Конспект к лекции 3. (Санкт-Петербург, 9 апреля 2017 г.)

7 Быстрый алгоритм алгоритм декодирования экспандерного кода

Пусть $G = (L, R, E)$ — двудольный экспандер с параметрами $(n, m, d, k, \varepsilon)$. В этой лекции мы дополнительно предполагаем, что степени всех вершин в правой доле графа не слишком велики (ограничены величиной $O(dn/m)$). Мы считаем, что параметры графа подобраны так, что $m = \Theta(n)$, $k = \Theta(n)$ и $d = O(1)$.

Построим на экспандере линейный код, как было описано в на лекции 2 — каждой вершине в левой доле графа сопоставляется бит кодового слова, а каждой вершине правой доли графа сопоставляется контрольная сумма; набор битов считается кодовым словом, если все его контрольный суммы равны нулю. Ранее мы показали, что в таком коде расстояние между кодовыми словами не меньше k. Это значит, что если в кодовом слове искажены (инвертированы) менее $k/2$ битов, то мы можем исправить внесённые ошибки и восстановить исходное кодовое слово. Однако наивный алгоритм исправления ошибок (перебор всех возможных способов инвертировать $< k/2$ битов в слове) требует огромного перебора. Главное достоинство экспандерных кодов — это существование быстрых алгоритмы декодирования.

Далее мы рассмотрим один из таких алгоритмов. Мы опишем очень простой паралелльный алгоритм декодирования. Он использует $O(n)$ параллельно работающих процессоров и исправляет ошибки за время $O(\log n)$.

Вход алгоритма: набор битов $x_1, \ldots, x_n$, приписанных вершинам левой доли графа.

1. Для каждой вершины w из правой доли графа вычислить соответствующую контрольную сумму

$$c_w := \bigoplus_{v \in L \ : \ (v,w) \in E} x_v$$

2. Если все контрольные суммы равны 0, закончить работу, выдав текущий набор битов $x_1, \ldots, x_n$.

3. Для каждой вершины $v \in L$ инвертировать бит x_v, если более половины контрольных сумм, включающих x_v не равны 0, т.е.

   число вершин $w \in R$ таких, что $(v, w) \in E$ и $c_w = 1$, больше $d/2$

4. Вернуться к пункту 1.

Замечание: Вычисления в пунктах 1 и 3 данного алгоритма можно выполнять параллельно для всех вершин графа.

Теорема 7.1 *Если $\varepsilon < 1/8$ и исходный набор битов $\bar{x} = x_1, \ldots, x_n$ отличается от некоторого кодового слова $\bar{x}' = x'_1, \ldots, x'_n$ в не более, чем $k/2$ позициях, то через $O(\log n)$ итераций описанный алгоритм остановится и выдаст в качестве результата кодовое слово $\bar{x}'$.*

Доказательство теоремы будет использовать следующее определение.

Определение 7.1 *Пусть $G = (L, R, E)$ — двудольный граф, и $A \subset L$ некоторое множество вершин левой доли этого графа. Будем называть вершина правой доли графа $w \in R$* уединённым соседом *множества A, если существует ровно одна вершина $v \in A$, соединённая ребром с v. Других соседи A будем называть* неуединёнными.

Лемма 7.1 (об уединённых соседях) *Пусть граф $G = (L, R, E)$ является двудольным экспандером с параметрами $(n, m, d, k, \varepsilon)$ (без кратных рёбер), и $A \subset L$ — некоторое множество вершин левой доли графа, $|A| \leqslant k$. Тогда число уединённых соседей A (в правой доле графа R) не меньше $(1 - 2\varepsilon)d|A|$.*

Доказательство леммы: Обозначим U множество всех уединённых соседей A. Из A выходит $d|A|$ рёбер. При этом $|U|$ из них приходят в вершины, являющиеся уединёнными соседями A в правой доле (по одному ребру в каждого уединённого соседа). А все остальные рёбра приходят в *неуединённых* соседей (в каждого неуединённого соседа A приходит не меньше двух рёбер). Таким образом, мы получаем

$$|\Gamma(A)| \leqslant |U| + \frac{d|A| - |U|}{2} = \frac{1}{2}d|A| + \frac{1}{2}|U|.$$

С другой стороны, по определению экспандера мы имеем

$$|\Gamma(A)| > (1 - \varepsilon)d|A|.$$

Следовательно,

$$(1 - \varepsilon)d|A| < \frac{1}{2}d|A| + \frac{1}{2}|U|,$$

и $|U| > (1 - 2\varepsilon)d|A|$. Лемма доказана.

Доказательство теоремы 7.1: Пусть $\bar{x} = (x_1, \ldots, x_n)$ — текущий набор битов, приписанных вершинам левой части графа. Мы предполагаем, что $\bar{x}$ не более, чем в $k/2$ позициях отличается от некоторого кодового слова $\bar{y} = (y_1, \ldots, y_n)$. Покажем, что после очередной итерации алгоритма расстояние между новым набором битов $\bar{x}' = (x'_1, \ldots, x'_n)$ и кодовым словом $\bar{y}$ сократится не менее, чем в c раз для некоторой константы $c > 1$. (Из этого свойства алгоритма немедленно следует, что через $O(\log n)$ итерации

расстояние станет равно нулю, т.е., текущий набор битов превратиться в нужное нам кодовое слово $\bar{y}$.)

Обозначим $A \subset \{1, \ldots, n\}$ множество позиций, в которых текущее $\bar{x} = (x_1, \ldots, x_n)$ отличается от кодового слова $\bar{y} = (y_1, \ldots, y_n)$. После окончания одной итерации алгоритма $\bar{x} = (x_1, \ldots, x_n)$ превратится в некоторый набор битов $\bar{x}' = (x'_1, \ldots, x'_n)$; мы обозначим $A' \subset \{1, \ldots, n\}$ множество позиций, в которых новый набор битов будет отличаться от $\bar{y}$.

Изучим соотношение между A и A' более подробно. Для этого разделим A' на две части: положим

$$A' = B \cup C,$$

где $B \subset A$ и $C \subset \{1, \ldots, n\} \backslash A$. Другими словами, B состоит из позиций, в которых сохранятся исходные «неправильные» биты, а C состоит из позиций, в которых сначала стояли «правильные» биты (такие же, как в $\bar{y}$), но после применения одного шага алгоритма декодирования эти биты стали «неправильными» (отличающимися от битов в $\bar{y}$).

Оценим отдельно размеры B и C. Прежде всего отметим, что по Лемме 7.1 число уединённых соседей A не меньше $(1 - 2\varepsilon)d|A|$. Во всех этих соседях контрольные суммы заведомо равны 1. (Если вершина $w \in R$ не соединена ребром ни с одной вершиной из A, то её контрольная сумма равна 0, как у контрольной суммы кодового слова $\bar{y}$. Если же вершина w является неуединённым соседом A, то мы не можем точно сказать, будет ли её контрольная сумма равна 0 или 1 — это зависит от чётности числа вершин в A, соединённых ребром с w). Следовательно, не более $2\varepsilon d|A|$ рёбер ведут из A в некоторого неуединённого соседа.

Это наблюдение позволяет нам оценить размер B. В самом деле, «неправильный» (принадлежащий A) бит x_i остаётся не инвертированным (т.е., $i \in B$), только если хотя бы половина (хотя бы $d/2$ из d) его контрольных сумм равна нулю. А это значит, что хотя бы половина соседей данной вершины являются неуединёнными соседями A. Таким образом,

$$|B| \leqslant \frac{2\varepsilon d|A|}{d/2} = 4\varepsilon|A|.$$

Теперь оценим размер множества вершин C. Для этого нам потребуется рассмотреть множество соседей объединения $A \cup C$. Во-первых, среди соседей этого объединения встречаются все соседи A (коих не больше $d|A|$). Во-вторых, среди этих соседей встречаются также вершины $w \in R$, соединнные ребром с C, но не соединённые с A. Но вершин второго типа не может быть очень много. В самом деле, у каждой вершины C более половины соседей имеют единичную контрольную сумму; такие вершины обязаны быть соседями A (точнее, соседями *нечётного* числа вершин из A). Таким образом, каждая вершина из C может иметь не больше $d/2$ соседей, не покрытых множеством $\Gamma(A)$. Следовательно,

$$|\Gamma(A \cup C)| \leqslant d|A| + \frac{1}{2}d|C|$$

Теперь предположим, что объединение $A \cup C$ не очень велико (содержит не более k вершин). Тогда к $A \cup C$ можно применить свойство расширения экспандера. Получаем

$$d(1-\varepsilon)(|A|+|C|) < |\Gamma(A \cup C)| \leqslant d|A| + \frac{1}{2}dC,$$

откуда вытекает

$$|C| \leqslant \frac{\varepsilon}{\frac{1}{2}-\varepsilon}|A|.$$

Что делать, если в $A \cup C$ содержится больше k вершин? Просто выбросим из C «лишние» вершины — обозначим C' произвольное подмножество C, состоящее из ровно $k - |A|$ вершин. Затем применим к $A \cup C'$ приведенное выше рассуждение и получим

$$|C'| < \frac{\varepsilon}{\frac{1}{2}-\varepsilon}|A|.$$

Но тогда

$$|A \cup C'| < (1 + \frac{\varepsilon}{\frac{1}{2}-\varepsilon})|A| < \frac{|A|}{1-2\varepsilon} < k,$$

что противоречит выбору C'.

Теперь мы можем объединить полученные оценки для B и C. При $\varepsilon < 1/8$ получаем

$$|A'| = |B \cup C| < 4\varepsilon|A| + \frac{\varepsilon}{\frac{1}{2}-\varepsilon}|A| < \frac{6\varepsilon}{1-2\varepsilon}|A| < |A|.$$

Это значит, что число «неправильных» битов среди x_i на каждой итерации алгоритма уменьшается в константу раз. Понятно, что через $O(\log n)$ шагов ни одной ошибки не останется. Теорема доказана.

Упражнение 7.1 *Будем считать, что алгоритм декодирования выполняется набором из $O(n)$ параллельных алгоритмов, имеющих совместный доступ к $O(n)$ ячейкам памяти, и экспандер задан явно (для каждой вершины известен список её соседей).*

Покажите, что хранение информации (текущие значения битов x_i и текущие значения контрольных сумм) можно организовать таким образом, что каждая итерация описанного алоритма выполняется за время $O(1)$ (текущие значения битов x_i и текущие значения контрольных сумм)

Замечание: Каждая итерация описанного алгоритма состоит из $O(n)$ операций. Таким образом, если использовать этот алгоритм без распараллеливания, его выполнение потребует времени $O(n \log n)$. В следующем параграфе мы опишем модификацию данного алгоритма, работающую за время $O(n)$.

8 Коэффициенты вершинного и рёберного расширение графа

Введём числовую характеристику графа — коэффициент вершинного расширения, который показывает, насколько хорошими «экспандерными» свойствами обладает данный граф.

Определение 8.1 *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 8.2 *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:

Замечание 1: Для любого d-однородного графа G

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

Замечание 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 однородным комбинаторным $(n, d+1, \varepsilon)$-экспандером в смысле определения из лекции 1.

Упражнение 8.1 *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* $O(\log n)$).