heur}_{\delta(n)}\mathbf{Size}[poly(n)t(n)].$$

Еще нетрудно заметить, что самоиправляемость позволяет переходить от сложности в среднем к сложности в наихудшем случае.

Лемма 2.2. Если язык L является a-самоисправляемым и $L \in \mathrm{Heur}_{\frac{1}{n^a}}\mathbf{Size}[f(n)]$, то $L \in \mathbf{Size}[f(n)poly(n)]$.

Доказательство. Преобразуем стандартным образом машину Тьюринга, вычисляющую корректор для L, в вероятностную схему B. Предположим, что вместо оракульных запросов B использует схему, разрешающую L. Таким образом, $L \in \mathbf{BPSize}[f(n)\mathrm{poly}(n)]$ и $L \in \mathbf{Size}[f(n)\mathrm{poly}(n)]$ по лемме 2.1. □

Для первой части доказательства нижней оценки нам потребуется язык из **PSpace** с высокой схемной сложностью (благодаря коллапсу он также окажется языком из **MA**).

В статье [9] Сантанам заметил, что доказательство из [32] можно перенести с языков, распознаваемых за экспоненциальное время, на языки распознаваемые с полиномиальной памятью. Достаточно следующей формулировки этого результата.

Лемма 2.3 ([9])**.** Существует константа $a > 0$, такая, что для всех k выполнено условие $\mathbf{PSpace} \not\subseteq \mathrm{Heur}_{\frac{1}{n^a}}\mathbf{Size}[n^k]$.

Для второго случая нам потребуется **PSpace**-полный язык с хорошими свойствами.

Лемма 2.4 ([33])**.** Существует **PSpace**-полный самокорректируемый n-контролируемый язык.

2.2 Эвристические сведения

Для доказательства нам потребуются сведения, похожие на вероятностные эвристические сведения [34], но несколько отличающиеся: нам не нужны полиномиальная вычислимость, дизъюнктность образов для разных случайных строк и равномерность распределений для входов.

Определение 2.3. Пусть L и L' — языки, а $c\colon \mathbb{N} \to \mathbb{R}$ — функция, тогда семейство функций $f_n\colon \{0,1\}^n \times \{0,1\}^{y_n} \to \{0,1\}^{m_n}$, где $m_n \geq n$, называется $c(n)$-эвристическим сведением языка L к L', если

$$\forall x \in \{0,1\}^n \ \forall r \in \{0,1\}^{y_n} \ L'(f_n(x,r)) = L(x), \qquad (\textit{корректность})$$

и

$$\forall n\ \forall S \subseteq \{0,1\}^n \times \{0,1\}^{y_n}\ \ \frac{|f_n(S)|}{2^{m_n}} > c(n)\frac{|S|}{2^{n+y_n}} \qquad (\textit{доминирование})$$

Лемма 2.5. Для любого $a > 0$, если $L' \in \mathrm{Heur}_{1-\frac{1}{n^{a+l+1}}}\mathbf{Size}[p(n)]$ и существует $\frac{d}{n^l}$-эвристическое сведение языка L к языку L', вычислимое схемой размера $q(n)$, то $L \in \mathrm{Heur}_{1-\frac{1}{n^a}}\mathbf{Size}[(p(m_n) + q(n))poly(n)]$ (где m_n — как в определении 2.3 и d — это константа).

Доказательство. Пусть D_n — это схема размера $q(n)$, вычисляющая сведение f_n, и пусть C_n — схема, распознающая $L'^{=n}$ с ошибкой $\frac{1}{n^{a+l+1}}$. По лемме 2.1 достаточно доказать, что для достаточно больших n $\Pr_x[\Pr_r[C(D(x,r)) \neq L(x)] \geq \frac{1}{4}] < \frac{1}{n^a}$ (отсюда и далее мы будем опускать нижние индексы для C и D). Предположим обратное. Тогда $\frac{|\{(x,r)|C(D(x,r))\neq L(x)\}|}{2^{n+y_n}} \geq \frac{1}{4n^a}$. Однако, используя корректность и доминирование, мы получаем

$$\frac{|\{y|C(y) \neq L'(y)\}|}{2^{m_n}} \geq \quad \frac{|\{D(x,r)|C(D(x,r)) \neq L'(D(x,r))\}|}{2^{m_n}} =$$
(воспользовавшись корректностью)
$$\frac{|\{D(x,r)|C(D(x,r)) \neq L(x))\}|}{2^{m_n}} \geq$$
(воспользовавшись доминированием)
$$\frac{d}{n^l}\frac{|\{(x,r)|C(D(x,r)) \neq L(x)\}|}{2^{n+y_n}} \geq$$
$$\frac{d}{4n^{a+l}} \geq \frac{1}{n^{a+l+1}} \geq \frac{1}{m_n^{a+l+1}},$$

что противоречит предположению про C. □

2.3 Нижние оценки

Для того, чтобы работать со сложностью в среднем, необходимо аккуратно работать с вероятностями входов. Поэтому нам потребуется правильно кодировать тройки входов, не сильно увеличивая длину.

Определение 2.4. Будем обозначать как $\langle \cdot, \cdot, \cdot \rangle$ функцию из $\{0,1\}^n \times \{0,1\}^{g(n)} \times \{0,1\}^{y_n}$ в $\{0,1\}^{2\lceil \log(n) \rceil + n + g(n) + y_n + 2}$, определенную следующим образом: $\langle x, p, z \rangle = \widehat{n}11xpz$, где $\widehat{n_1 n_2 \dots} = n_1 0 n_2 0 \dots$ и g — это многочлен.

Замечание 2.1. Несложно заметить, что

$$\frac{|\{\langle x, p, z \rangle | x \in \{0,1\}^n, p \in \{0,1\}^{g(n)}, z \in \{0,1\}^{y_n}\}|}{|\{0,1\}^{2\lceil \log(n) \rceil + n + g(n) + y_n + 2}|} = \frac{1}{4n^2}$$

В следующих леммах мы построим язык из Avg**MA**.

Лемма 2.6. Пусть g и f — фиксированные многочлены, k — целое число и A — вероятностный полиномиальный по времени алгоритм с параметрами x, C и z, использующий $g(|x|)$ случайных битов. Тогда язык

$$L = \{\langle x, p, z \rangle |\ |p| = g(|x|), \exists C\ \Pr[A(x, C, z) = 1] \geq 0.p \wedge |C| < f(|x|, |z|)\}$$

принадлежит Avg**MA**, а следовательно и Heur**MA**, где $0.p$ означает рациональное число меньшее 1, запись в двоичной системе счисления которого совпадает с $0.p_1 p_2 \dots p_{|p|}$.

Доказательство. Рассмотрим следующий протокол, демонстрирующий, что $L \in$ Avg**MA**.

1) Вначале Артур проверяет, что код кодирует корректную тройку $\langle x, p, z \rangle$, и если нет, то возвращает 0.

2) Артур получает C от Мерлина.
   Если $|C| > f(|x|, |z|)$ или $|p| \neq g(|x|)$, то возвращает 0.

3) Если $\delta > \frac{1}{2^{g(|x|)}}$, то Артур

   (a) запускает $\frac{16}{\delta^2}$ раз алгоритм $A(x, C, z)$ и вычисляет долю $\bar{q}$ случаев, когда A принимает;

   (b) если $\bar{q} \geq 0.p + \frac{\delta}{4}$, то возвращает 1;

   (c) если $\bar{q} \leq 0.p - \frac{\delta}{4}$, то возвращает 0;

(d) в противном случае возвращает $\perp$.

4) Если $\delta \leq \frac{1}{2^{g(|x|)}}$, то Артур

(a) вычисляет $q = \Pr[A(x, C, z) = 1]$, запуская $A(x, C, z)$ на всех возможных случайных битах.

(b) если $q \geq 0.p$, то возвращает 1, а иначе 0.

Сначала покажем, что размер множества $S_{n,\delta}$, где протокол вычисляет ответ правильно, достаточно большое. Если $\delta \leq \frac{1}{2^{g(|x|)}}$, то протокол, по построению, на всех входах работает правильно. Если же $\delta > \frac{1}{2^{g(|x|)}}$, то положим $S_{n,\delta} = \{0,1\}^n \setminus \{\langle x, p, z\rangle \in \{0,1\}^n |\ |q(x,z) - 0.p| \leq \frac{\delta}{2}\}$ (заметим, что $\frac{|S_{n,\delta}|}{2^n} \geq 1 - \delta$, так как для фиксированых x и z доля подходящих p равна $1 - \delta$), где $q(x, z) = \max_C \Pr[A(x, C, z) = 1]$. Если $x \in S_{n,\delta}$, рассмотрим два случая:

1) $\boxed{\langle x, p, z\rangle \in L}$: если Мерлин присылает C, такое, что $\Pr[A(x, C, z) = 1] > 0.p + \frac{\delta}{2}$, то по оценке Чернова Артур отвергает с вероятностью не больше $\Pr[\bar{q} < 0.p - \frac{\delta}{4}] < 2e^{-2\frac{\delta^2}{4}\frac{16}{\delta^2}} = 2e^{-8} < \frac{1}{3}$;

2) $\boxed{\langle x, p, z\rangle \notin L}$: для любого C верно, что $\Pr[A(x, C, Z)] < 0.p - \frac{\delta}{2}$, значит, по оценке Чернова Артур принимает с вероятностью не больше $\Pr[\bar{q} > 0.p + \frac{\delta}{4}] < 2e^{-2\frac{\delta^2}{4}\frac{16}{\delta^2}} = 2e^{-8} < \frac{1}{3}$.

В противном случае, если $x \notin S_{n,\delta}$, то снова рассмотрим два случая:

1) $\boxed{\langle x, p, z\rangle \in L}$: если Мерлин присылает C, такое, что $\Pr[A(x, C, z) = 1] > 0.p$, то по оценке Чернова Артур отвергает с вероятностью $\Pr[\bar{q} \leq 0.p - \frac{\delta}{4}] < 2e^{-8} < \frac{1}{6}$, значит, если Артур принимает с вероятностью $\Pr[\bar{q} \geq 0.p + \frac{\delta}{4}] \leq \frac{2}{3}$, то Артур возвращает $\perp$ с вероятностью $\Pr[|\bar{q} - 0.p| < \frac{\delta}{4}] > \frac{1}{6}$;

2) $\boxed{\langle x, p, z\rangle \notin L}$: для любого C верно, что $\Pr[A(x, C, z) = 1] \leq 0.p$. Значит, по оценке Чернова Артур принимает с вероятностью $\Pr[\bar{q} \geq$

$0.p + \frac{\delta}{4}] < 2e^{-8} < \frac{1}{6}$; таким образом, если Артур отвергает с вероятностью $\Pr[\bar{q} \geq 0.p - \frac{\delta}{4}] \leq \frac{2}{3}$, то Артур возвращает $\perp$ с вероятностью $\Pr[|\bar{q} - 0.p| < \frac{\delta}{4}] > \frac{1}{6}$.

□

Лемма 2.7. Если **PSpace** $\subseteq$ **P**/poly, то существует константа $a > 0$, такая, что для любого k выполнено **MA** $\not\subseteq \mathrm{Heur}_{\frac{1}{n^a}}$**Size**$[n^k]$.

Доказательство. Широко известно (доказательство можно найти, например, в [32, теорема 8.22]), что **PSpace** $\subseteq$ **P**/poly влечет **MA** = **PSpace** (потому, что прувер в интерактивном протоколе для `QBF` [35] может быть заменен схемой, присылаемой Мерлином). Тогда лемма 2.3 дает нам искомый язык из **MA** с большой эвристической сложностью на равномерном распределении. □

Теорема 2.1. Существует константа $a > 0$, такая, что для любого $k \in \mathbb{Q}_+$ выполняется

$$\mathrm{Avg}\mathbf{MA} \not\subseteq \mathrm{Heur}_{\frac{1}{n^a}}\mathbf{Size}[n^k].$$

Доказательство. Пусть L — это язык из леммы 2.4, и M — это его контролер (определение 2.2). Зафиксируем $k \in \mathbb{Q}_+$. Пусть $g(n)$ — такой полином, что M использует меньше чем $g(n)$ случайных битов.

Если $L \in$ **P**/poly, то **PSpace** $\subseteq$ **P**/poly, а значит, по лемме 2.6 мы доказали, что $\mathrm{Avg}\mathbf{MA} \not\subseteq \mathrm{Heur}_{\frac{1}{n^a}}\mathbf{Size}[n^k]$.

Предположим, что $L \notin$ **P**/poly. Мы добавим к нему паддинг так, чтобы его схемная сложность была полиномиальная, но больше n^k. Также добавим ко входу число, которое будем использовать как пороговое значение для контролера. Говоря формально, рассмотрим язык

$$L' = \{\langle x, p, z\rangle |\ |p| = g(|x|), \exists \text{ схема } C\ \Pr[M^C(x) = 1] \geq 0.p \wedge |C| < (|z| + 1)^{k+1}\}.$$

Замечание. Несложно увидеть, что если убрать требование на размер схемы C, взять $p = 2^{g(|x|)}$ и выбрать в качестве C схему для L, мы получим L с паддингом (благодаря абсолютной полноте контролера).

По лемме 2.6 очевидно, что $L' \in \mathrm{Avg}\mathbf{MA}$.

Теперь осталось показать, что $L' \notin \mathrm{Heur}_{1-\frac{1}{n^a}}\mathbf{Size}[n^k]$.

Пусть b — такая константа, что L является b-самоисправляемым и $a = b + 3$. Докажем от противного. Предположим, что $L' \in \mathrm{Heur}_{\frac{1}{n^a}}\mathbf{Size}[n^k]$. Обозначим за $s(n)$ схемную сложность L. Пусть y_n — это такая последовательность, что $y_n^{k+1} \leq s(n) < (y_n + 1)^{k+1}$. Рассмотрим отображение $f_n \colon \{0,1\}^n \times \{0,1\}^{g(n)+y_n-1} \to \{0,1\}^{2\log(n)+2+n+g(n)+y_n}$, такое, что $f_n(x, r_1 r_2) = \langle x, 1r_1, r_2\rangle$, где $|r_1| = g(|x|) - 1$ и $|r_2| = y_n$. Докажем, что f_n является $\frac{1}{8n^2}$-эвристическим сведением L к L'.

- Свойство доминирования выполняется потому, что все тройки образуют долю $\frac{1}{4n^2}$ всех строк; в тройке же мы фиксируем только первый бит второго элемента.

- Сведение корректно: в случае $x \in L$ существует схема для L с размером между y_n^{k+1} и $(y_n + 1)^{k+1}$, а значит, благодаря абсолютной полноте контролера для любого r_1 верно, что $\langle x, 1r_1, r_2\rangle \in L'$.

  Для $x \notin L$ нет ни одной схемы, которая заставит принимать контролер с вероятностью большей чем $\frac{1}{2^n}$ (заметим, что зафиксировав первый бит во втором элементе тройки мы потребовали, что вероятность должна быть больше $\frac{1}{2}$). В итоге $\langle x, 1r_1, r_2\rangle \notin L'$.

Лемма 2.5 при $l = 2$ и $d = \frac{1}{8}$ влечет $L \in \mathrm{Heur}_{1-\frac{1}{n^b}}\mathbf{Size}[((y_n + g(n) + 2\log(n) + n + 2)^k + (n + g(n) + y_n + 2\log(n) + 2))\mathrm{poly}(n)]$. Так как L — b-самоисправляемый, по лемме 2.2 выполняется $L \in \mathbf{Size}[(n+y_n+g(n)+2\log(n)+2)^k\mathrm{poly}(n)] \subseteq \mathbf{Size}[y_n^k\mathrm{poly}(n)]$. Значит, $y_n^{k+1} < s(n) < y_n^k\mathrm{poly}(n)$, а значит y_n растет полиномиально. Таким образом, $L \in \mathbf{P}/\mathrm{poly}$, что противоречит нашему предположению. □

2.4 Heur**AM**, Heur**NP** и препятствия к доказательству нижних оценок

Ицыксон и Соколов [36] доказали, что языки из **AM** могут быть дерандомизованы добавлением паддинга и переходом к эвристической сложности.

Пусть q — многочлен, введем обозначение $pad_q(L) = \{(x, r)|x \in L, r \in \{0,1\}^*, |r| \geq q(|x|)\}$, где L — некоторый язык.

Определение 2.5. • Язык L принадлежит Heur**NP** тогда и только тогда, когда существует алгоритм $A(x, y, \delta)$ (здесь x — это вход, y — это подсказка и δ — это параметр корректности) и семейство множеств $\{S_{n,\delta} \subseteq \{0,1\}^n\}_{\delta \in \mathbb{Q}_+, n \in \mathbb{N}}$ (большие множества, где алгоритм работает корректно), такие, что для любых n и δ

– $U_n(S_{n,\delta}) \geq 1 - \delta$,

– $A(x, y, \delta)$ работает $\mathrm{poly}(\frac{n}{\delta})$ шагов, и

– для любого x из $S_{n,\delta}$:

$$x \in L \Rightarrow \exists y\ A(x, y, \delta) = 1,$$
$$x \notin L \Rightarrow \forall y\ A(x, y, \delta) = 0.$$

• Язык L принадлежит $\mathrm{Heur}_{\delta(n)}$**NP** тогда и только тогда, когда существует алгоритм $A(x, y)$ (здесь x — это вход и y — это подсказка) и семейство множеств $\{S_n \subseteq \{0,1\}^n\}_{n \in \mathbb{N}}$ (большие множества, где алгоритм работает корректно), такие, что для любого n

– $U_n(S_n) \geq 1 - \delta(n)$,

– $A(x, y)$ работает $\mathrm{poly}(n)$ шагов, и

– для любого x из S_n:

$$x \in L \Rightarrow \exists y\ A(x, y) = 1,$$
$$x \notin L \Rightarrow \forall y\ A(x, y) = 0.$$

Теорема 2.2 ([36])**.** Для любого языка $L \in \mathbf{AM}$ существует многочлен g, такой, что $pad_g(L) \in \mathrm{Heur}\mathbf{NP}$.

Несложно показать, что доказательство этой теоремы останется верным, даже если рассмотреть $L \in \mathrm{Heur}\mathbf{AM}$.

Теорема 2.3. Для любого языка из $L \in \mathrm{Heur}\mathbf{AM}$ и положительного a существует многочлен g, такой, что $pad_g(L) \in \mathrm{Heur}_{\frac{1}{n^a}}\mathbf{NP}$. Более того, если эвристический протокол Артур-Мерлин для языка L использует $t(n)$ случайных битов для параметра корректности $\delta = \frac{1}{n^a}$, то $g(n) \leq \mathrm{poly}(t(n))$.

В данной теореме пришлось изменить $\mathrm{Heur}\mathbf{NP}$ на $\mathrm{Heur}_{\frac{1}{n^a}}\mathbf{NP}$, так как число случайных битов зависит от параметра корректности.

Доказанная выше нижняя оценка на $\mathrm{Heur}\mathbf{MA}$ влечет нижнюю оценку на $\mathrm{Heur}\mathbf{AM}$, так как $\mathrm{Heur}\mathbf{MA} \subseteq \mathrm{Heur}\mathbf{AM}$ и кажется, что теорема 2.3 дает направление для доказательства нижних оценок на $\mathrm{Heur}_{\delta}\mathbf{NP}$, что повлечет далеко идущие последствия.

Гипотеза 2.1. Существует многочлен p, такой, что если существует полиномиальный эвристический протокол Мерлин-Артур для L, такой, что на входах длины n и с параметром корректности δ Артур использует $q(n, \frac{1}{\delta})$ случайных битов (q — это многочлен), то существует эвристический протокол Артур–Мерлин, использующий $p(q(n, \delta), n, \delta)$ случайных битов.

Стоит отметить, что многочлен p в этом предположении не зависит от протокола, в то время как в стандартном доказательстве включения $\mathbf{MA}$ в $\mathbf{AM}$ этот многочлен зависит от длины доказательства Мерлина.

Теорема 2.4. Если гипотеза 2.1 верна, то существует $a > 0$, такое, что для любого $k \in \mathbb{Q}_+$,

$$\mathrm{Heur}\mathbf{NP} \not\subseteq \mathrm{Heur}_{\frac{1}{n^a}}\mathbf{Size}[n^k].$$

Однако, несложно увидеть, что:

Теорема 2.5. Если для некоторых $a > 0$ и $k \in \mathbb{Q}_+$ выполняется Heur$\mathbf{NP} \not\subseteq$ Heur$_{\frac{1}{n^a}}\mathbf{Size}[n^k]$, то $\mathbf{NP} \not\subseteq \mathbf{Size}[n^k]$.

Доказательство. Рассмотрим язык L, такой, что $L \in$ Heur$\mathbf{NP}$ и $L \notin$ Heur$_{1-\frac{1}{n^a}}\mathbf{Size}[n^k]$. Пусть M — недетерминированная машина, разрешающая L в Heur$\mathbf{NP}$. Определим $L' = \{x \in \{0,1\}^* | \exists y \in \{0,1\}^*\, M(x, y, \frac{1}{n^a}) = 1\}$.

Заметим, что $\Delta(L, L') \leq \frac{1}{n^a}$, $L \notin$ Heur$_{\frac{1}{n^a}}\mathbf{Size}[n^k]$ и $L \in \mathbf{NP}$. Таким образом, $L' \notin \mathbf{Size}[n^k]$, но $L' \in \mathbf{NP}$. □

Таким образом, получаем следствие:

Следствие 2.1. Если предположение 2.1 выполнено, то $\mathbf{NP} \not\subseteq \mathbf{Size}[n^k]$ для любого $k \in \mathbb{Q}_+$.

Из работы [37] известно, что $Promise\mathbf{MA} \not\subseteq \mathbf{Size}[n^k]$ алгебраизуется, а $\mathbf{NP} \not\subseteq \mathbf{Size}[n^k]$ не алгебраизуется. В то же время, неизвестно, алгебраизуется ли Heur$\mathbf{MA} \not\subseteq$ Heur$_{\frac{1}{n^a}}\mathbf{Size}[n^k]$. Таким образом, было бы интересно найти препятствия на пути усиления результата про Heur$\mathbf{MA}$ до результата про Heur$\mathbf{NP}$. В остатке этой главы мы докажем, что гипотеза 2.1 в некотором смысле не релятивизуется.

Все известные доказательства вложения $\mathbf{MA} \subseteq \mathbf{AM}$ используют амплификацию (заключающуюся в повторении протокола) вероятности успеха. Получающийся протокол использует число случайных битов, пропорциональное размеру доказательства Мерлина. Мы докажем, что для моделирования $\mathbf{MA}$-протокола $\mathbf{AM}$-протоколами необходимо итерировать эти протоколы.

Определение 2.6. Пусть $f\colon \{0,1\}^* \to \{0,1\}$, $k \in \mathbb{N}$. Будем говорить, что $L \in \mathbf{AM}^{f[k]}$ тогда и только тогда, когда существует оракульный вероятностный алгоритм $A^\bullet(x, y, r)$ (Артур), такой, что

- для любого x, C и r, Артур делает не больше k запросов к оракулу,

- для любого $x \in \{0,1\}^n$

$$x \in L \Rightarrow \Pr_r[\exists C\ A^f(x,C,r) = 1] = 1$$
$$x \notin L \Rightarrow \Pr_r[\exists C\ A^f(x,C,r) = 1] \leq \frac{1}{2},$$

- $A(x,C,r)$ работает $\mathrm{poly}(|x|)$ шагов.

Определение 2.7. Пусть $f\colon \{0,1\}^* \to \{0,1\}$, $k \in \mathbb{N}$. Будем говорить, что $L \in \mathbf{MA}^{f[k]}$ тогда и только тогда, когда существует оракульный вероятностный алгоритм $A^\bullet(x,y,r)$ (Артур), такой, что

- для любого x, C и r, Артур делает не больше k запросов к оракулу,

- для любого $x \in \{0,1\}^n$

$$x \in L \Rightarrow \exists C\ \Pr_r[A^f(x,C,r) = 1] = 1$$
$$x \notin L \Rightarrow \forall C\ \Pr_r[A^f(x,C,r) = 1] \leq \frac{1}{2}$$

- $A(x,C,r)$ работает $\mathrm{poly}(|x|)$ шагов.

Теорема 2.6. Существует f, такая, что $\mathbf{MA}^{f[1]} \not\subseteq \mathbf{AM}^{f[1]}$.

Так как мы не можем итерировать протоколы из $\mathbf{AM}^{f[1]}$, оставаясь в $\mathbf{AM}^{f[1]}$, эта теорема непрямо указывает на то, что для моделирования итерировать необходимо.

Пусть $f\colon \mathbb{N} \times \{0,1\} \times \{0,1\} \to \{0,1\}$. Будем обозначать L_f язык $\{0^n | \exists y \in \{0,1\}\ f(n,y,0) = f(n,y,1)\}$.

Лемма 2.8. Для любого отображения f язык $L_f \in \mathbf{MA}^{f[1]}$.

Доказательство. Рассмотрим следующий алгоритм для Артура: Артур получает y и z от Мерлина, берет случайную строку $r \in \{0,1\}$ и проверяет, что $z = f(n,y,r)$. Очевидно, что если $0^n \in L_f$, то вероятность ошибки 0, и если $0^n \notin L_f$, то вероятность ошибки не больше $\frac{1}{2}$. □

Лемма 2.9. Существует отображение f, такое, что $L_f \notin \mathbf{AM}^{f[1]}$.

Доказательство. Перечислим все полиномиальные оракульные алгоритмы $A_i^\bullet$, которые могут использоваться как Артур. Предположим, что $A_i^\bullet$ имеет полиномиальный будильник p_i. Пусть $n_1 = 1$ и $n_{i+1} = p_i(n_i)+1$. Заметим, что $A_i^\bullet$ на входах длины n_i делает запросы ко входам с длиной меньше чем n_{i+1}. Будем говорить, что $A_i^\bullet$ не имеет ложноотрицательных ответов на функции f, если

$$0^{n_i} \in L_f \Rightarrow \exists C \ \Pr_r[A_i^f(0^{n_i}, C, r) = 1] = 1,$$

и не имеет ложноположительных ответов на функции f, если

$$0^{n_i} \notin L_f \Rightarrow \forall C \ \Pr_r[A_i^f(0^{n_i}, C, r) = 1] \leq \frac{1}{2}.$$

Для любого $n \neq n_i$, любых y и b мы определим $f(n, y, b) = b$ (тем самым $L_f \subseteq \{0^{n_i} | i \in \mathbb{N}\}$). Покажем, что существует f, такое, что для любого n алгоритм $A_n^\bullet$ работает некорректно на f (не соблюдает ограничения на вероятность или дает некорректные ответы). Строить f будем последовательно для каждой длины. Точнее, мы построим последовательность функций $f_i \colon \mathbb{N} \times \{0,1\} \times \{0,1\} \to \{0,1\}$, таких что $f_{-1}(n, y, b) = b$, а для каждого $i \geq 0$ и $n < n_i$ выполнено, что $f_i(n, y, b) = f_{i-1}(n, y, b)$ и $A_i^\bullet$ имеет ложноположительные или ложноотрицательные ответы на f_i. Также для каждого $i \geq 0$ и $n = n_i$ определим $f(n, y, b) = f_i(n, y, b)$.

Докажем существование f_i от противного. Предположим, что для любой функции $h \colon \mathbb{N} \times \{0,1\} \times \{0,1\} \to \{0,1\}$ алгоритм $A_i^\bullet$ не имеет ложноположительных или ложноотрицательных ответов на h, если для любого $i \geq 0$, $n < n_i$, $y \in \{0,1\}$ и $b \in \{0,1\}$ выполняется $h(n, y, b) = f_{i-1}(n, y, b)$.

Для каждого $y \in \{0,1\}$ рассмотрим g_y, такое, что для любого z выполняется $g_y(n_i, y, z) = 0$, $g_y(n_i, 1-y, z) = z$ и $g_y(n, y, b) = f_{i-1}(n, y, b)$ для всех $n \neq n_i$ и $b \in \{0,1\}$.

В связи с тем, что $A_i^\bullet$ не имеет ложноотрицательных ответов на g_y, для всех r, y и i существует такое C_y^r, что $A_i^{g_y}(0^{n_i}, C_y^r, r) = 1$. Заметим, что для всех j и y алгоритм $A_i^\bullet(0^{n_i}, C_y^r, r)$ спрашивает оракул с вероятностью не меньше $\frac{1}{2}$. В противном случае Артур имел бы ложноположи-

тельные ответы (с сертификатом C_y^r) на функции g (совпадающей с g_y везде, кроме точки (n_i, y, j)), такой, что

$$\begin{aligned} g(n_i, t, z) &= z \text{ для } t \neq y, \\ g(n_i, y, j) &= 1, \\ g(n_i, y, 1-j) &= 0, \end{aligned}$$

так как $\Pr[A_i^g(0^{n_i}, C_y^r, r) = 1] \geq \Pr[A^\bullet \text{ не спрашивает } (0^{n_i}, y, j)] > \frac{1}{2}$ (данное неравенство верно из-за того, что если A не спросит точку $(0^{n_i}, y, j)$, то будет вести себя как на функции g_y), что противоречит $0^{n_i} \notin L_g$.

Пусть $R_{y,j} = \{r | A_i^\bullet(0^{n_i}, C_y^r, r) \text{ спрашивает } (0^{n_i}, y, j)\}$. Предыдущий аргумент демонстрирует, что для любых y и j выполняется $\Pr[R_{y,j}] = \frac{1}{2}$. Теперь докажем от противного, что $R_{0,j_0} = R_{1,j_1}$ для всех $j_0, j_1 \in \{0,1\}$. Предположим, что это не так. Рассмотрим g, такое, что

$$\begin{aligned} g(n_i, 0, j_0) &= 0, \\ g(n_i, 0, 1-j_0) &= 1, \\ g(n_i, 1, j_1) &= 0, \\ g(n_i, 1, 1-j_1) &= 1. \end{aligned}$$

На этой функции Артур имеет ложноположительный ответ: если $r \in R_{y,j_y}$, Мерлин может послать C_y^r, и поэтому

$$\Pr[\exists C\ A^g(0^{n_i}, C, r) = 1] \geq \Pr_r[\exists y\ A^g(0^{n_i}, C_y^r, r) = 1] \geq \Pr[R_{0,j_0} \cup R_{1,j_1}] > \frac{1}{2}.$$

Получили противоречие.

Значит, $R_{0,0} = R_{1,0} = R_{0,1}$, но это невозможно, поскольку для каждого y $R_{y,0} \cap R_{y,1} = \emptyset$.

В итоге мы доказали, что существует отображение f_i, такое, что $A_i^{f_i}$ работает некорректно на длине n_i. □

Замечание 2.2. Заметим, что результат теоремы 2.6 выполняется даже если мы не ограничиваем **AM**-протоколы полиномиальным временем.

Глава 3

Иерархии относительно сложности языков

В данной главе доказываются различные теоремы о существовании иерархий для эвристических вычислений. В секции 3.2 приведено новое доказательство иерархии для эвристических вероятностных вычислений с ограниченной ошибкой. В секции 3.3 доказывается иерархия для вероятностных вычислений с ограниченной ошибкой при условии существования односторонних функцй и доказывается иерархия для вероятностных вычислений с ограниченной ошибкой при условии $\mathbf{NP} \subseteq \mathbf{BPP}$, тем самым доказывается иерархия по времени для вероятностных вычислений с ограниченной ошибкой в двух мирах Импальяцо [28] (в Алгоритмике и в Криптомании). В секции 3.4 доказывается иерархия функций для эвристических вероятностных вычислений с ограниченной ошибкой. В секции 3.5 приведено доказательство иерархии для эвристических вероятностных вычислений с ограниченной ошибкой против произвольного распределения, сэмплируемого за время n^a. В секции 3.6 доказывается иерархия для эвристических недетерминированных вычислений.

Результаты этой глав опубликованы в [23].

3.1 Основные определения и обозначения

В дополнение к классу Heur**BPP**, состоящему из пар языков и распределений, определим класс $\mathrm{Heur}_{\delta(n)}\mathbf{FBPTime}[f(n)]$, состоящий из таких пар (F, D), что

1) $F : \{0,1\}^* \to \{0,1\}^*$ — это функция,

2) D — это ансамбль распределений и

3) существует такой вероятностный алгоритм A, работающий $O(f(n))$ шагов, что для любого n выполняется неравенство $\Pr\limits_{x \leftarrow D_n}[\Pr[A(x) = F(x)] \geq \frac{3}{4}] \geq 1 - \delta(n)$, где внутренняя вероятность берется по случайным битам алгоритма A.

Обозначим $\mathrm{Heur}_{\delta(n)}\mathbf{FBPP} = \bigcup\limits_{k \geq 0} \mathrm{Heur}_{\delta(n)}\mathbf{FBPTime}[n^k]$.

Также в дальнейшем мы будем опускать явное указание на равномерное распределение. Например, если будет написано, что $L \in \mathrm{Heur}_{\delta(n)}\mathbf{BPP}$, формально это будет означать $(L, U) \in \mathrm{Heur}_{\delta(n)}\mathbf{BPP}$.

3.2 Иерархия для Heur**BPP**

Следующая теорема является частным случаем теоремы из статьи [18]. Однако ее доказательство намного проще оригинального доказательства Ватсона.

Теорема 3.1 ([18])**.** Для любого $a > 0$ и $\epsilon = \frac{1}{\mathrm{poly}(n)} > 0$ существуют $b > 0$ и ансамбль случайных величин $\gamma_n \in \mathbf{DSamp}[n^b]$ таких, что они принимают значение на множестве $\{0,1\}$ и для любого ансамбля $\alpha_n \in \mathbf{DSamp}[n^a]$ существует n_0, такое что статистическое расстояние между α_{n_0} и γ_{n_0} как минимум $\frac{1}{2} - \epsilon$.

Пусть γ_n — ансамбль случайных величин с носителем $\{0,1\}$. Обозначим $L_\gamma = \bigcup\limits_n \{r \in \{0,1\}^n \mid \Pr[\gamma_n = 1] > 0.r\}$.

Лемма 3.1. Для любой полиномиально сэмплируемой случайной величины γ_n, принимающей значения из $\{0,1\}$, язык L_γ лежит в $\mathrm{Heur}_\epsilon\mathbf{BPP}$ для любого $\epsilon = \frac{1}{\mathrm{poly}(n)} > 0$.

Доказательство. Рассмотрим алгоритм A, который сэмплирует независимо N раз случайную величину γ_n, вычисляет долю значений 1, которую мы обозначим q, и возвращает 1, если $q \geq 0.r$, и 0 в противном случае. По оценке Чернова, если $|0.r - \Pr[\gamma_n = 1]| > \epsilon/4$, то $\Pr[A(r) \neq L_\gamma(r)] < 2e^{-\frac{1}{8}\epsilon^2 N}$; это меньше чем $\frac{1}{4}$ для $N = O(\frac{1}{\epsilon^2})$. Заметим, что $\Pr_r[|0.r - \Pr[\gamma_n = 1]| \leq \epsilon/4] \leq 2^{-n} + \epsilon/2$, а это меньше ϵ для достаточно больших n. □

Лемма 3.2. Пусть L — это язык, такой, что для любого n выполнено неравенство $|\Pr_{x \leftarrow U_n}[x \in L] - \Pr[\gamma_n = 1]| < \delta$ и $L \in \mathrm{Heur}_{\epsilon(n)}\mathbf{BPTime}[n^k]$ для некоторого $\epsilon(n), \delta \geq 0$. Тогда существует такой ансамбль случайных величин β_n, что $\beta_n \in \mathbf{DSamp}[n^{k+1}]$ и $\Delta(\beta_n, \gamma_n) \leq \epsilon(n) + \delta + \frac{1}{2^n}$.

Доказательство. Пусть E — это вероятностный алгоритм, разрешающий L в $\mathrm{Heur}_\epsilon\mathbf{BPTime}[n^k]$, и $\hat{E}(x)$ выполняет $N = O(n)$ раз $E(x)$, а затем возвращает самый частый ответ. Рассмотрим ансамбль случайных величин α_n, который сэмплируется следующим образом: выбирается случайная строка $x \in \{0,1\}^n$ и возвращается $L(x)$. Также рассмотрим случайную величину β_n, которая сэмплируется так: выбирается случайная строка $x \in \{0,1\}^n$ и возвращается $\hat{E}(x)$. Так как $|\Pr_{x \leftarrow U_n}[x \in L] - \Pr[\gamma_n = 1]| < \delta$, мы получаем, что $\Delta(\alpha_n, \gamma_n) < \delta$. Пусть C — это множество таких x, что $\Pr[E(x) = L(x)] \geq \frac{3}{4}$. Из оценки Чернова следует, что для $x \in C$ верно неравенство $\Pr[\hat{E}(x) = L(x)] > 1 - \frac{1}{2^n}$.

Таким образом, мы можем оценить расстояние между β_n и γ_n:

$$\Delta(\beta_n, \gamma_n) = |\Pr_{x,r}[\hat{E}(x) = 1] - \Pr[\gamma_n = 1]|$$
$$\leq |\Pr_{x,r}[\hat{E}(x) = 1] - \Pr_x[x \in L]| + |\Pr_x[x \in L] - \Pr[\gamma_n = 1]| \leq \epsilon(n) + 2^{-n} + \delta.$$

□

Теорема 3.2 ([5])**.** Для любых $b > 0$ и $\delta = \frac{1}{\text{poly}(n)} > 0$ существует такой язык L, что $L \notin \text{Heur}_{\frac{1}{2}-\delta}\mathbf{BPTime}[n^b]$, и для всех $\tau = \frac{1}{\text{poly}(n)}$ $L \in \text{Heur}_{\tau}\mathbf{BPP}$.

Доказательство. Пусть γ_n — ансамбль случайных величин из теоремы 3.1 для $\epsilon = \delta/2$ и $a = b+1$. По лемме 3.1 $L_\gamma \in \text{Heur}_{\tau}\mathbf{BPP}$. Предположим, что $L_\gamma \in \text{Heur}_{\frac{1}{2}-\delta}\mathbf{BPTime}[n^b]$. Заметим, что по построению L_γ выполнено неравенство $|\Pr_{x \leftarrow U_n}[x \in L_\gamma] - \Pr[\gamma_n = 1]| < \frac{1}{2^n}$. Значит, по лемме 3.2 существует $\beta_n \in \mathbf{DSamp}[n^a]$ и $\Delta(\beta_n, \gamma_n) \leq \frac{1}{2} - \delta + \frac{1}{2^n} + \frac{1}{2^n} < \frac{1}{2} - \frac{\delta}{2}$ для достаточно больших n, что противоречит теореме 3.1. □

3.3 Условные иерархии для BPP

Теорема 3.3. Предположим, что существует односторонняя функция. Тогда для любых $\epsilon > 0$ и $a > 0$ существует такой язык $L \in \mathbf{P}$, что $L \notin \text{Heur}_{\frac{1}{2}-\epsilon}\mathbf{BPTime}[n^a]$.

Доказательство. Рассмотрим случайную величину γ_n из теоремы 3.1, и пусть S — это генератор для γ_n. Будем считать, что S принимает случайные биты как второй параметр. Пусть S использует $p(n)$ случайных битов, а G — это псевдослучайный генератор, который переводит n случайных битов в $p(n)$ псевдослучайных (его существование следует из существования односторонней функции [38]). Рассмотрим случайную величину $S(1^n, G(r))$, где $r \leftarrow U_n$. Так как G — псевдослучайный генератор, мы знаем, что $\Delta(S(1^n, G(U_n)), \gamma_n) = \Delta(S(1^n, G(U_n)), S(1^n, U_{p(n)})) < \epsilon/4$ для всех достаточно больших n.

Рассмотрим язык $L = \bigcup_n \{r \in \{0,1\}^n \mid S(1^n, G(r)) = 1\}$. Очевидно, что $L \in \mathbf{P}$. Из леммы 3.2 и теоремы 3.1 следует, что $L \notin \text{Heur}_{\frac{1}{2}-\epsilon}\mathbf{BPTime}[n^a]$. □

Мы также докажем, что в **BPTime** верна теорема об иерархии, если все языки из **NP** «простые».

Теорема 3.4. Если $\mathbf{NP} \subseteq \mathbf{BPP}$, то $\mathbf{BPTime}[n^k] \subsetneq \mathbf{BPP}$ для всех $k > 0$.

Доказательство. Предположим, что $\mathbf{BPP} \subseteq \mathbf{BPTime}[n^k]$. По утверждению, аналогичному теореме Адлемана, $\mathbf{BPTime}[n^k] \subseteq \mathbf{Size}[n^{2k+2}]$. Из результатов [39] следует, что если $\mathbf{NP} \subseteq \mathbf{BPP}$, то $\mathbf{PH} \subseteq \mathbf{BPP}$. Таким образом, если $\mathbf{BPP} = \mathbf{BPTime}[n^k]$, то $\mathbf{PH} \subseteq \mathbf{Size}[n^{2k+2}]$, что противоречит теореме Каннана [6]. □

3.4 Теорема об иерархии для функций

В этой секции мы усилим теорему об иерархии по эвристическому времени и докажем ее для функций, принимающих константное число значений. Теперь нам понадобится полная версия теоремы Ватсона.

Теорема 3.5 ([18])**.** Для любых $a > 0, k > 0$ и $\epsilon = \frac{1}{\mathrm{poly}(n)} > 0$ существуют $b > 0$ и ансамбль случайных величин $\gamma_n \in \mathbf{DSamp}[n^b]$, принимающих значение из множества $\{0, 1, \ldots, k-1\}$, таких, что для любого ансамбля $\alpha_n \in \mathbf{DSamp}[n^a]$ существует такое n_0, что статистическое расстояние между α_{n_0} и γ_{n_0} как минимум $1 - \frac{1}{k} - \epsilon$.

Пусть γ_n — это ансамбль случайных величин, принимающих значения из $\{0, 1, \ldots, k-1\}$. Тогда для каждого n мы разобьем отрезок $[0, 1)$ на k непересекающихся частей $I_0^{(n)} = \left[p_0^{(n)} = 0, p_1^{(n)}\right), I_1^{(n)} = \left[p_1^{(n)}, p_2^{(n)}\right), \ldots, I_{k-1}^{(n)} = \left[p_{k-1}^{(n)}, p_k^{(n)} = 1\right)$. Для всех $i \in \{0, 1, \ldots, k-1\}$ определим $p_{i+1}^{(n)} = p_i^{(n)} + \Pr[\gamma_n = i]$. Также определим функцию $F_\gamma : \{0,1\}^* \to \{0, 1, \ldots, k-1\}$ следующим образом: для любого $r \in \{0,1\}^n$ верно $F_\gamma(r) = i$ тогда и только тогда, когда $0.r \in I_i^{(n)}$. Более формально, $F_\gamma(r) = \min\{i \in \{0, 1, \ldots, k-1\} \mid \Pr[\gamma_n \in \{0, 1, \ldots, i\}] > 0.r\}$.

Следующая лемма является усилением леммы 3.1.

Лемма 3.3. Для любого полиномиально сэмплируемого ансамбля случайных величин γ_n, принимающих значения из $\{0, 1, \ldots, k-1\}$, функция $F_\gamma \in \mathrm{Heur}_\epsilon\mathbf{FBPP}$ для всякого $\epsilon = \frac{1}{\mathrm{poly}(n)} > 0$.

Доказательство. Рассмотрим следующий алгоритм A: вычислить γ_n независимо $N = O\left(\frac{k^4 \log(k)}{\epsilon^2}\right)$ раз. Пусть q_i — это доля результатов i. Найдем минимальное такое j, что $\sum_{i=1}^{j} q_i > 0.r$ и вернем j. По оценке Чернова $\Pr[|q_i - \Pr[\gamma_n = i]| > \epsilon/2k^2] < 2e^{-\frac{1}{2k^4}\epsilon^2 N}$.

Предположим, что для всех $i \in \{0, 1, \ldots, k-1\}$ верно неравенство $|0.r - p_i^{(n)}| > \frac{\epsilon}{2k}$. В таком случае $A(r) \neq F_\gamma(r)$ тогда и только тогда, когда для некоторого $j \leq i$ выполняется $|q_j - \Pr[\gamma_n = j]| > \frac{\epsilon}{2k^2}$, а это происходит с вероятностью не больше $2e^{-\frac{1}{2k^4}k\epsilon^2 N} < \frac{1}{4}$ для $N = O\left(\frac{k^4 \log(k)}{\epsilon^2}\right)$. Значит, $\Pr[A(r) \neq F_\gamma(r)] < \frac{1}{4}$.

Заметим, что $\Pr_r[\forall i \in \{0, 1, \ldots, k-1\}\ |0.r - \Pr[\gamma_n = i]| \leq \frac{\epsilon}{2k}] \leq k2^{-n} + \epsilon/2$, а это меньше чем ϵ для достаточно больших n. □

Лемма 3.4. Пусть $F \in \mathrm{Heur}_{\epsilon(n)}\mathbf{FBPTime}[n^a]$ — это функция из $\{0,1\}^*$ в $\{0, 1, \ldots, k-1\}$, такая, что для всех n статистическое расстояние между $F(U_n)$ и γ_n не более δ для некоторых $\epsilon(n), \delta \geq 0$. Тогда существует ансамбль случайных величин β_n, таких, что $\beta_n \in \mathbf{DSamp}[n^{a+1}]$ и $\Delta(\beta_n, \gamma_n) \leq \epsilon(n) + \delta + \frac{1}{2^n}$.

Доказательство. Доказательство повторяет доказательство леммы 3.2. □

Теорема 3.6. Для любого $b > 0$, $k > 0$ и $\delta = \frac{1}{\mathrm{poly}(n)} > 0$ существует такая функция $F\colon \{0,1\} \to \{0, 1, \ldots, k-1\}$, что $F \notin \mathrm{Heur}_{1-\frac{1}{k}-\delta}\mathbf{FBPTime}[n^b]$ и $F \in \mathrm{Heur}_\tau\mathbf{FBPP}$ для любого $\tau = \frac{1}{\mathrm{poly}(n)} > 0$

Доказательство. Пусть γ_n — это ансамбль из теоремы 3.5 для $\epsilon = \frac{\delta}{2}$ и $a = b + 1$. По лемме 3.3 $F_\gamma \in \mathrm{Heur}_\tau\mathbf{FBPP}$. Предположим, что $F_\gamma \in \mathrm{Heur}_{1-\frac{1}{k}-\delta}\mathbf{FBPTime}[n^b]$. Теперь заметим, что по построению F_γ для всякого $i \in \{0, 1, \ldots, k-1\}$ выполняется неравенство $|\Pr_{x \leftarrow U_n}[F_\gamma(x) = i] - \Pr[\gamma_n = i]| < \frac{1}{2^n}$. Значит, $\Delta(F_\gamma(U_n), \gamma_n) \leq \frac{k}{2^{n+1}}$. Таким образом, по лемме 3.4 существует такой $\beta_n \in \mathbf{DSamp}[n^a]$, что $\Delta(\beta_n, \gamma_n) \leq 1 - \frac{1}{k} - \delta + \frac{k}{2^{n+1}} + \frac{1}{2^n} < 1 - \frac{1}{k} - \frac{\delta}{2}$ для достаточно больших n. Однако это противоречит теореме 3.5. □

3.5 Иерархия с произвольными распределениями

В этой секции результаты из предыдущих секций будут перенесены на произвольные распределения. К сожалению, теперь теоремы Ватсона нам недостаточно, но, к счастью, доказательство Ватсона позволяет доказать чуть более сильное утверждение:

Теорема 3.7. Для любых $a > 0$, $k > 0$ и $\epsilon = \frac{1}{\text{poly}(n)} > 0$ существуют $b > 0$ и такой ансамбль случайных величин $\gamma_n \in \mathbf{DSamp}[n^b]$, что γ_n принимает значения из $\{0, 1, \ldots, k-1\}$ и для любого ансамбля $\alpha_n \in \mathbf{DSamp}[n^a]$ и целого $N > 0$ существует $c \in \{0, 1, \ldots, k-1\}$ и $n_0 > N$, что $\Pr[\gamma_{n_0} = c] \geq 1 - \epsilon/2$ и $\Pr[\alpha_{n_0} = c] \leq \frac{1}{k} + \epsilon/2$.

Доказательство теоремы 3.7 для $k = 2$ в точности повторяет доказательство теоремы 3.1.

Пусть γ_n — это ансамбль случайных величин, принимающих значения из $\{0, 1, \ldots, k-1\}$. Как и в прошлой секции, мы разобьем отрезок $[0, 1)$ на k частей: $I_0^{(n)}, \ldots, I_{k-1}^{(n)}$. Определим функцию $H_\gamma : \{0,1\}^* \to \{0, 1, \ldots, k-1\}$, так что для всех $r \in \{0,1\}^n$, выполнено $H_\gamma(r) = i$ тогда и только тогда, когда $\Theta_r \in I_i^{(n)}$, где $\Theta_r = \frac{3}{8} + \frac{0.r}{4}$.

Доказательство следующей леммы почти в точности повторяет доказательство леммы 3.3.

Лемма 3.5. Для любого полиномиально сэмплируемого ансамбля случайных величин γ_n, принимающих значения из $\{0, 1, \ldots, k-1\}$ верно, что $H_\gamma \in \text{Heur}_\epsilon\mathbf{FBPP}$ для любого $\epsilon = \frac{1}{\text{poly}(n)} > 0$.

Теорема 3.8. Для любых $b > 0$, $k > 0$ и $\delta = \frac{1}{\text{poly}(n)} > 0$ существует функция $H\colon \{0,1\}^* \to \{0, 1, \ldots, k-1\}$, такая, что для любого $D \in \mathbf{DSamp}[n^b]$ $(H, D) \notin \text{Heur}_{1-\frac{1}{k}-\delta}\mathbf{FBPTime}[n^b]$ и для любого $\tau = \frac{1}{\text{poly}(n)}$ выполнено $H \in \text{Heur}_\tau\mathbf{FBPP}$.

Доказательство. Пусть γ_n — это ансамбль из теоремы 3.1 для $\epsilon = \frac{\delta}{2}$ и $a = b + 1$. По лемме 3.5 мы получаем $H_\gamma \in \text{Heur}_\delta\mathbf{FBPP}$.

Предположим, что для некоторого $D \in \mathbf{DSamp}[n^b]$ выполняется, что $(H_\gamma, D) \in \mathrm{Heur}_{1-\frac{1}{k}-\delta}\mathbf{FBPTime}[n^b]$. Пусть E — вероятностный алгоритм, разрешающий (H_γ, D), и G — генератор для D. Пусть алгоритм $\hat{E}(x)$ запускает $N = O(n)$ раз алгоритм $E(x)$ и возвращает самый частый ответ. Теперь рассмотрим ансамбль случайных величин, который сэмплируется следующим алгоритмом: сначала выберем случайный $x \leftarrow G(1^n)$, а затем вернем $\hat{E}(x)$. Заметим, что α_n сэмплируется за $O(n^{b+1})$ шагов.

По теореме 3.7, примененной к α_n и $\epsilon = \frac{\delta}{2}$, существуют n_0 и $c \in \{0, 1, 2, \ldots, k-1\}$, такие, что $\Pr[\gamma_{n_0} = c] \geq 1 - \frac{\delta}{4}$ и $\Pr[\alpha_{n_0} = c] \leq \frac{1}{k} + \frac{\delta}{4}$. Размер $I_c^{(n_0)}$ не меньше $\frac{3}{4}$ для $\delta < 1$. Таким образом, $\Theta_r \in I_c^{(n_0)}$ для любого $r \in \{0,1\}^{n_0}$. Значит, $H_\gamma(x) = c$ для всех $x \in \{0,1\}^{n_0}$, по определению α_n и оценке Чернова $\Pr[\alpha_{n_0} = c] \geq \frac{1}{k} + \delta - 2^{-n_0} > \frac{1}{k} + \delta/2$ для достаточно больших n_0, что противоречит теореме 3.7. □

3.6 Иерархия для HeurNP

В этой секции при помощи того же метода доказывается теорема об иерархии для эвристической версии **NP**.

Определение 3.1. Ансамбль случайных величин γ_n недетерминированно сэмплируем за время $O(f(n))$ с использованием n^k случайных битов тогда и только тогда, когда существует такой недетерминированный алгоритм A, что на входе $(1^n, r)$ он работает $O(f(n))$ шагов и $A(1^n, r)$ распределено так же, как и γ_n, если r распределено равномерно на $\{0,1\}^{n^k}$. Мы обозначим множество ансамблей, сэмплируемых за время $O(f(n))$ с использованием n^k случайных битов, как $\mathbf{NSamp}_{n^k}[f(n)]$.

Для доказательства иерархии эвристической версии **NTime** нам понадобятся булевы сэмплеры, введенные в работе [40] вместо миксеров, которые использовал Первышев. На самом деле не очень важно, использовать ли сэмплеры или миксеры, но сэмплеры позволяют сделать изложение более элегантным.

Определение 3.2. Введем обозначение $\bar{f} = \frac{1}{2^n} \sum\limits_{x \in \{0,1\}^n} f(x)$, где $f\colon \{0,1\}^n \to \{0,1\}$. Булев сэмплер — это вероятностный оракульный алгоритм S, принимающий на вход целое число n и рациональные числа δ, ϵ, имеющий оракульный доступ к функции $f\colon \{0,1\}^n \to \{0,1\}$, делает несколько неадаптивных запросов к функции f и возвращающий такое число q из отрезка $[0,1]$, что $\Pr[|q - \bar{f}| \geq \epsilon] < \delta$.

Булев сэмплер называется усредняющим, если он возвращает среднее значение запрошенных значений.

Теорема 3.9 ([40])**.** Существует усредняющий булев сэмплер S, использующий n случайных бит, делающий $q(n, \epsilon, \delta) = O(\frac{1}{\epsilon^2 \delta})$ запросов и работающий полиномиальное от n, $\frac{1}{\epsilon}$ и $\frac{1}{\delta}$ время.

Следствие 3.1. Существует усредняющий булев сэмплер S, использующий $n-1$ случайных битов, делающий $q(n, \epsilon, \delta) = O(\frac{1}{\epsilon^2 \delta})$ запросов и работающий полиномиальное от n, $\frac{1}{\epsilon}$ и $\frac{1}{\delta}$ время.

Доказательство. Пусть S — это сэмплер из теоремы 3.9. Определим алгоритм S' следующим образом: на входе (n, ϵ, δ) наш алгоритм возвращает $\frac{1}{2}S^{f_0}(n-1, \epsilon, \frac{\delta}{2}) + \frac{1}{2}S^{f_1}(n-1, \epsilon, \frac{\delta}{2})$, где $f_0, f_1 : \{0,1\}^{n-1} \to \{0,1\}$ и $f_0(x) = f(x0)$, $f_1(x) = f(x1)$, S^{f_0} и S^{f_1} используют одни и те же $n-1$ случайных битов. Заметим, что $\frac{\bar{f}_0 + \bar{f}_1}{2} = \bar{f}$. Значит,

$$\Pr[|S'^f(n, \epsilon, \frac{\delta}{2}) - \bar{f}| \geq \epsilon] \leq$$
$$\Pr[|S^{f_0}(n, \epsilon, \frac{\delta}{2}) - \bar{f}_0| \geq \epsilon] + \Pr[|S^{f_1}(n, \epsilon, \delta) - \bar{f}_1| \geq \epsilon] < \frac{\delta}{2} + \frac{\delta}{2} = \delta.$$

□

Следующая теорема аналогична теореме 3.1 для распределений, сэмплируемых недетерминированными алгоритмами с фиксированным числом случайных битов. В доказательстве теоремы 3.1 мы считали вероятность того, что $\gamma_n = 1$, простым сэмплированием, теперь мы для экономии случайных битов воспользуемся сэмплерами.

Теорема 3.10. Для любого $a > 0$ и $\epsilon > 0$ существуют $b > 0$ и такой ансамбль случайных величин $\gamma_n \in \mathbf{NSamp}_n[n^b]$, принимающий значения из $\{0,1\}$, что для любого ансамбля $\alpha_n \in \mathbf{NSamp}_n[n^a]$ со значениями из $\{0,1\}$ существует такое n, что статистическое расстояние между α_n и γ_n как минимум $\frac{1}{2} - \epsilon$.

Доказательство. Мы будем использовать отложенную диагонализацию. Пусть E_i — перечисление всех недетерминированных алгоритмов (мы будем интерпретировать их как генераторы случайных величин использующие n случайных битов); мы предположим, что все E_i оснащены будильником, который останавливает исполнение на входе $(1^n, r)$ после n^{a+1} шагов. Пусть S — это булев сэмплер из следствия 3.1. Определим последовательность n_i следующим образом $n_1 = 1$, $n_{i+1} = n_i^* + 1$ и $n_i^* = 2^{n_i^{a+1}}$. Теперь рассмотрим случайную величину γ_n, которая сэмплируется следующим алгоритмом $\Gamma(1^n, r)$ ($r \leftarrow U_n$ — строка случайных битов) для таких n, что $n_i \leq n \leq n_i^*$:

- если $n = n_i^*$, то $\Gamma(1^n, r)$ всегда возвращает элемент $\{0,1\}$ с самой маленькой вероятностью относительно $E_i(1^{n_i}, r)$, где r равномерно распределено на $\{0,1\}^n$, это может быть сделано простым перебором;

- если $n_i \leq n < n_i^* - 1$, то запустим $S^f(1^{n+1}, \frac{\epsilon}{2}, \frac{1}{4})$, используя r в качестве случайных битов, где $f : \{0,1\}^{n+1} \to \{0,1\}$ и $f(z) = E_i(1^{n+1}, z)$; затем вернем 1, если результат больше $\frac{1}{2}$, и 0 в противном случае; Здесь мы используем, что S — усредняющий сэмплер, и недетерминированные вычисления замкнуты относительно монотонных операций.

Пусть α_n сэмплируется недетерминированным алгоритмом $A(1^n, r)$, использующим n случайных битов и работающим $O(n^a)$ шагов. В соответствии с построением последовательности E_i существует такое i, что E_i эквивалентен A. Докажем от противного, что существует такое n ($n_i \leq n \leq n_i^*$), что $\Delta(\gamma_n, \alpha_n) > \frac{1}{2} - \epsilon$. Предположим, что это не так.

Пусть b — это значение, которое случайная величина $\gamma_{n_i^*}$ принимает с вероятностью 1 (по определению $\Pr[E_i(1^{n_i}) = b] \leq \frac{1}{2}$). Мы докажем индукцией по k (для $0 \leq k \leq n_i^* - n_i$), что $\Pr[\gamma_{n_i^*-k} = b] > 1 - \frac{\epsilon}{2}$. База при $k = 0$ очевидна. Докажем шаг индукции. По предположению индукции $\Pr[\alpha_{n_i^*-k} = b] \geq \Pr[\gamma_{n_i^*-k} = b] - \frac{1}{2} + \epsilon > 1 - \frac{\epsilon}{2} - \frac{1}{2} + \epsilon = \frac{1}{2} + \frac{\epsilon}{2}$. Значит, по определению булева сэмплера $\Pr[\gamma_{n_i^*-k-1} = b] \geq 1 - \frac{\epsilon}{2}$.

Завершая доказательство, мы получаем противоречие, так как $\Pr[\alpha_{n_i} = b] \leq \frac{1}{2}$. □

Теорема 3.11 ([5])**.** Для любых $b > 0$ и $\delta > 0$ существует такой язык L, что $L \notin \mathrm{Heur}_{\frac{1}{2}-\delta}\mathbf{NTime}[n^b]$ и $L \in \mathbf{NP}$.

Доказательство. Пусть γ_n — ансамбль распределений из теоремы 3.10 для $\epsilon = \delta/2$ и $a = b + 1$, а S — это генератор для этого распределения. Рассмотрим язык $L = \{x | S(1^{|x|}, x) = 1\}$. Несложно увидеть, что этот язык из $\mathbf{NP}$. Покажем, что $L \notin \mathrm{Heur}_{\frac{1}{2}-\delta}\mathbf{NTime}[n^b]$. Предположим обратное, пусть недетерминированный алгоритм A разрешает L за время n^b с ошибкой меньше $\frac{1}{2} - \delta$. В таком случае случайная величина α_n, распределенная в соответствии с $A(x)$ при $x \leftarrow U_n$, близка к γ_n, а точнее, $\Delta(\alpha_n, \gamma_n) < \frac{1}{2} - \delta$ для всех n. Но это противоречит теореме 3.10. □

Глава 4

Иерархии относительно сложности распределений

В данной главе доказывается связь иерархий распределений с иерархиями распределенных задач относительно сложности распределений. Также доказывается наличие иерархии с растущим расстоянием по времени для сэмплирования распределений и иерархии с растущим расстоянием по времени вычисления распределений.

В секции 4.1 приводятся необходимые обозначения. В секции 4.2 доказываются наличие сильной и слабой иерархии по времени сэмплирования распределений и их связь с иерархией распределенных задач относительно сложности сэмплирования распределений. В секции 4.3 изложено доказательство иерархии по времени вычисления распределений, а также доказана иерархия распределенных задач относительно сложности вычисления распределений.

Результаты этой главы опубликованы в [24].

4.1 Основные определения и обозначения

В этой главе нам понадобится эвристический аналог рекурсивных языков. Распределенная задача (L, D) принадлежит классу $\mathrm{Heur}_{\delta(n)}\mathbf{R}$ тогда и только тогда, когда существует такой алгоритм A, что $\Pr\limits_{x \leftarrow D_n}[A(x) \neq L(x)] \leq \delta(n)$ для всех n.

4.2 Сэмплируемые распределения

4.2.1 Строгая сложность

Будем говорить, что функция $f(n)$ конструируема, если существует алгоритм, который по входу n вычисляет $f(n)$ за время $O(f(n))$.

Определение 4.1. Будем говорить, что две конструируемые функции f и g удовлетворяют *свойству иерархии сэмплируемых распределений* с параметром $\lambda(n)$, если существует ансамбль распределений $F \in \mathbf{DSamp}[f(n)]$, такой, что для любого ансамбля распределений $G \in \mathbf{DSamp}[g(n)]$ существует бесконечно много таких n, что статистическое расстояние между F_n и G_n, как минимум, $1 - \lambda(n)$.

Определение 4.2. Будем говорить, что две конструируемые функции f и g удовлетворяют *свойству иерархии распределенных задач* с параметрами $\alpha(n) > 0$ и $\beta(n) > 0$, если существуют такой язык L и такой ансамбль распределений $F \in \mathbf{DSamp}[f(n)]$, что

- $(L, G) \in \mathrm{Heur}_{\alpha(n)}\mathbf{P}$ для всех $G \in \mathbf{DSamp}[g(n)]$;
- $(L, F) \notin \mathrm{Heur}_{1-\beta(n)}\mathbf{P}$.

Также мы будем говорить, что f и g удовлетворяют *строгому* свойству иерархии распределенных задач, если

- существует такой алгоритм A, работающий линейное время, что для всех $G \in \mathbf{DSamp}[g(n)]$ и всех достаточно больших n выполняется неравенство $\Pr\limits_{x \leftarrow G_n}[A(x) = L(x)] \geq 1 - \alpha(n)$,
- $(L, F) \notin \mathrm{Heur}_{1-\beta(n)}\mathbf{R}$.

Лемма 4.1. Для всех конструируемых функций $f(n)$, $h(n)$ и $g(n) \geq n$, таких, что $g(n) \log g(n) = o(h(n))$, если f и h удовлетворяют свойству иерархии сэмплируемых распределений с параметром $\lambda(n)$, то f и g удовлетворяют свойству иерархии распределенных задач с параметрами $\alpha(n)$ и $\lambda(n)$ для некоторого $\alpha(n) = \omega(\lambda(n))$.

Доказательство. Пусть A_i — это перечисление всех вероятностных алгоритмов, оснащенных будильником, который прекращает выполнение алгоритма после $O(g(n))$ шагов. Мы будем думать об A_i, как об алгоритмах, которые сэмплируют распределения; выход $A_i(1^n)$ будем интерпретировать как строку каким-то фиксированным образом. Пусть B — это алгоритм, сэмплирующий распределение следующим образом: на входе 1^n с вероятностью $\frac{1}{2}$ он запускает $A_1(1^n)$ (и возвращает его результат), с вероятностью $\frac{1}{2^2}$ запускает $A_2(1^n)$, ..., с вероятностью $\frac{1}{2^{n-1}}$ запускает $A_{n-1}(1^n)$ и с вероятностью $\frac{1}{2^{n-1}}$ запускает $A_n(1^n)$. Пусть ансамбль распределений E сэмплируется алгоритмом B. Очевидно, что $E \in \mathbf{DSamp}[h(n)]$.

Благодаря тому, что f и h удовлетворяют свойству иерархии сэмплируемых распределений, существует такой ансамбль $D \in \mathbf{DSamp}[f(n)]$, что $\Delta(D_n, E_n) \geq 1 - \lambda(n)$ для бесконечно многих n. Обозначим множество таких n как $I = \{n_1, n_2, \dots\}$. Для $n \in I$ существует такое множество $S_n \subseteq \{0,1\}^n$, что $D_n(S_n) - E_n(S_n) \geq 1 - \lambda(n)$, а значит, $E_n(S_n) \leq \lambda(n)$.

Определим язык L таким образом, что $L \subseteq \bigcup\limits_{n \in I} S_n$. Пусть T_i — это перечисление всех алгоритмов. Определим язык L так, что для любого $x \in S_{n_k}$, $x \in L$ тогда и только тогда, когда T_k не останавливается или отвергает на входе x. По определению $(L, D) \notin \mathrm{Heur}_{1-\lambda(n)}\mathbf{R}$.

Рассмотрим алгоритм, который возвращает 0 на всех входах. Если $R \in \mathbf{DSamp}[g(n)]$, то существует i, такое, что A_i сэмплирует R. Для $n \geq i$ для любого множества $S \subseteq \{0,1\}^n$ выполняется неравенство $E(S) \geq 2^{-i}R(S)$. Значит, для любого ансамбля $R \in \mathbf{DSamp}[g(n)]$ этот алгоритм имеет ошибку не более $c\lambda(n)$, где c — это константа, зависящая только от R; $c\lambda(n) < \alpha(n)$ для достаточно больших n. □

Также можно доказать обратную импликацию.

Лемма 4.2. Если f и g удовлетворяют свойству иерархии распределенных задач с параметрами $\alpha(n)$ и $\beta(n)$, то f и g удовлетворяют свойству иерархии сэмплируемых распределений с параметром $\alpha + \beta$.

Доказательство. Для любого $F \in \mathbf{DSamp}[g(n)]$ существует полиномиальный алгоритм A, разрешающий (L, F) в $\mathrm{Heur}_{\alpha(n)}\mathbf{P}$, и в то же время $(L, D) \notin \mathrm{Heur}_{1-\beta(n)}\mathbf{P}$. Пусть S_n — это множество таких $x \in \{0,1\}^n$, что $A(x) = L(x)$. Мы знаем, что $F_n(S_n) \geq 1 - \alpha(n)$ для всех n и $D_n(S_n) \leq \beta(n)$ тоже для всех n. Значит, $\Delta(D_n, F_n) \geq F_n(S_n) - D_n(S_n) \geq 1 - \alpha(n) - \beta(n)$ для бесконечно многих n. □

Лемма 4.1 и лемма 4.2 влекут, что если f и g удовлетворяют свойству иерархии распределенных задач с параметрами $\alpha(n)$ и $\beta(n)$ стремящимися к нулю, то f и $g/\log^2 g$ удовлетворяют строгому свойству иерархии распределенных задач с бесконечно малыми параметрами.

Несложно видеть, что теорема Ватсона, сформулированная ранее, доказывает, что для любых $a > 0$, $\epsilon > 0$ и любой константы k существует такое $b > 0$, что n^a и n^b удовлетворяют свойству иерархии сэмплируемых распределений с параметром $\frac{1}{k} + \epsilon$.

Ватсон предположил, что для любого $a > 0$ существуют такая монотонно стремящаяся к нулю функция $\alpha(n)$ и такая константа $b > 0$, что n^a и n^b удовлетворяют свойству иерархии сэмплируемых распределений с параметром $\alpha(n)$. Это утверждение все еще является открытым вопросом.

Однако мы докажем следующее, более слабое, утверждение:

Теорема 4.1. Для любых a, b, c, таких, что $0 < a < b$ и $c > 0$, функции $f(n) = n^{\log^b n}$ и $g(n) = n^{\log^a n}$ удовлетворяют свойству иерархии сэмплируемых распределений с параметром $\lambda(n) = \frac{1}{2^{(\log\log\log n)^c}}$.

Следствие 4.1. Для любых a, b, c, таких, что $0 < a < b$ и $c > 0$, функции $f(n) = n^{\log^b n}$ и $g(n) = n^{\log^a n}$ удовлетворяют строгому свойству иерархии распределенных задач с параметрами $\alpha(n) = \beta(n) = \frac{1}{2^{(\log\log\log n)^c}}$.

Доказательство. Следует из леммы 4.1 и теоремы 4.1. □

Следствие 4.2. Для любых $\epsilon > 0$ и $c > 0$ существуют язык L и линейный по времени алгоритм A, такие, что для любого полиномиально сэмплируемого ансамбля распределений F и любого натурального n выполняется $\Pr\limits_{x \leftarrow F_n}[A(x) = L(x)] \geq 1 - \frac{1}{2^{(\log\log\log n)^c}}$ и существует такой ансамбль $D \in \mathbf{DSamp}[n^{\log^\epsilon n}]$, что для любого алгоритма B и для бесконечно многих n выполняется $\Pr\limits_{x \leftarrow D_n}[B(x) = L(x)] \leq \frac{1}{2^{(\log\log\log n)^c}}$.

Прежде чем дать формальное доказательство теоремы 4.1, опишем основные идеи доказательства.

В дальнейшем, для удобства, будем предполагать, что распределения распределены на множестве $\{0, 1, \ldots, 2^n - 1\}$ вместо $\{0,1\}^n$.

Следующее доказательство, подобно доказательству Ватсона, основано на древесной диагонализации. Построим распределение D и диагонализуемся против всех распределений, сэмплируемых за $O(g(n))$ шагов (перечислив их генераторы A_i). Для i-го распределения докажем, что статистическое расстояние между D и $A_i(1^n)$ большое для некоторого n между $[n_i, n_i^*]$, где n_i^* сильно больше чем n_i. Для каждого i построим дерево T_i, вершины которого помечены числами из $[n_i, n_i^*]$ без повторений. Корень дерева помечен n_i^* и листья T_i помечены числами, примерно равными n_i. Число в родителе больше, чем числа в детях, но при этом ограничено квазиполиномом от чисел в детях. Пусть t — это элемент $\{0, 1, \ldots, 2^{n_i} - 1\}$, такой, что во всех листьях его вероятность относительно A_i меньше чем $\lambda(m_i)$, где m_i — максимальный номер в листьях. Такое t существует из-за того, что у нас не очень много листьев, размер носителя распределения как минимум 2^{n_i} и для любого распределения доля элементов с вероятностью большей чем $\lambda(n)$ не больше чем $\frac{1}{\lambda(n)}$. Распределение $D_{n_i^*}$ определим сосредоточенным на t. Предположим для всех $n \in [n_i; n_i^*]$, что статистическое расстояние между $A_i(1^n)$ и D_n меньше $1 - \lambda(n)$. Наша цель — доопределить так D, что хотя бы один лист будет почти сосредоточен на t. Так мы получим противоречие.

Мы будем передавать информацию о t от родителя к как минимум одному ребенку. Распределение D на детях p имеет следующее свойство:

если D_p почти сосредоточено на каком-то элементе (какой-то элемент имеет вероятность $1-\epsilon$), то D_n почти сосредоточено на нем же, где n — сын p. Так мы пройдем от корня к листу и получим то, что искали.

Для того чтобы гарантировать это свойство, заметим, что из предположения о статистическом расстоянии мы получаем, что $\Pr[A_i(1^p) = t] \geq \lambda(p) - \epsilon$, значит, существует не более $\frac{2}{\lambda}$ кандидатов на роль t. Мы будем генерировать список, который будет с высокой вероятностью содержать все элементы с вероятностью $\lambda(p) - \epsilon$ относительно $A_i(1^p)$. После этого мы определим D в первом сыне p, сконцентрированном на первом элементе списка, во втором сыне на втором элементе и так далее; так как искать этот список мы будем вероятностно, то в разных сыновьях может получаться разный список; мы решим эту проблему, используюя несколько пороговых функций (более формально это будет объяснено в следующей лемме).

Лемма 4.3. Существует такой алгоритм $C^\bullet(n, i, \delta, \lambda)$, имеющий оракульный доступ к некоторой случайной величине γ, принимающей значения из $\{0, 1, \ldots, 2^n - 1\}$, что для любого положительного n и констант $\delta, \lambda \in (0, 1]$, если $\Pr[\gamma = t] \geq \lambda$ для некоторого t, то существует такое число $0 \leq i \leq \lceil 1 + \frac{1}{\lambda} \rceil^2$, что $\Pr[C^\gamma(n, i, \delta, \lambda) = t] \geq 1 - \delta$ и $C^\bullet$ работает не больше $\mathrm{poly}(n, \log \frac{1}{\delta}, \frac{1}{\lambda})$ шагов.

Доказательство. Рассмотрим следующий алгоритм $C^\gamma(n, i, \delta, \lambda)$:

1) пусть $k = \lceil \frac{1}{\lambda} + 1 \rceil$ и $\epsilon = \frac{\lambda^3}{10k}$;

2) будем интерпретировать i как пару (a, b), где $a, b \in [k]$;

3) запросим у оракула $N = \lceil \frac{2(n+1+\log \frac{1}{\delta})}{\epsilon^2} \rceil$ сэмплов γ;

4) теперь рассмотрим список $y_1, \ldots, y_s$ всех элементов, которые были выданы больше чем $(\lambda - \epsilon a)N$ раз;

5) если $b \leq s$, вернем y_b, а в противном случае вернем 0.

Заметим, что для $\lambda \in (0, 1]$

$$k(\lambda - \epsilon(2k)) \geq (\frac{1}{\lambda} + 1)(\lambda - \lambda^3/5) = 1 + \lambda - \lambda^2/5 - \lambda^3/5 > 1. \quad (4.1)$$

Значит, число таких элементов x, что $\Pr[\gamma = x] > \lambda - \epsilon k$ меньше чем k; по аналогичному рассуждению $s < k$, где s — это размер списка на шаге 4 алгоритма C.

Рассмотрим интервалы $I_j = [\lambda - \epsilon j - \epsilon/2; \lambda - \epsilon j + \epsilon/2]$. Существует такое $a \in [k]$, что $\Pr[\gamma = x] \notin I_a$ для всех x, потому что иначе $1 = \sum\limits_x \Pr[\gamma = x] \geq k(\lambda - \epsilon k - \epsilon/2)$, что противоречит неравенству 4.1. Значит, существует $a \in [k]$, такое, что $|\Pr[\gamma = x] - \lambda - \epsilon a| > \epsilon/2$ для всех x.

Пусть $x_1, \ldots, x_l$ — это список всех таких x, что $\Pr[\gamma = x] > \lambda - \epsilon a$. Мы знаем, что если $\Pr[\gamma = x] > \lambda - \epsilon a$, то $\Pr[\gamma = x] > \lambda - \epsilon a + \epsilon/2$, и если $\Pr[\gamma = x] \leq \lambda - \epsilon a$, то $\Pr[\gamma = x] < \lambda - \epsilon a - \epsilon/2$. Для фиксированного a для любого $j \in [l]$ строка x_j встречается в списке, возникающем на шаге 4 алгоритма C, с вероятностью как минимум $1 - 2e^{-\epsilon^2 N/2}$. Если $\Pr[\gamma = x] \leq \lambda - \epsilon a$, то по оценке Чернова x не встречается в списке с вероятностью как минимум $1 - 2e^{-\epsilon^2 N/2}$. Так как носитель γ не больше 2^n с вероятностью $1 - 2^{n+1}e^{-\epsilon^2 N/2} \geq 1 - \delta$, список, генерируемый на шаге 4 — в точности список $x_1, \ldots, x_l$. Так как $\Pr[D = t] > \lambda$, существует такой номер b, что $x_b = t$. Значит, если $i = (a, b)$, то $\Pr[C^\gamma(n, i, \delta, \lambda) = t] \geq 1 - \delta$. □

Доказательство теоремы 4.1. Напомним, что наше доказательство будет использовать древесную диагонализацию. Мы диагонализуемся против всех вероятностных алгоритмов, оснащенных $O(g(n))$-будильником, будем интерпретировать их как генераторы для распределений. Пусть $A_1, A_2, \ldots$ — это перечисление всех таких алгоритмов.

Рассмотрим $\epsilon > 0$, такое, что $(1 + a)(1 + \epsilon) < (1 + b)$, и зафиксируем c. Определим такие целочисленные последовательности n_i и n_i^*, что $n_1 = 1$, $n_i^* = 2^{(\log n_i)^{(1+\epsilon)^{d_i}}}$, где $d_i = \lceil \log_{1+\epsilon} 2 \rceil \lceil (\log \log n_i)^2 \rceil$ и $n_{i+1} = n_i^* + 1$. Для любого i определим такой ансамбль распределений D_n ($n \in \{n_i, n_i + 1, \ldots, n_i^*\}$), что существует $k \in \{n_i, n_i + 1, \ldots, n_i^*\}$

удовлетворяющее условию $\Delta(D_k, A_i(1^k)) \geq 1 - \lambda(k)$.

Лемма 4.4. Для каждого $\epsilon > 0$ существует такое семейство деревьев T_i, что

1) множество всех вершин T_i — это подмножество $\{n_i, n_i + 1, \ldots, n_i^*\}$;

2) n_i^* — это корень T_i;

3) все листья T_i имеют номера не больше $m_i = 2n_i$;

4) глубина T_i равна $d_i = \lceil \log_{1+\epsilon} 2 \rceil \lceil (\log \log n_i)^2 \rceil$;

5) если p — предок n, то $p \leq 2^{\log^{1+\epsilon} n}$;

6) существует такой алгоритм, что для любой вершины n дерева T_i он выводит ее предка p и число детей p, меньших n, за $poly(n)$ шагов;

7) для любой внутренней вершины v дерева T_i, v имеет $k = \lceil \frac{1}{\lambda(n_i^*)} + 1 \rceil^2$ детей.

Доказательство. Введем обозначение: $\delta = \lceil \log_{1+\epsilon} 2 \rceil$.

Определим T_i как полное сбалансированное дерево глубины d_i. Число листьев в дереве может быть оценено как $k^{d_i} \leq (2^{(\log \log \log n_i^*)^{3c}})^{\delta (\log \log n_i)^2} \leq (2^{(\log \log n_i)^{12c}})^{\delta (\log \log n_i)^2} = 2^{\delta (\log \log n_i)^{24c}} \leq n_i$.

Корень — единственная вершина на нулевом уровне. Существует в точности k^s вершин на уровне s. Пусть $a_{i,j} = 2^{(\log n_i)^{(1+\epsilon)^j}}$, где $j \in \{0, 1, 2, \ldots\}$. Вершины T_i на уровне $(d_i - s)$ — это вершины с номерами $[a_{i,s}; a_{i,s} + k^{d_i - s} - 1]$.

Заметим, что $a_{i,s+1} - a_{i,s} \geq a_{i,1} - a_{i,0} = 2^{(\log n_i)^{(1+\epsilon)}} - n_i \geq 2^{(\log n_i)^{(1+\epsilon)} - 1} > n_i \geq k^{d_i} \geq k^{d_i - s}$. Значит, на всех уровнях хватит места на все вершины.

Предок вершины с номером j на уровне s имеет номер $\lfloor \frac{j}{k} \rfloor$. Пусть $h(n) = n^{\log^{\epsilon} n}$. Так как $h(n+k) \geq h(n) + k$, мы получаем, что $h(2^{\log^{(1+\epsilon)^s} n_i} + j) \geq h(2^{\log^{(1+\epsilon)^s} n_i}) + j \geq 2^{\log^{(1+\epsilon)^{s+1}} n_i} + j/k$, а значит, свойство 5 выполнено. Проверка остальных свойств очевидна. □

Теперь опишем алгоритм, который будет сэмплировать D_n на $n \in \{n_i, \ldots, n_i^*\}$ за $O(f(n))$ шагов.

1) Если $n = n_i^*$, то выведем минимальный $t_i \in \{0, 1, \ldots, 2^{n_i} - 1\}$, такой что для всех $l \in [n_i; m_i]$ выполняется неравенство $\Pr[A_i(1^l) = t_i] < \lambda(n_i)/2$. Такой t_i существует, так как для всех l существует не больше $\frac{2}{\lambda(n_i)}$ элементов z, таких, что $\Pr[A_i(1^l) = z] \geq \lambda(n_i)/2$ и $\frac{2}{\lambda(n_i)} m_i \leq 2^{n_i}$. Такой t_i может быть найден перебором за $m_i c_i g(m_i) 2^{c_i g(m_i)}$ шагов, где c_i — константа зависящая от i и

$$m_i c_i g(m_i) 2^{c_i g(m_i)} \leq 2^{m_i g(m_i)} \leq 2^{2n_i g(2n_i)} \leq 2^{2^{n_i (2n_i)^{2\log^a n_i}}} < 2^{2^{4\log^{(1+a)} n_i}} \leq 2^{2^{2^{4(1+a)\log\log n_i}}} \leq 2^{2^{2^{(\log\log n_i)^2}}} < n_i^* = o(f(n_i^*)).$$

2) Если n не является вершиной T_i, то вернем 0.

3) В противном случае, пусть p — это родитель n и j — это номер n в списке детей p. По свойству T_i, $p \leq 2^{\log^{1+\epsilon} n}$ и p может быть найден за полиномиальное время. Вернем $C^{A_i(1^p)}(p, j, \lambda(n)/2, \lambda(p)/2)$, где C — это алгоритм из леммы 4.3. По лемме 4.3 алгоритм C работает не больше $\mathrm{poly}(p)$ шагов, ответ на запрос к оракулу требует моделировать $A_i(1^p)$, на что, в свою очередь тратится $c_i g(p)$ шагов. Заметим, что $c_i g(p)\mathrm{poly}(p) < 2^{2\log^{a+1} p} < 2^{2\log^{1+a}(2^{\log^{1+\epsilon} n})} = 2^{2\log^{(1+a)(1+\epsilon)} n} < 2^{\log^{(1+b)} n} = f(n)$.

Будем доказывать от противного. Предположим, что для всех $n \in \{n_i, \ldots, n_i^*\}$ статистическое расстояние между D_n и $A_i(1^n)$ меньше $1 - \lambda(n)$. Индукцией по глубине вершины в дереве докажем, что на любом уровне существует вершина v, что $D_v(t_i) \geq 1 - \lambda(v)/2$. Нетрудно видеть, что если v — это лист, то мы получили противоречие с определением t_i.

База индукции следует из определения $D_{n_i^*}$. Докажем индукционный переход. Пусть для уровня s утверждение верно, докажем для $s + 1$. Пусть v — вершина на уровне s, такая, что $D_v(t_i) \geq 1 - \lambda(v)/2$. Если v — это лист, то все доказано. В противном случае $\Pr[A_i(1^v) = t_i] >$

$\lambda(v)/2$, так как $\Delta(D_v, A_i(1^v)) < 1 - \lambda(v)$. Значит, по лемме 4.3 существует такой сын u с номером j (среди всех детей вершины v), что $\Pr[C^{A_i(1^v)}(v, j, \lambda(u)/2, \lambda(v)/2) = t_i] > 1 - \lambda(u)/2$. □

Следующее доказательство, в противоположность доказательству Ватсона, не использует коды, исправляющие ошибку. Это делается за счет того, что мы находим один редкий элемент для всех листьев дерева. Этот трюк невозможен в случае доказательства Ватсона, так как в нем все распределения распределены на константном множестве элементов.

Теперь попробуем объяснить трудности, возникающие на пути усиления этого результата на $g(n) = n^a$ и полиномиальное $f(n)$. Проблема состоит в том, что для неконстантного $\lambda(n)$ дерево T_i должно иметь неконстантную степень (растущую). В то же время в корне дерева мы хотим сделать экспоненциальное относительно номера любого листа количество шагов. Но предок произвольной вершины n должен быть не более чем на полином от нее удален. Соответственно, для любого листа l расстояние между корнем и l как минимум $\Omega(\log l)$. Пусть m_i — это лист с максимальным номером, тогда расстояние между m_i и корнем как минимум $L = \Omega(\log m_i)$. Пусть S — множество таких вершин, что их номера меньше m_i, но номера их родителей больше. Заметим, что все вершины на расстоянии L от корня либо лежат в S, либо их предок лежит в S. Значит, размер S как минимум k_i^L, что больше m_i, противоречие.

4.2.2 Слабая сложность

В этой секции мы рассмотрим постановку данной задачи со слабой сложностью, но с более тонкой иерархией ($f(n) = poly(g(n))$). Начнем с эквивалентной переформулировки.

Предложение 4.1. Следующие условия эквивалентны:

1) Существуют монотонно стремящиеся к нулю функции $\beta(n)$ и $\alpha(n) =$

$o(\beta(n))$, такие, что для любого $a > 0$ существуют ансамбль распределений $D \in \mathbf{PSamp}$ и язык L, удовлетворяющие условиям:

- $(L, F) \in \mathrm{Heur}_{\alpha(n)}\mathbf{P}$ для всех $F \in \mathbf{DSamp}[n^a]$;
- $(L, D) \notin \mathrm{Heur}_{\beta(n)}\mathbf{P}$.

2) Существуют такие бесконечно малые функции $\beta(n)$ и $\alpha(n) = o(\beta(n))$, что для любого $a > 0$ существует ансамбль распределений $D \in \mathbf{PSamp}$, возрастающая последовательность целых чисел l_n и последовательность множеств $S_n \subseteq \{0,1\}^{l_n}$, что следующие условия выполнены:

- $D(S_n) > \beta(l_n)$ для всех n;
- Для всякого $F \in \mathbf{DSamp}[n^a]$ для бесконечно многих n выполняется $F(S_n) \leq \alpha(n)$.

3) Существуют такие бесконечно малые (монотонно стремящиеся к нулю) функции $\beta(n)$ и $\alpha(n) = o(\beta(n))$, что для всех $a > 0$ существуют такой ансамбль распределений $D \in \mathbf{PSamp}$ и такой язык L, что следующее условие выполнено:

- Существует такой линейный по времени алгоритм A, что для всех $F \in \mathbf{DSamp}[n^a]$ и достаточно больших n выполняется неравенство $\Pr\limits_{x \leftarrow F_n}[L(x) = A(x)] \geq 1 - \alpha(n)$;
- $(L, D) \notin \mathrm{Heur}_{\beta(n)}\mathbf{R}$.

Доказательство. Заметим, что если $\alpha(n) = o(\beta(n))$, то $\alpha(n) = o(\sqrt{\alpha(n)\beta(n)})$ и $\sqrt{\alpha(n)\beta(n)} = o(\beta(n))$.

$1 \to 2$. Применим утверждение 1 к $a' = 2a$. Пусть A_i — это перечисление всех детерминированных алгоритмов с будильником $n^{1.5a}$. Будем рассматривать A_i как генераторы распределений и интерпретировать результат их работы как строки из $\{0,1\}^n$ каким-то фиксированным образом. Пусть F — это алгоритм, который сэмплирует ансамбль распределений следующим образом: на входе 1^n с вероятностью $\frac{1}{2}$ он возвра-

щает результат работы $A_1(1^n)$, с вероятностью $\frac{1}{2^2}$ — результат работы $A_2(1^n)\ldots$, с вероятностью $\frac{1}{2^{n-1}}$ — результат работы $A_{n-1}(1^n)$ и с вероятностью $\frac{1}{2^{n-1}}$ — результат работы $A_n(1^n)$. Пусть F определяет ансамбль распределений E. Очевидно, что $E \in \mathbf{DSamp}[n^{a'}]$. Условие 1 влечет, что $(L, E) \in \mathrm{Heur}_{\alpha(n)}\mathbf{P}$. Пусть T разрешает (L, E) в классе $\mathrm{Heur}_{\alpha(n)}\mathbf{P}$. Обозначим $S_n = \{x \in \{0,1\}^n \mid T(x) \neq L(x)\}$. Значит, $E_n(S_n) \leq \alpha(n)$ для всех n. Таким образом, для любого $R \in \mathbf{DSamp}[n^a]$ существует такая константа C, что $R_n(S_n) \leq C\alpha(n)$, что ,в свою очередь, меньше чем $\sqrt{\alpha(n)\beta(n)}$ (для достаточно больших n). Так как $(L, D) \notin \mathrm{Heur}_{\beta(n)}\mathbf{P}$, мы знаем, что $D(S_n) > \beta(n)$. В итоге мы нашли такие последовательности натуральных чисел l_n и множеств $S_n \subseteq \{0,1\}^{l_n}$, что выполнены следующие условия:

- $D(S_n) > \beta(l_n)$ для всех n;
- Для всех $R \in \mathbf{DSamp}[n^a]$ и достаточно больших n выполняется неравенство $R(S_n) \leq \sqrt{\alpha(n)\beta(n)}$.

$2 \to 3$. Доказательство аналогично концу доказательства леммы 4.1. Применим утверждение 2 к $a' = a+1$. Пусть A_i — это перечисление всех алгоритмов с будильником $O(n^a)$; будем, как обычно, рассматривать их как генераторы ансамблей распределений. Пусть F — это алгоритм, который сэмплирует некоторый ансамбль распределений следующим образом: на входе 1^n с вероятностью $\frac{1}{2}$ возвращает $A_1(1^n)$, с вероятностью $\frac{1}{2^2}$ возвращает $A_2(1^n)$, $\ldots$, с вероятностью $\frac{1}{2^{n-1}}$ возвращает $A_{n-1}(1^n)$ и с вероятностью $\frac{1}{2^{n-1}}$ возвращает $A_n(1^n)$. Пусть E — это ансамбль распределений, сэмплируемый F. Очевидно, что $E \in \mathbf{DSamp}[n^{a+1}]$. Пусть D и S_n — из утверждения 2 для $a' = a+1$. Определим такое множество $I = \{n_1, n_2, \ldots\}$, что оно состоит из всех таких n, что $E_n(S_n) \leq \alpha(n)$.

Определим язык $L \subseteq \bigcup\limits_{n \in I} S_n$. Пусть T_i — это перечисление всех алгоритмов. Определим так L, что для любого $x \in S_{n_k}$, $x \in L$ тогда и только тогда, когда T_k не останавливается или отвергает на входе x. По определению $(L, D) \notin \mathrm{Heur}_{\beta(n)}\mathbf{R}$.

Рассмотрим алгоритм, который возвращает 0 на всех входах. Если $R \in \mathbf{DSamp}[n^a]$, то существует такое i, что A_i сэмплирует R. Для $n \geq i$ для любого множества $S \subseteq \{0,1\}^n$ верно следующее неравенство: $E(S) \geq 2^{-i}R(S)$. Значит, для любого ансамбля R из $\mathbf{DSamp}[n^a]$ этот алгоритм имеет ошибку не больше $c\alpha(n)$ (где c — это константа, зависящая только от ансамбля R); $c\alpha(n) < \sqrt{\alpha(n)\beta(n)}$ для достаточно больших n.

$3 \to 1$. Это следствие очевидно. □

Мы докажем утверждение, чуть более слабое, чем утверждение 2 из предложения 4.1. Фактически, мы докажем утверждение 2 при $\alpha(n) = \beta(n) = \frac{1}{n^b}$. Данное рассуждение несложно обобщить на случай других бесконечно убывающих функций.

Теорема 4.2. Для всех целых $a > 0$ и $b > 0$ существуют такие ансамбль распределений $D \in \mathbf{PSamp}$, последовательность целых чисел l_n и последовательность множеств $S_n \subseteq \{0,1\}^{l_n}$, что следующие условия выполнены:

- $D(S_n) > \frac{1}{l_n^b}$ для всех n;
- для любого $F \in \mathbf{DSamp}[n^a]$ выполнено $F(S_n) \leq \frac{1}{l_n^b}$ для бесконечно многих n.

Начнем с неформального объяснения доказательства. Для простоты сначала докажем оценку $\frac{1}{2}$ вместо $\frac{1}{l_n^b}$. Будем использовать отложенную диагонализацию. Рассмотрим целочисленные последовательности n_i и n_i^*, такие, что $n_1 = 1$, $n_i = n_i^* + 1$, $n_i^* = 2^{n_i^a}$. Пусть F_i — это такое перечисление всех вероятностных алгоритмов с будильником n^{a+1}, что каждый алгоритм встречается бесконечно много раз в этом перечислении. Будем рассматривать F_i как генераторы распределений.

Обозначим за $T_{0,n}$ и $T_{1,n}$ множества бинарных строк длины n, начинающиеся с 0 и 1 соответственно. Рассмотрим следующий генератор распределения D_n: если $n = n_i^*$, найдем такое $t_i \in \{0,1\}$, что

$\Pr[F_i(1^{n_i}) \in T_{t_i,n_i}] \leq \frac{1}{2}$ и вернем случайный элемент из T_{t_i,n_i^*}; если $n_i \leq n < n_i^*$, то запустим $F_i(1^{n+1})$, и если строка, которую он вернул, начинается с $s \in \{0,1\}$, вернем случайный элемент из $T_{s,n}$.

Предположим, что существует такое распределение $E \in \mathbf{DSamp}[n^a]$, что для всех $n > n_0$, если для некоторого $S \subseteq \{0,1\}^n$ выполнено неравенство $\Pr[D_n \in S] > \frac{1}{2}$, то $\Pr[F_i(1^n) \in S] > \frac{1}{2}$. Пусть F_i — это генератор для этого распределения E и $i > n_0$. Мы знаем, что $\Pr[D_{n_i^*} \in T_{t_i,n_i^*}] = 1$, а значит, $\Pr[F_i(1^{n_i^*}) \in T_{t_i,n_i^*}] > \frac{1}{2}$, значит, $\Pr[D_{n_i^*-1} \in T_{t_i,n_i^*-1}] > \frac{1}{2}$ и так далее. В итоге мы получим, что $\Pr[F_i(1^{n_i}) \in T_{t_i,n_i}] > \frac{1}{2}$, а это противоречит определению t_i.

Для доли ошибки $\frac{1}{k}$ доказательство будет аналогично, но мы разобьем $\{0,1\}^n$ на k частей. Однако возникнет новая проблема, заключающаяся в том, что при доле ошибки $\frac{1}{n^b}$ на разных длинах будет разное число отрезков; поэтому мы будем рассматривать деревья интервалов вместо цепей.

Доказательство теоремы 4.2. Рассмотрим такое перечисление всех вероятностных алгоритмов F_i с будильником n^{a+1}, что каждый алгоритм встречается бесконечно много раз в этом перечислении. Будем рассматривать F_i как генераторы для распределений. Рассмотрим целочисленные последовательности n_i и n_i^*, такие, что $n_1 = 1$, $n_i = 2n_i^*$, $n_i^* = 2^{n_i^a}$.

Разобьем множество строк длины n на n^b непустых подмножеств; будем называть их интервалами и обозначать их $T_{j,n}$ для $j \in \{1, 2, \ldots, n^b\}$. Определим следующий граф (этот граф будет лесом):

- множество вершин графа — это множество всех интервалов $T_{j,n}$ для $n = 2^k$, $n_i \leq n \leq n_i^*$ и $j \in \{1, 2, \ldots, n^b\}$;
- все интервалы T_{j,n_i} — это корни деревьев;
- для $n \in \{n_i, 2n_i, 4n_i, \ldots, n_i^*/2\}$, $T_{j,n}$ имеет 2^b детей: $\{T_{j',2n} \mid 2^b(j-1) \leq j' \leq 2^b j - 1\}$.
- все интервалы T_{j,n_i^*} являются листьями леса.

Определим генератор D следующим образом. На входе 1^n:

- Если $n = n_i^*$ для некоторого i, то найдем интервал T_{j,n_i} с минимальной вероятностью в соответствии с $F_i(1^{n_i})$ (нетрудно увидеть, что $\Pr[F_i(1^{n_i}) \in T_{j,n_i}] \leq \frac{1}{n_i^b}$). Если таких несколько, то выберем с минимальным j (заметим, что это может быть сделано за $poly(n_i^*)$ шагов простым перебором). Затем выберем случайного потомка T_{j,n_i} среди листьев и вернем случайную строку из этого потомка.

- Если $n_i \leq n < n_i^*$ для некоторого i, то запустим $F_i(1^{2n})$, и если он вернул строку, принадлежащую предку $T_{k,n}$ для некоторого k, то вернем случайную строку из $T_{k,n}$.

Докажем, что для любого i существуют такие j и $n \in [n_i; n_i^*]$, что $\Pr[D_n \in T_{j,n}] > \frac{1}{n^b}$ и $\Pr[F_i(1^n) \in T_{j,n}] \leq \frac{1}{n^b}$ (из этого последует условие теоремы). Предположим обратное, что для всех j и $n \leq n_i^*$ если $\Pr[F_i(1^n) \in T_{j,n}] \leq \frac{1}{n^b}$, то $\Pr[D_n \in T_{j,n}] < \frac{1}{n^b}$. Пусть T_{j,n_i} — это интервал с наименьшей вероятностью в соответствии с распределением $F_i(1^{n_i})$, тогда $\Pr[F_i(1^{n_i}) \in T_{j,n_i}] \leq \frac{1}{n_i^b}$. Докажем индукцией по l, что для всех $n = 2^l n_i$ (и $n \leq n_i^*$) существует такое k, что $T_{k,n}$ — это потомок T_{j,n_i} и $\Pr[D_n \in T_{k,n}] \leq \frac{1}{n^b}$. База при $l = 0$ очевидна. Докажем индукционный переход от l к $l+1$. Пусть $n = 2^l n_i$. Предположим, что $\Pr[D_n \in T_{k,n}] \leq \frac{1}{n^b}$, тогда по принципу Дирихле и построению D существует интервал $T_{k',2n}$ являющийся одним из 2^b сыновей $T_{k,n}$, что $\Pr[F_i(1^{2n}) \in T_{k',2n}] \leq \frac{1}{(2n)^b}$. В этом случае $\Pr[D_{2n} \in T_{k',2n}] \leq \frac{1}{(2n)^b}$. Таким образом, доказали, что существует такое k, что $\Pr[D_{n_i^*} \in T_{k,n_i^*}] \leq \frac{1}{(n_i^*)^b}$ и T_{k,n_i^*} потомок T_{j,n_i}. Но по построению D любой потомок T_{j,n_i} на длине n_i^* имеет вероятность ровно $\frac{n_i^b}{(n_i^*)^b} > \frac{1}{(n_i^*)^b}$, противоречие. □

Следствие 4.3. Для любого $a > 0$ и $b > 0$ существуют такие ансамбль распределений $D \in \mathbf{PSamp}$, язык L и линейный по времени алгоритм A, что следующие условия выполнены:

- $\Pr_{x \leftarrow F_n}[A(x) \neq L(x)] = O(\frac{1}{n^b})$ для всех $F \in \mathbf{DSamp}[n^a]$;

- $(L, D) \notin \mathrm{Heur}_{\frac{1}{n^b}}\mathbf{R}$.

Доказательство. Данное следствие следует из теоремы 4.2 по соображениям, аналогичным доказательству следствия $2 \to 3$ в предложении 4.1. □

4.3 Вычислимые распределения

Лемма 4.5. Если ансамбль распределений D сэмплируем за полиномиальное время и для всех n распределение D_n сконцентрировано на одном элементе, то $D \in \mathbf{PComp}$.

Доказательство. Для того, чтобы вычислить функцию распределения, достаточно найти элемент x_0 с вероятностью относительно D равной 1 и сравнить данный вход с x_0. Если мы запустим сэмплирующий алгоритм со всеми нулями вместо его случайных битов, то он вернет x_0. □

Предложение 4.2. Для любого $a > 0$ существует такой ансамбль распределений $D \in \mathbf{PComp}$, что для любого ансамбля $F \in \mathbf{Comp}[n^a]$ существует бесконечно много таких n, что $\Delta(D_n, F_n) \geq 1 - 2^{-n}$.

Доказательство. Пусть A_i — это такое перечисление всех детерминированных алгоритмов с будильником n^{a+1}, что алгоритм A_i встречается в этой последовательности бесконечно много раз. Мы будем интерпретировать A_i как алгоритм, вычисляющий функцию распределения. Однако в этой последовательности могут встречаться и некорректные, не задающие никакое распределение. Если A_i соответствует какому-то распределению F, то мы определим такое распределение D_i, что $\Delta(D_i, F_i) \geq 1 - 2^{-i}$. Для этого мы сконцентрируем D_i на строке, вероятность которой относительно F_i не больше 2^{-i}. По лемме 4.5 такое распределение окажется также и вычислимым.

Теперь опишем, как находить элемент с вероятностью, не большей 2^{-i} относительно F_i. Будем использовать бинарный поиск. Если A_i не

соответствует никакому распределению, то мы либо поймем это в процессе бинарного поиска, в таком случае мы остановимся и вернем 0^i, либо мы найдем такой $x \in \{0,1\}^i$, что $A_i(x) - A_i(x') \leq 2^{-i}$ (где x' лексикографический предшественник x, а если $x = 0^i$, то будем считать, что $A_i(x') = 0$). □

Теперь докажем свойство, эквивалентное свойству иерархии распределенных задач для функций n^a и n^b, но для вычислимых распределений.

Теорема 4.3. Для любого $a > 0$ существуют такие язык L и ансамбль распределений $D \in \mathbf{PComp}$, что

- для всех $F \in \mathbf{Comp}[n^a]$ существует такая константа $c > 0$, что $(L, F) \in \mathrm{Heur}_{\frac{c}{2^n}}\mathbf{DTime}[n]$;
- $(L, D) \notin \mathrm{Heur}_{1-\frac{1}{2^{n-1}}}\mathbf{R}$.

Доказательство. Мы не можем просто повторить доказательство леммы 4.1 несмотря на то, что мы уже доказали предложение 4.2. Причина в том, что не всякий алгоритм вычисляет какую-то функцию распределения (например, он может вычислять не монотонную функцию) и неизвестно как эффективно проверить, что алгоритм вычисляет функцию распределения.

Пусть A_i — это перечисление всех детерминированных алгоритмов с будильником Cn^a, где C — это некоторая константа. Мы будем интерпретировать их как алгоритмы, вычисляющие функции распределения (однако будем помнить, что некоторые из этих алгоритмов некорректны). Будем интерпретировать результат $A_i(x)$ как рациональное число между 0 и 1.

Для каждого n мы покажем, что за полиномиальное время возможно найти такой $x_n \in \{0,1\}^n$, что если $i \in \{1, 2, \ldots, n\}$ и A_i вычисляет функцию распределения, то вероятность x_n относительно A_i не больше 2^{i-n}. Распределение D_n будет сосредоточено на x_n; получившийся ансамбль

будет вычислим за полиномиальное время по лемме 4.5. Если мы нашли такой x_n, то мы можем определить L подобно тому, как он определялся в лемме 4.1. Точнее говоря, выберем L так, чтобы $L \subseteq \bigcup_n \{x_n\}$ и $x_n \in L$ тогда и только тогда, когда алгоритм с номером n в перечислении всех алгоритмов отвергает x_n. Для всех $F \in \mathbf{Comp}[n^a]$ алгоритм, который возвращает 0 на всех входах, распознает (L, F) в $\mathrm{Heur}_{2^{i-n}}\mathbf{DTime}[n]$, если F вычислимо алгоритмом A_i из нашего перечисления. По построению $(L, D) \notin \mathrm{Heur}_{1-\frac{1}{2^n-1}}\mathbf{P}$.

Теперь опишем процедуру поиска x_n. Изначально $I = \{1, 2, \ldots, n\}$, Мы будем удалять i из I, если обнаружим, что A_i не вычисляет функцию распределения на $\{0,1\}^n$. На каждой итерации мы определим $F(x) = \sum_{i \in I} \frac{1}{2^i} A_i(x)$. Бинарным поиском попробуем найти такой $x \in \{0,1\}^n$, что $F(x) - F(x') \leq 2^{-n}$ (где x' — это лексикографический предок x и $F(x') = 0$, если $x = 0^n$). Если бинарный поиск работает успешно, то x_n сделаем равным x. Если бинарный поиск нашел немонотонность $F(x)$, найдем такое $i \in I$, что A_i в этой точке не монотонен и удалим ее из I, а затем начнем новую итерацию. Если оказалось, что I пусто, то положим $x_n = 0^n$. Теперь заметим, что если A_i вычисляет функцию распределения, то вероятность x_n относительно этого распределения не больше 2^{i-n}. $\square$

Заключение

В данной работе исследована эвристическая схемная сложность полиномиальных в среднем протоколов Мерлин–Артур, доказан ряд эвристических иерархий по времени для семантических вычислений и для вероятностных вычислений, а также исследован вопрос повышения сложности языка при смене распределения с простого на сложное.

Наиболее естественным вопросом в качестве продолжения исследований первой темы является вопрос о существовании нижних оценок на эвристическую схемную сложность полиномиальных протоколов Мерлин–Артур. Другим важным вопросом, и скорее всего более простым вопросом, в данной области является вопрос существования нижних оценок на эвристическую схемную сложность с параметром ошибки, экспоненциально близким к $\frac{1}{2}$, для полиномиальных в среднем протоколов Мерлин–Артур.

Логичным продолжением второй темы, эвристических иерархий по времени, является вопрос существования иерархии по времени для вероятностных вычислений с ограниченной ошибкой (справедливость $\mathbf{BPP} \not\subseteq \mathbf{BPTime}[n^k]$ для всех k), однако, по-видимому, это и самый сложный вопрос в данной оболасти. Тем не менее, не все и более простые вопросы на данный момент решены. Так, все еще не известно даже эвристической иерархии для вероятностных вычислений с односторонней ошибкой. Также было бы интересно доказать возможность улучшений параметров иерархии, но без перехода от языков к функциям.

Наконец, в третьей теме все еще открытым остается вопрос о более точной иерархии по времени сэмплировния с расстоянием, стремящимся к нулю. Поэтому не известно, выполняется ли для каких-нибудь много-

членов свойство иерархии распределенных задач с параметрами, стремящимися к нулю.

Литература

[1] Hartmanis J., Stearns R. E. On the Computational Complexity of Algorithms // Journal of Symbolic Logic. — 1967. — Vol. 32, no. 1. — P. 120–121.

[2] Cook S. A. A Hierarchy for Nondeterministic Time Complexity // J. Comput. Syst. Sci. — 1973. — Vol. 7, no. 4. — P. 343–353.

[3] Zak S. A Turing machine time hierarchy // Theoretical Computer Science. — 1983. — Vol. 26, no. 3. — P. 327–333.

[4] Fortnow L., Santhanam R. Hierarchy Theorems for Probabilistic Polynomial Time // 45th Symposium on Foundations of Computer Science (FOCS 2004), 17-19 October 2004, Rome, Italy, Proceedings. — 2004. — P. 316–324.

[5] Pervyshev K. On Heuristic Time Hierarchies // IEEE Conference on Computational Complexity. — 2007. — P. 347–358.

[6] Kannan R. Circuit-Size Lower Bounds and Non-Reducibility to Sparse Sets // Information and Control. — 1982. — Vol. 55, no. 1-3. — P. 40–56.

[7] Karp R. M., Lipton R. J. Some Connections between Nonuniform and Uniform Complexity Classes // Proceedings of the 12th Annual ACM Symposium on Theory of Computing, April 28-30, 1980, Los Angeles, California, USA. — 1980. — P. 302–309.

[8] Jin-Yi Cai. $S^p_2 \subseteq ZPP^{NP}$ // 42nd Annual Symposium on Foundations of Computer Science, FOCS 2001, 14-17 October 2001, Las Vegas, Nevada, USA. — 2001. — P. 620–629.

[9] Santhanam R. Circuit Lower Bounds for Merlin–Arthur Classes // SIAM J. Comput. — 2009. — Vol. 39, no. 3. — P. 1038–1061.

[10] Li M., Vitanyi P. M. Average Case Complexity under the Universal Distribution Equals Worst Case Complexity // Information Processing Letters. — 1992. — Vol. 42. — P. 145–149.

[11] Gurevich Y., Shelah S. Expected Computation Time for Hamiltonian Path problem // SIAM J. Comput. — 1987. — Vol. 16, no. 3. — P. 486–502.

[12] Find M. G., Golovnev A., Hirsch E. A., Kulikov A. S. A Better-Than-3n Lower Bound for the Circuit Complexity of an Explicit Function // IEEE 57th Annual Symposium on Foundations of Computer Science, FOCS 2016, 9-11 October 2016, Hyatt Regency, New Brunswick, New Jersey, USA. — 2016. — P. 89–98.

[13] Iwama K., Morizumi H. An Explicit Lower Bound of 5n - o(n) for Boolean Circuits // Mathematical Foundations of Computer Science 2002, 27th International Symposium, MFCS 2002, Warsaw, Poland, August 26-30, 2002, Proceedings. — 2002. — P. 353–364.

[14] Buhrman H., Fortnow L., Thierauf T. Nonrelativizing Separations // Proceedings of the 13th Annual IEEE Conference on Computational Complexity, Buffalo, New York, USA, June 15-18, 1998. — 1998. — P. 8–12.

[15] Barak B. A Probabilistic-Time Hierarchy Theorem for "Slightly Non-uniform" Algorithms // Randomization and Approximation Techniques, 6th International Workshop, RANDOM 2002, Cambridge,

MA, USA, September 13-15, 2002, Proceedings. — 2002. — P. 194–208.

[16] Karpinski M., Verbeek R. Randomness, Provability, and the Seperation of Monte Carlo Time and Space // Computation Theory and Logic, In Memory of Dieter Rödding. — 1987. — P. 189–207.

[17] van Melkebeek D., Pervyshev K. A Generic Time Hierarchy with One Bit of Advice // Computational Complexity. — 2007. — Vol. 16, no. 2. — P. 139–179.

[18] Watson T. Time hierarchies for sampling distributions // Innovations in Theoretical Computer Science, ITCS '13, Berkeley, CA, USA, January 9-12, 2013. — 2013. — P. 429–440.

[19] Babai L., Erdos P., Selkow S. RANDOM GRAPH ISOMORPHISM // SIAM J. Comput. — 1980. — Vol. 9, no. 3. — P. 628–635.

[20] van Dama E., Muzychuk M. Some implications on amorphic association schemes // Journal of Combinatorial Theory, Series A. — 2010. — Vol. 117. — P. 111–127.

[21] Ben-David S., Chor B., Goldreich O., Luby M. On the theory of average case complexity // J. Comput. Syst. Sci. — 1992. — Vol. 44, no. 2. — P. 193–219.

[22] Knop A. Circuit Lower Bounds for Average-Case MA // Computer Science - Theory and Applications - 10th International Computer Science Symposium in Russia, CSR 2015, Listvyanka, Russia, July 13-17, 2015, Proceedings. — 2015. — P. 283–295.

[23] Itsykson D., Knop A., Sokolov D. Heuristic Time Hierarchies via Hierarchies for Sampling Distributions // Algorithms and Computation - 26th International Symposium, ISAAC 2015, Nagoya, Japan, December 9-11, 2015, Proceedings. — 2015. — P. 201–211.

[24] Itsykson D., Knop A., Sokolov D. Complexity of distributions and average-case hardness // Algorithms and Computation - 27th International Symposium, ISAAC 2016, Australia, Sydney, December 12-14, 2016, Proceedings. — 2016. — P. 38:1–38:12.

[25] Knop A. Circuit Lower Bounds for Heuristic MA // Electronic Colloquium on Computational Complexity (ECCC). — 2013. — Vol. 20. — P. 37. — URL: http://eccc.hpi-web.de/report/2013/037.

[26] Itsykson D., Knop A., Sokolov D. Heuristic time hierarchies via hierarchies for sampling distributions // Electronic Colloquium on Computational Complexity (ECCC). — 2014. — Vol. 21. — P. 178. — URL: http://eccc.hpi-web.de/report/2014/178.

[27] Itsykson D., Knop A., Sokolov D. Complexity of distributions and average-case hardness // Electronic Colloquium on Computational Complexity (ECCC). — 2015. — Vol. 22. — P. 174. — URL: http://eccc.hpi-web.de/report/2015/174.

[28] Impagliazzo R. A Personal View of Average-Case Complexity // Proceedings of the Tenth Annual Structure in Complexity Theory Conference, Minneapolis, Minnesota, USA, June 19-22, 1995. — 1995. — P. 134–147.

[29] Itsykson D. Structural complexity of AvgBPP // Ann. Pure Appl. Logic. — 2010. — Vol. 162, no. 3. — P. 213–223.

[30] Trevisan L., Vadhan S. P. Pseudorandomness and Average-Case Complexity Via Uniform Reductions // Computational Complexity. — 2007. — Vol. 16, no. 4. — P. 331–364.

[31] Bennett C. H., Gill J. Relative to a Random Oracle A, P^A != NP^A != co-NP^A with Probability 1 // SIAM J. Comput. — 1981. — Vol. 10, no. 1. — P. 96–113.

[32] Arora S., Barak B. Computational Complexity - A Modern Approach. — Cambridge University Press, 2009. — ISBN: 978-0-521-42426-4.

[33] Trevisan L., Vadhan S. P. Pseudorandomness and Average-Case Complexity via Uniform Reductions // Proceedings of the 17th Annual IEEE Conference on Computational Complexity, Montréal, Québec, Canada, May 21-24, 2002. — 2002. — P. 129–138.

[34] Bogdanov A., Trevisan L. Average-Case Complexity // Foundation and Trends in Theoretical Computer Science. — 2006. — Vol. 2, no. 1. — P. 1–106.

[35] Shamir A. IP = PSPACE // J. ACM. — 1992. — Vol. 39, no. 4. — P. 869–877.

[36] Itsykson D., Sokolov D. On fast non-deterministic algorithms and short heuristic proofs. // Fundamenta Informaticae. — 2014. — Vol. 132. — P. 113–129.

[37] Aaronson S., Wigderson A. Algebrization: A New Barrier in Complexity Theory // TOCT. — 2009. — Vol. 1, no. 1.

[38] Håstad J., Impagliazzo R., Levin L. A., Luby M. A Pseudorandom Generator from any One-way Function // SIAM J. Comput. — 1999. — Vol. 28, no. 4. — P. 1364–1396.

[39] Zachos S. Probabilistic Quantifiers and Games // J. Comput. Syst. Sci. — 1988. — Vol. 36, no. 3. — P. 433–451.

[40] Goldreich O. Studies in Complexity and Cryptography. Miscellanea on the Interplay between Randomness and Computation. — Springer, 2011. — Vol. 6650 of Lecture Notes in Computer Science. — ISBN: 978-3-642-22669-4.