Конспект к лекции 1. (Казань, 4 апреля 2017 г.)

1 Используемые обозначения

Для неориентрированных графов мы используем обозначение $G = (V, E)$, где V есть множество вершин, а E — множество рёбер. При этом мы допускаем графы с петлями и с кратными (параллельными) рёбрами[1].

Если A и B являются подмножествами (возможно, пересекающимися) вершин графа, мы обозначаем $E(A, B)$ множество рёбер, у которых один из концов принадлежит A, а второй B. Также мы обозначаем $\Gamma(v)$ множество соседей вершины v (множество всех вершин w, соединённых с v ребром). Аналогичное обозначение используется для множеств вершин: если A есть подмножество вершин графа, то $\Gamma(A)$ обозначает множество всех соседей A, т.е.,

$$\Gamma(A) = \bigcup_{v \in A} \Gamma(v).$$

Векторы-строки мы обозначаем $\mathbf{x} = (x_1, \ldots, x_n)$. Транспонирование матрицы M мы обозначаем $M^{\perp}$; в частности, если $\mathbf{x} = (x_1, \ldots, x_n)$, то соответствующий вектор-столбец обозначается

$$\mathbf{x}^{\perp} = \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{pmatrix}.$$

2 Комбинаторное определение экспандера

В этой главе мы даём базовое определение экспандера и докажем существование графов, удовлетворяющих этому определению.

Мы определим экспандер как однородный граф со свойством вершинного расширения (потребуем, чтобы у каждого не сишком большого множества вершин графа имелось достаточно много соседей). Сформулируем определение более точно.

Определение 2.1 *Граф $G = (V, E)$ называется однородным комбинаторным (n, d, ε)-экспандером (расширяющим графом), если $|V| = n$ (в графе n вершин), степени всех вершин равны d (допускаются кратные ребра и петли), и выполняется следующее свойство* вершинного расширения*: для любого множества $S \subset V$, $|S| \leqslant n/2$ множество соседей S достаточно велико: $|\Gamma(S)| > (1 + \varepsilon)|A|$.*

[1] Для графа с кратными рёбрами правильнее называть E не множеством, а *мультимножеством* рёбер, поскольку для каждой пары вершин в E может содержаться больше одного ребра с данными концами.

Замечание 1: Чем больше значение ε в определении 2.1, тем более сильное свойство требуется от графа.

Замечание 2: Степень вершины графа — это число рёбер, для которых данная вершина является концом. Это определение распространяется и на петли (ребра, у которых концы совпадают). Таким образом, если некоторой вершине инцидентны d_1 рёбер, не являющихся петлями, и ещё d_2 петель, то степень этой вершины равна $d_1 + d_2$ (каждая петля учитывается с кратностью один, как и всякое другое ребро).

Теорема 2.1 *Пусть ε – некоторое положительное число меньшее 1. Тогда для всех достаточно больших четных d и всех n существует однородный (n, d, ε)-экспандер.*

Доказательство: Мы выберем граф случайно и покажем, что с положительной (и даже довольно близкой к 1) вероятностью такой граф оказывается экспандером. Отсюда будет следовать, что экспандеры существуют.

Прежде всего, нам нужно уточнить, что означает *случайный выбор графа.* Другими словами, нужно зафиксировать распределение вероятностей на графах. Мы выберем случайно $d/2$ перестановок π_i на множестве вершин графа,

$$\pi_i \ : \ \{1, \ldots, n\} \to \{1, \ldots, n\}, \ i = 1, \ldots, d/2.$$

(Каждая перестановка π_i выбирается среди $n!$ равновероятных вариантов; при этом все $d/2$ перестановок выбираются незамисимо друг от друга.) Ребрами графа будем считать все (неупорядоченные) пары вершин $\{v, \pi_i(v)\}$. Таким образом, из каждой вершины v выходит $d/2$ рёбер $\{v, \pi_i(v)\}$ и ещё $d/2$ рёбер $\{v, \pi_i^{-1}(v)\}$. У перестановок могут быть неподвижные точки (перестановка может оставлять некоторые вершины на месте), так что в случае $v = \pi_i(v)$ мы получаем петлю — ребро, оба конца которой совпадают с v. Чтобы степень каждой вершины была равна d, мы будем учитывать каждую петлю дважды.

Отметим, что в случайно выбранном графе с положительной вероятностью появляются кратные рёбра (поскольку одно и то же ребро $\{v, \pi_i(v)\}$ может получаться из нескольких перестановок π_i).

Теперь оценим вероятность того, что полученный в результате граф *не* окажется экспандером. Согласно определению, граф не является экспандером, если найдется множество вершин S (состоящее из не более, чем $n/2$ вершин), все соседи которого лежат в некотором множестве T, состоящем из $\lfloor (1+\varepsilon)|S| \rfloor$ вершин.

Зафиксируем некоторые множества вершин S и T. Зафиксируем номер перестановки π_i. Вероятность того, что для каждой вершины $v \in S$ второй конец ребра $\{v, \pi_i(v)\}$ попадёт в T, равна

$$\frac{|T|}{n} \cdot \frac{|T|-1}{n-1} \cdot \ldots \cdot \frac{|T|-|S|+1}{n-|S|+1} \leqslant \left(\frac{|T|}{n}\right)^{|S|}.$$

Поскольку мы выбираем $d/2$ перестановок независимо, вероятность того, что данное событие произойдёт для всех i, не превосходит $\left(\frac{|T|}{n}\right)^{(d/2)\cdot|S|}$. Таким образом,

$$\text{Prob[свойство экспандерности графа нарушено]} \leqslant \sum_{S,T} \left(\frac{|T|}{n}\right)^{(d/2)\cdot|S|},$$

где суммирование происходит по всем множествам вершин S размера не более $n/2$ и по всем множествам T размера $\lfloor(1+\varepsilon)|S|\rfloor$.

На самом деле интересующая нас вероятностью ещё меньше — чтобы свойство экспандерности нарушилось, для каждой вершины $v \in S$ все рёбра вида $\{v, \pi_i^{-1}(v)\}$ также должны попасть в T. Но мы не будем этого учитывать; рёбер вида $\{v, \pi_i(v)\}$ уже достаточно, чтобы получить нужную нам оценку на вероятность «неприятного» события.

Оценим интересующую нас сумму:

$$\sum_{S,T} \left(\frac{|T|}{n}\right)^{(d/2)\cdot|S|} \leqslant \sum_{s=1}^{n/2} C_n^s \cdot C_n^{(1+\varepsilon)s} \cdot \left(\frac{(1+\varepsilon)s}{n}\right)^{sd/2}. \tag{1}$$

Каждый биномиальный коэффициент C_n^k можно оценить сверху величиной $\left(\frac{ne}{k}\right)^k$ (см. упражнение 2.1 ниже). Таким образом, сумма (1) не превосходит

$$\sum_{s=1}^{n/2} \left(\frac{ne}{s}\right)^s \cdot \left(\frac{ne}{(1+\varepsilon)s}\right)^{(1+\varepsilon)s} \cdot \left(\frac{(1+\varepsilon)s}{n}\right)^{sd/2} =$$

$$= \sum_{s=1}^{n/2} \left[(s/n)^{d/2-2-\varepsilon} \cdot (1+\varepsilon)^{d/2} \cdot \frac{e^{1+\varepsilon}}{(1+\varepsilon)^{1+\varepsilon}}\right]^s. \tag{2}$$

Остаётся заметить, что $s \leqslant n/2$, а $1+\varepsilon < 2$. Таким образом, можно подобрать такое $d = d(\varepsilon)$, чтобы выражение в квадратных скобках в правой части (2) было меньше $1/2$ при всех значениях s. Следовательно, сумма (1) меньше единицы. А это и означает, что с положительной вероятностью случайный граф является (n, d, ε)-экспандером. Теорема доказана.

Упражнение 2.1 *Докажите оценку для биномиальных коэффициентов, которую мы использовали в доказательстве теоремы 2.1:*

$$C_n^k \leqslant \left(\frac{ne}{k}\right)^k,$$

где e — основание натурального логарифма.

Упражнение 2.2 *Оцените асимптотику зависимости $d = d(\varepsilon)$ в теореме 1: насколько большой должна быть степень графа, чтобы гарантировать существование экспандера с данным параметром расширения ε?*

Упражнение 2.3 *Докажите , что утверждение теоремы 2.1 выполнено для графов без петель: для любого $\varepsilon < 1$, для всех достаточно больших чётных d и всех n существует однородный комбинаторный (n, d, ε)-экспандер без петель.*

Следующее упражнение показывает, что $d = 3$ есть минимальная степень графа, для которой определение однородного комбинаторного экспандера имеет смысл.

Упражнение 2.4 *(а) Докажите, что для некоторого $\varepsilon > 0$ и всех достаточно больших чётных n существует однородный комбинаторный $(n, 3, \varepsilon)$-экспандер.*

(б) Докажите, что для всякого $\varepsilon > 0$ найдётся такое n_0, что при $n > n_0$ однородных комбинаторных $(n, 2, \varepsilon)$-экспандеров не существует.

Замечание: Не следует воспринимать определение 2.1 догматически — в некоторых случаях может оказаться удобно его немного подправить. В стандартном определении требуется, чтобы свойство расширения выполнялось для множеств, содержащих не более 50% от всех вершин графа. Выбор границы $n/2$ в определении достаточно произволен и не существенен для построения теории экспандеров. Для приложений иногда бывает удобнее потребовать, чтобы свойство расширения выполнялось лишь для достаточно малых множеств A (например, для множеств, содержащих не более 1% всех вершин графа) или, напротив, даже для достаточно больших множеств (например, для всех множеств, содержащих не более 99% всех вершин графа).

Упражнение 2.5 *Докажите, что для любого целого $d \geqslant 3$, любого $\delta > 0$ найдётся такое $\rho > 0$, что всех достаточно больших n, для большинства d-регулярных графов с n вершинами*

$$\min_{S \subset V,\ |S| \leqslant \rho n} \frac{|\Gamma(S)|}{|S|} \geqslant d - 1 - \delta$$

Объясните, почему оценку $(d - 1 - \delta)$ в правой части неравенства нельзя заменить на величну $d - \delta$.

3 Коэфициенты вершинного и рёберного расширение графа

В нашем основном определении 2.1 на странице 1 требуется, чтобы граф обладал свойством «вершинного расширения». Введём числовую характеристику графа — коэффициент вершинного расширения, который показывает, насколько хорошими «экспандерными» свойствами обладает данный граф.

Определение 3.1 *Будем называть* коэффициентом вершинного расширения *графа* $G = (V, E)$ *число*

$$h_V(G) = \min_{0<|S|\leqslant|V|/2} \frac{|\Gamma(S)\backslash S|}{|S|}$$

(*минимум берётся по множествам* S*, содержащим не более половины вершин графа*).

Заметим, что у каждого однородного (n, d, ε)-экспандера коэффициент вершинного расширения не меньше ε. С другой стороны, если к d-однородному графу без петель с n вершинами и с коэффициентом вершинного расширения ε добавить петли (в каждой из вершин), мы получим однородный $(n, d+1, \varepsilon)$-экспандер. Также можно рассмотреть свойство *рёберного расширения* графа:

Определение 3.2 *Будем называть* коэффициентом рёберного расширения *графа* $G = (V, E)$ *число*

$$h_E(G) = \min_{0<|S|\leqslant|V|/2} \frac{|E(S, \bar{S})|}{d|S|}$$

(*снова минимум берётся по всем множествам, содержащим не более половины вершин графа*).

Большое значение коэффициента рёберного расширения означает, что для любого множества вершин S достаточно большая доля рёбер, выходящих из вершин этого множества, приходят в вершины вне S (так сказать, «торчат наружу»). На лекции мы доказали следующие несложные утверждения:

Утверждение 3.1 *Докажите, что для любого* d*-однородного графа* G

$$h_V(G) \geqslant h_E(G).$$

Утверждение 3.2 *Пусть некоторый* d*-однородный граф с* n *вершинами имеет коэффициент рёберного расширения* ε*. Добавим к каждой вершине графа петлю. Докажите, что получившийся в результате граф будет* $(n, d+1, \varepsilon)$*-экспандером в смысле определения 2.1.*

Упражнение 3.1 *Докажите, что диаметр всякого однородного комбинаторного* $(n, 10, 1/100)$*-экспандера равен* $O(\log n)$ (*т.е., любые две вершины графа соединены путём длины не более* $O(\log n)$).

4 Применение экспандеров: уменьшение вероятности ошибки алгоритма без увеличения числа случайных битов

В этом параграфе мы рассмотрим несложный способ получения «псевдослучайных» битов с помощью экспандера. Мы покажем, как уменьшить

вероятности ошибки вероятностного алгоритма *без увеличения числа используемых случайных битов*. Мы ограничимся рассмотрением алгоритмов с односторонней ошибкой. Напомним стандартное определение класса задач, для которых существует полиномиальный вероятностный алгоритм с односторонней ошибкой.

Определение 4.1 *Язык L принадлежит сложностному классу* RP*, если существует полиномиальный алгоритм $\mathcal{A}$ такой что*

1. *если $x \in L$, то для случайно выбранного набора битов $r \in \{0,1\}^{\mathrm{poly}(n)}$ $\mathrm{Prob}_r \mathcal{A}(x,r) \geqslant 1/2$,*
2. *если $x \notin L$, то $\mathcal{A}(x,r) = 0$ для всех $r \in \{0,1\}^{\mathrm{poly}(n)}$.*

Покажем, что для любого $\delta > 0$ всякий полиномиальный вероятностный алгоритм $\mathcal{A}$ можно переделать в полиномиальный вероятностный алгоритм $\mathcal{A}'$ таким образом, что вероятность ошибки уменьшится с $1/2$ до δ, а число используемых случайных битов при этом не изменится.

Пусть исходный алгоритм использует $k = k(n)$ случайных битов для вычислений на входах длины n. Зафиксируем однородный $(2^k, d, \varepsilon)$-экспандер G для некоторого $\varepsilon > 0$. Индекс (номер) каждой вершины в этом графе записывается последовательностью из k нулей и единиц. Таким образом, мы можем отождествить вершины G и наборы из k битов.

Новый алгоритм действует следующим образом: выбирается случайная вершина v графа (для этого требуется k случайных битов); затем исходный алгоритм $\mathcal{A}$ последовательно запускается на всех d наборах случайных битов, соответствующих соседям вершины v в графе. Если все полученные ответы равны 0, новый алгоритм также возвращает 0; если же получен хотя бы один положительный ответ, то алгоритм возвращает 1.

Покажем, что у нового алгоритма вероятность ошибки не превосходит $\frac{1}{2(1+\varepsilon)}$. Обозначим $B = B(x)$ множество всех *плохих* (для данного x) вершин графа — множество таких вершин w из правой доли графа, которые соответствуют неверному ответу старого алгоритма на входе x. Аналогично, обозначим $C = C(x)$ множество таких вершин v графа, которые для которых новый алгоритм даёт неверный ответ на входе x. Очевидно, C состоит из вершин, все соседи которых лежат в B.

Предположим, что C содержит не менее $\frac{1}{2(1+\varepsilon)}$ вершин. Произвольным образом выберем из множества C некоторое подмножество, состоящее *ровно* из $\frac{1}{2(1+\varepsilon)}$ вершин и назовём его C'. Из определения экспандера следует, что

$$|\Gamma(C)| > (1+\varepsilon)|C'| = n/2.$$

Это противоречит тому, что все соседи C' лежат в B.

Таким образом, мы построили алгоритм, в котором ошибка снизилась с $\frac{1}{2}$ до $\frac{1}{2(1+\varepsilon)}$. Покажем, как понизить вероятность ошибки ещё больше. Зададимся некоторым числом t и построим алгоритм, вероятность ошибки которого меньше $\frac{1}{2(1+\varepsilon)^t}$. В новом алгоритме мы выбираем в графе случайную

вершину v (для по-прежнему этого требуется k случайных битов); затем запускаем исходный алгоритм $\mathcal{A}$ на всех наборах случайных битов, соответствующих вершинам w графа, в которые можно попасть из v за t шагов (таких вершин заведомо не более d^t). Если все полученные ответы равны 0, новый алгоритм также возвращает 0; в противном случае возвращается 1.

Оценим вероятность ошибки нового алгоритма. Снова обозначим $C = C(x)$ множество таких вершин v графа, которые для которых новый алгоритм даёт неверный ответ на входе x. Предположим, что C содержит не менее $\frac{1}{2(1+\varepsilon)^t}$ вершин. Выберем среди них подмножество, состоящее из ровно $\frac{1}{2(1+\varepsilon)^t}$ вершин и назовём его C'. Из определения экспандера следует, что

$$|\underbrace{\Gamma(\Gamma(\Gamma(\ldots\Gamma(}_{t \text{ итераций}} C')\ldots)))| > (1+\varepsilon)^t|C'| = n/2$$

Это противоречит тому, что все цепочки из t рёбер, начинающиеся вершиной из C', обязаны заканчиваться вершиной из B.

Выбирая параметр t достаточно большим, мы получим алгоритм с вероятностью ошибки менее $\frac{1}{2(1+\varepsilon)^t} < \delta$. При этом значение t зависит от желаемой вероятности ошибки δ, но не зависит от размера входа n.

Остаётся обсудить время работы построенного алгоритма. Мы используем старый алгоритм как «чёрный ящик» и вызываем его (на разных наборах случайных битов) d^t раз. Поскольку d и t – некоторые константы (не зависящие от входа алгоритма), и исходный алгоритм $\mathcal{A}$ работал за полиномиальное время, можно заключить, что все требуемые вызовы выполняются за полиномиальное время.

Однако кроме нескольких вызовов старого алгоритма нам требуется производить манипуляции с графом G — нужно уметь быстро находить всех соседей заданной вершины графа. Чтобы иметь возможность делать это за полиномиальное время, нам нужна *явная* конструкция экспандера. Более того, нам нужен экспандер *явный в сильном смысле*: размер графа экспоненциально растёт с увеличением k, и нам необходим алгоритм, который по заданному номеру вершины w за время $\mathrm{poly}(k)$ находит список номеров всех соседей w.

Следующее упражнение описывает один из классических вероятностных алгоритмов с односторонней ошибкой.

Упражнение 4.1 (Алгоритм Соловея–Штрассена для проверки простоты числа) *(а) Докажите, что значение* символа Якоби $(\frac{a}{b})$ *можно вычислить за время, полиномиально зависящее от длины входа* (*т.е., от длины двоичных записей* a *и* b).

(б) Вероятностный алгоритм Соловея–Штрассена тестирует простоту натурального числа n *следующим образом. Выбирается случайное число* a *из интервала* $2, \ldots, n-1$. *Если наибольший общий делитель* a *и* n *больше единицы, алгоритм собщает, что число* n *составное. В противном*

случае проверяется сравнение $a^{\frac{n-1}{2}} = (\frac{a}{n}) \mod n$. *Если оно не выполняется, то алгоритм сообщает, что* n *составное; иначе алгоритм сообщает, что* n *простое.*

Докажите, что алгоритм всегда возвращает правильный ответ для каждого простого n *и ошибается с вероятностью не более* $1/2$ *для каждого из составных* n.

Упражнение 4.2 (Проверка существования совершенного паросочетания в двудольном графе, Mulmuley–Vazirani–Vazirani) *Пусть задан двудольный граф* $G = (L, R, E)$, *где* $|L| = |R| = n$ (*множество вершин состоит из двух долей по* n *вершин в каждой*), *и* $E \subset L \times R$ (*каждое ребро соединет некоторую вершину из доли* L *с некоторой вершиной из доли* R). *Такой граф можно задать матрицей* M *размера* $n \times n$,

$$m_{ij} = \begin{cases} 1, & \text{если } i\text{-ая вершина } L \text{ осединена ребром с } j\text{-ой вершиной } R, \\ 0, & \text{иначе.} \end{cases}$$

Заменим каждую единицу в клетке с координатами (i, j) *в матрице* M *на переменную* x_{ij}. *Определитель полученной матрицы будет многочленом от переменных* x_{ij} (*его можно рассматривать как многочлен над* $\mathbb{R}$ *или над конечным полем* $\mathbb{F}_q$) .

(а) Докажите, что определитель построенной матрицы является тождественно нулевым, если и только если в графе нет совершенного паросочетания (*совершенным паросочетанием называется такое множества рёбер* P, *что каждая вершина графа инцидентна ровно одному ребру из* P).

(б) Докажите, что для любого набора значений x_{ij} (*из конечного поля* $\mathbb{F}_q$) *можно вычислить за время, полиномиально зависящее от* n *и от размера поля.* Указание: *чтобы вычислить определить матрицы размера* $n \times n$, *нет необходимости выписывать его в явном виде как сумму* $n!$ *мономов.*

(в) Рассмотрим конечное поле из q *элементов,* $q > 2n$. *Подставим в определитель описанной матрицы случайно и независимо выбранные элементы поля вместо каждой переменной* x_{ij}. *Докажите, что*

- *если в графе есть совершенное паросочетание, то значение определителя вычисленного на случайном наборе элементов поля оказывается равным нулю с вероятностью менее* $1/2$;
- *если в графе нет совершенного паросочетания, то значение определителя вычисленного на любом наборе элементов поля равно нулю.*

Указание: *если многочлен небольшой степени не равен нулю тождественно, то он обращается в ноль в сравнительно небольшом числе точек.*

(в) С помощью наблюдения из пункта (б) постройте полиномиальный вероятностный алгоритм, проверяющий существование совершенного паросочетния в двудольном графе размера $n \times n$: *если соврешенного паросочетания нет, алгоритм должен всегда выдавать ответ* 0, *а если хотя*

бы одно паросочетание есть, то алгоритм должен выдавать ответ 1 *с вероятностью* $>1/2$.