О методах построения LDPC-кодов с заданными характеристиками.

Ананьев К.Ю.

В работе представлены алгоритмы построения проверочных матриц для LDPC - кодов на основе графа Таннера с обхватом 8. Также, в качестве параметров графа выступают разбиение степеней символьных вершин: отношение вершин степени 3 и степени 4 к общему числу символьных вершин, и скорость полученного кода. Код строится для произвольной скорости и произвольного разбиения за линейное, относительно колличество элементов матрицы, время.

Ключевые слова: LDPC - коды, граф Таннера, двудольные графы, распределение степеней вершин.

При передаче, информации разбивается на блоки определенной длины. Блоки преобразуются кодером или кодируются. Полученные блоки, которые называются кодовыми словами, передаются по каналу, возможно с ошибками. На обратной стороне декодер преобразовывает кодовые слова в исходную последовательность, исправляя, насколько возможно, ошибки.

Обобщение LDPC-кодов - коды на графах, были предложены Таннером. Проверочной матрице можно однозначно сопоставить двудольный граф следующим образом: пусть проверочная матрица имеет размер $(l \times m)$. Сопоставим ей граф $G = (V, W)$, причем $V = V_c \sqcup V_s$. $V_c = \{v_0^c, \dots v_{l-1}^c\}$ - множество проверочных вершин, $V_s = \{v_0^s, \dots v_{m-1}^s\}$ - множество символьных вершин. Тогда W множество ребер типа $W \subseteq V_c \times V_s$. Причем ребро $(v_i^c, v_j^s) \in W$, если на пересечении соответствующих строк и столбцов в проверочной матрице стоит 1. Число ребер, связывающих данный символьный/проверочный узел с проверочными/символьными узлами называется степенью этого узла.

Одной из характеристик кода является «скорость». Величина скорости показывает степень «избыточности» кода: чем больше скорость, тем эффективнее алгоритм кодирования.

Также важной характеристикой матрицы LDPC кода является отсутствие «циклов» определенной длины. Под «циклами» матрицы понимаются циклы в соответствующем Графе Таннера. Практика показала, что наличие циклов малой длины существенно усложняет процесс декодирования и увеличивает вероятность ошибки.

В данной работе приведен пример построения семейства LDPC-кодов для произвольной скорости без циклов длины 4 и 6 для некоторого распределения степеней символьных узлов за линейное относительно размеров матрицы время.

1. Постановка задачи и формулировка основных результатов

Пусть $\mathbf{G}$ некоторое семейство проверочных матриц (графов), которые будут получаться при реализации алгоритма. Тогда, опишем параметры графы, на которые мы будем обращать внимание при построении $\mathbf{G}$.

Скорость кода определяется соотношением $v = \frac{k}{m}$, где k - длина кодируемых данных, m - длина кодовых данных. Тогда проверочная матрица H имеет размер $((n-k) \times m) = (l \times m)$.

Для любой скорости v должна существовать проверочная матрица $g \in \mathbf{G}$ со скоростью $v_1 \geq v$.

Будем строить семейство матриц, с обхватом 8. Для любой матрицы $g \in \mathbf{G}$ обхват соответствующего графа Таннера больше 6.

Теперь опишем требования для степеней вершин. Среди проверочных вершин не должно быть «висячих» вершин: степень всех проверочных вершин больше единицы. Степени всех символьных вершин должны быть больше 3-х. Семейство $\mathbf{G}$ будет допускать построение графов, у которых степени символьных вершин 3 или 4. Причем для любой доли символьных вершин со степенью четыре - p_4 будет существовать граф $g \in \mathbf{G}$ такой, что доля вершин степени четыре в нем p_4^g в графе g приблизительно равна p_4.

В ходе работы будут построены два семейства графов $\mathbf{G}_1$ и $\mathbf{G}_2$. Разделение $\mathbf{G}_1$ и $\mathbf{G}_2$ обусловлена тем, что длина кода в $\mathbf{G}_1$ существенно ниже.

Первое семейство $\mathbf{G}_1$. Будет доказано, что обхват всех матриц из $\mathbf{G}_1$ будет больше 6-и, а все символьные вершины имеют степень 3 и будет справедлива следующая теорема:

Теорема 1. *Для любой скорости кода v существует такая $g \in \boldsymbol{G}_1$, у которой все символьные вершины имеют степень 3, и скорость полученного кода $v_1 > v$.*

Опишем $\mathbf{G}_2$. Помимо свойств $\mathbf{G}_1$, $\mathbf{G}_2$ будет допускать построение матриц, с заданным распределением символьных вершин степеней 3 и 4. Поэтому для $\mathbf{G}_2$ будет доказана следующая теорема.

Теорема 2. *Для любой скорости кода v и доли символьных вершин степени p_4 существуют матрица $g \in \boldsymbol{G}_2$, у которой все символьные вершины имеют степень 3 или 4, причем доля вершин степени 4 приблизительно равна p_4, а скорость полученного кода $v_1 > v$.*

2. Построение графа для произвольной скорости кода со степенью символьных узлов равной 2-м

Строение графа будет зависеть он нескольких параметров. Пусть $n > 1$ - произвольное натуральное число.

Выберем одну проверочную вершину и назовем ее «корневой». Из нее выходят ребра в n символьных узлов, а каждую символьную с новой проверочной. Пронумеруем последние проверочные вершины числами $1 \dots n$. Из каждой проверочной вершины крайнего слоя проведем i ребер в новые символьные вершины, где i номер вершины. Таким образом на последнем слое у нас $\frac{n^2+n}{2}$ вершин. Каждую из символьных вершин, выходивших из вершины под номером n соединяем с новыми n различными вершинами. Выбираем из них $n-1$ и соединяем со всеми вершинами выходящими из вершины с номером $n-1$. Затем выбираем из оставшихся $n-2$ вершины и т.д. Затем соединяем n вершин из последнего рассматриваемого слоя с новыми n символьными, а те в свою очередь сводим в одну проверочную. На рисунке 1 приведен граф при $n = 3$

Лемма 1. *Обхват получившегося графа равен 8.*

Пронумерует слои графа числами от 1 до 7. Будем рассматривать только простые циклы в графе.

Если цикл проходит через вершину первого(седьмого) слоя, то в силу построения он проходят через вершину пятого(третьего) слоя, и в цикле будет минимум 8 вершин. В этом случае цикл может проходит ровно через одну вершину в пятом(третьем) слое.

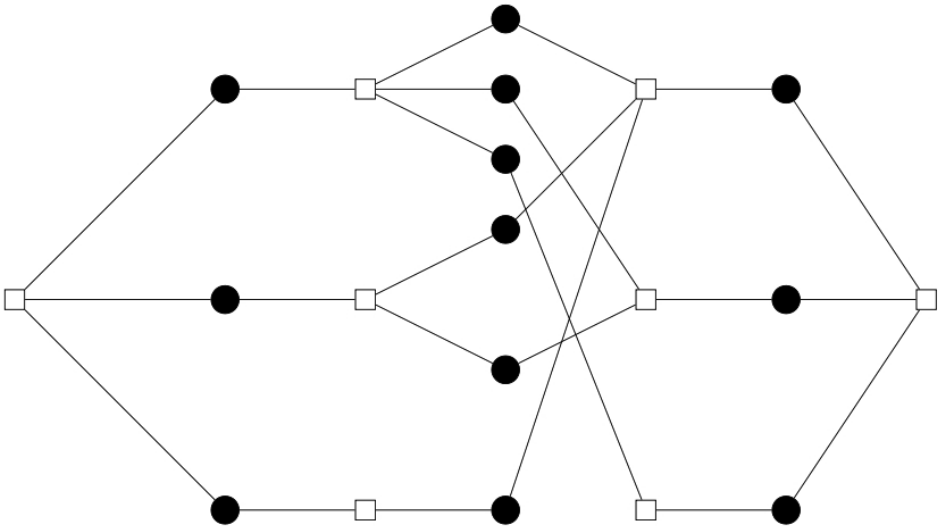

Рис.1

Если цикл проходит через вершину второго (шестого) слоя, то он проходит и через первый (седьмой) слой. Следовательно, переходим к предыдущему случаю.

Если цикл не проходит через вершины первых двух слоев, то в нем присутствует, как минимум, две вершины из третьего слоя v_1 и v_2. Т.к. вершин второго слоя в цикле нет, то, в силу построения, в цикле будут, минимум, четыре вершины из четвертого слоя: две из которых соединены с v_1 и две с v_2. В силу построения, вершины соединенные с v_i в четвертом слое соединяются с разными вершинами в пятом слое. Следовательно, в цикле есть минимум две вершины из пятого слоя. Таким образом, длина цикла не меньше 8.

Следовательно, обхват графа равен 8. Лемма доказана.

Скорость кода, соответственно, у таких графов равна:

$$v = 1 - \frac{2(1+n)}{2n + \frac{n^2+n}{2}} = 1 - \frac{4(1+n)}{n^2+5n}$$

и можно доказать следующую лемму:

Лемма 2. *Для любой скорости v_1 существует код построенный на графах данного типа со скоростью $v > v_1$.*

Несмотря на хорошие оценки скорости в этом графе есть существенный недостаток: наличие символьных вершин степени два может вы-

звать ошибки при декодировании. Для этого необходимо модифицировать исходные графы, чтобы степени символьных вершин были не меньше трех.

3. Первая модификация: построение семейства графов G_1 со степенями символьных вершин равными 3-м

Возьмем граф состоящий из первых пяти слоев графа, полученного в первой части работы, и назовем его «базисным». Сделаем $2 \cdot q$ копий базисного графа, пронумеруем их и начнем соединять следующим образом: вершины второго слоя первого графа соединяем с вершинами пятого слоя следующего и т.д. Это можно сделать, т.к. во втором и пятом слоях ровно по n вершин. Вершины последнего q-ого графа второго слоя с вершинами пятого слоя первого графа. Вершины второго слоя - символьные, а вершины пятого слоя проверочные, поэтому получившийся граф также двудольный.

Теперь добавим два дополнительного набора по $\frac{n^2+n}{2}$ проверочных вершин. Вершины из первого набора соединим с вершинами из 4 слоя базисных графов с четными номерами, а вершины второго набора соответственно с нечетными. Это можно сделать, т.к. в четвертом слое базисного графа также ровно $\frac{n^2+n}{2}$ вершин. Для того, чтобы не было висячих вершин в графе сделаем $q \geq 2$. Пример графа при $n = 3$ и $k = 2$ приведен на рисунке 2.

Лемма 3. *Обхват полученного графа равен 8.*

Случаи когда цикл расположен в одном базисном графе рассмотрены ранее.

Пусть цикл не содержит вершин из дополнительного множества и не лежит полностью в одном базисном графе. Тогда он лежит, минимум в двух базисных графах. Следовательно, ему принадлежат вершины из четвертых слоев, минимум, двух базисных графов. Граф двудольный и, следовательно, ему принадлежат минимум 4 символьные вершины. Таким образом длины цикла не меньше 8.

Пусть цикл содержит одну вершину v_1 из дополнительного множества. Вершина v_1 смежна с вершинами четвертого слоя не соседних базисных графов. Но путь между четвертыми слоями не соседних базисных

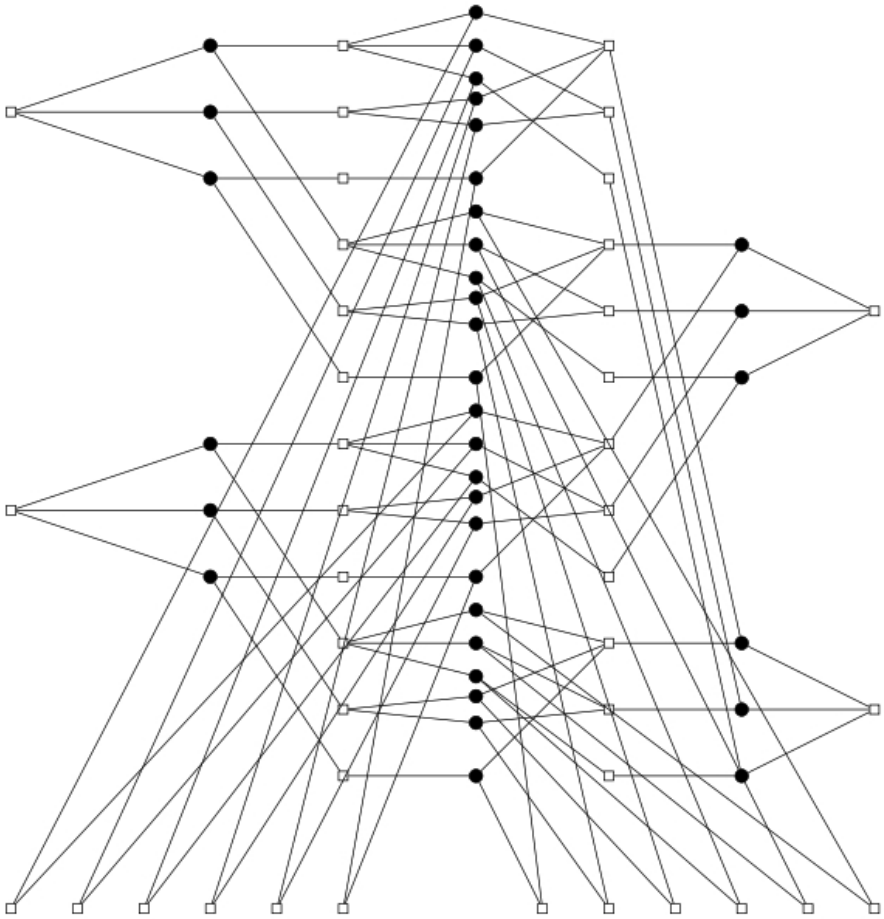

Рис.2

графов без использования допонительных вершин имеет длину больше 8. Следовательно, цикл имеет длину больше 10.

Пусть циклу принадлежат, минимум, 2 вершины из дополнительно множества. Тогда в силу построения циклу принадлежит 4 вершины из четвертых базисных слоев. Следовательно, длина цикла будет больше 8.

Таким образом, в графе нет циклов длины 4 и 6 и лемма доказана.

Оценим скорость кода для полученного графа.

$$v = 1 - \frac{2q(1+2n)+n^2+n}{2q(\frac{n^2+n}{2}+n)} = 1 - \frac{2q(1+2n)+n^2+n}{q(n^2+3n)}$$

Можно заметить, что при увеличении q доля дополнительных вершин, среди всех проверочных, стремится к 0:

$$\lim_{q->\infty} v = \lim_{q->\infty} (1 - \frac{2q(1+2n)+n^2+n}{q(n^2+3n)}) = 1 - \frac{2+4n}{n^2+3n}$$

Таким образом доказана следующая теорема:

Теорема 3. *Для любой скорости кода v существуют такие n, q и двудольный граф задающий проверочную матрицу LDPC кода, у которой все символьные вершины имеют степень 3, и скорость полученного кода $v_1 > v$.*

4. Вторая модификация: построение семейства графов G_2 со степенями символьных вершин равными 4-м

Будем использовать конструкция подобную той, что была описана в предыдущем разделе. Только теперь вместо двух дополнительных наборов проверочных вершин возьмем p наборов по $\frac{n^2+n}{2}$, где число p зависит только от q - числа копий базисного графа.

Суть построения заключается в том, чтобы соединить вершины из дополнительного набора с вершинами четвертых слоев базисных графов так, чтобы каждая вершина из четвертого слоя была соединена с двумя дополнительными вершинами, вместо одной, как это было при прошлом построении. При этом не должно остаться висячих вершин и обхват графа должен по-прежнему быть равен 8-ми.

Введем следующее обозначение: Пронумеруем базисные графы числами $\{1 \dots q\}$ и дополнительные наборы числами $\{1 \dots p\}$. Тогда запись $\{(a_{11} \dots a_{1w_1})(a_{21} \dots a_{2w_2}) \dots \ \dots (a_{p1} \dots a_{pw_p})\}$ обозначает, что вершины первого дополнительного набора соединены с вершинами четвертого слоя базисных графов под номерами $(a_{1,1} \dots a_{1,w_1})$ и так далее до p-ого дополнительного слоя.

Таким образом, в предыдущем построении было соединение типа: $\{(0\ 2\ \dots 2q-2)(1\ 3\ \dots 2q-1)\}$.

Перейдем непосредственно к построению. Для этого рассмотрим несколько случаев, которые могут привести к появлению циклов длины меньше 8-ми.

А) Для любых различных базисных графов с номерами $i, j, k \in \overline{1 \dots q}$ не существует трех дополнительных наборов типа: $(\dots i \dots j \dots)$, $(\dots i \dots k \dots)$

и $(\dots j \dots k \dots)$. Иначе опять на выходе получится цикл длины 6(Рисунок 3).

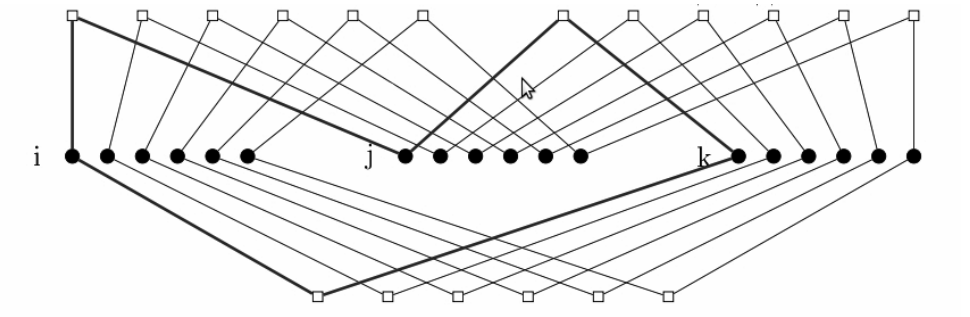

Рис.3

B) Из одного дополнительного набора, не должны выходить ребра в соседние базисные графы. Т.е. не должно быть соединения типа: $(\dots i\ i+1\dots)$. Иначе на выходе получаются циклы длины 6 (Рисунок 4).

C) Для любых различных базисных графов с номерами $i, j \in \overline{1\dots q}$ существует не более одного дополнительного набора, который соединен с ними. Иначе на выходе получатся циклы длины 4 (Рисунок 4).

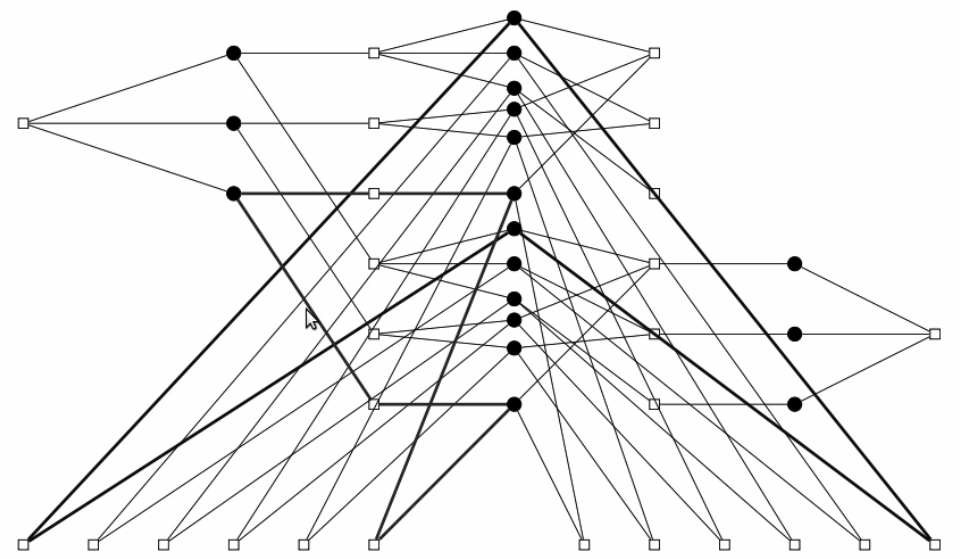

Рис.4

Таким образом, можно сформулировать следующую лемму:

Лемма 4. *Для некоторых q, n, существует p такое, что граф построенный с помощью данного алгоритма с соединениями типа $\{(a_{11} \dots a_{1w_1})(a_{21} \dots a_{2w_2}) \dots \dots (a_{p1} \dots a_{pw_p})\}$, и эти соединения удовлетворяют условиям **A, B, C**, имеет обхват 8.*

Будем рассматривать только простые циклы.

Если цикл не проходит через вершины дополнительного набора, то его длина больше 6. Это доказывалось раньше.

Если цикл проходит через одну вершины дополнительного набора, то единственный случай, когда может появиться цикл длины 6, это когда в одном соединении лежат два соседних базисных графов. Но это противоречит пункту **B**. Если в одном соединении нет соседних базисных графов, то длина подобных циклов будет больше 6-ти, т.к. расстояние между четвертыми слоями не соседних базисных графов больше 8-ми.

Если цикл содержит только две вершины из одного дополнительного набора, то длина цикла будет больше 8-ми, т.к. эти 2 вершины соединены с 4-мя разными вершинами из четвертых слоев базисных графов. Следовательно, в силу двудольности графа длина цикла будет не меньше 8-ми.

Если цикл содержит две вершины $v_1,\ v_2$ из разных дополнительных наборов. Если их соединения не имеют общих базисных графов, то цикл, очевидно, будет больше 8. Если они имеют единственное общее соединение с графом под номером i. Тогда $v_1,\ v_2$ могут иметь общую смежную вершину из графа i (иначе можно свести к предыдущему пункту). Также $v_1,\ v_2$ соединены с другими различными графами $j,\ k$ соответственно. Но путь соединяющий четвертые слои базисных графов без использования вершин дополнительного множества содержит больше четырех вершин. Следовательно, длина всех таких циклов будет больше 6.

Если цикл содержит 3 вершины из разных дополнительных наборов. Тогда нет соединений типа $(\dots i \dots j \dots)$, $(\dots i \dots k \dots)$ и $(\dots j \dots k \dots)$. Следовательно, цикл содержит вершины, по крайней мере, четырех разных базисных графов и длина цикла будет не меньше 8.

Если цикл содержит больше трех вершины из дополнительных наборов, то его длина будет больше 8-ми, в силу двудольности графа.

Следовательно, в графе нет циклов длины 4 и 6.

Пусть q- четное. Возьмем $p = q$ и разбиение типа $\{(0,2)(1,3)...(2q-2,0)(2q-1,1)\}$. Соединение удовлетворяет **A, B, C**. Следовательно, существуют графы удовлетворяющие заданным условиям и имеющие обхват 8. Лемма доказана.

5. Подсчет скорости и оптимизация числа p

Скорость кода на заданном графе равна:

$$v = 1 - \frac{q(1+2n) + p(\frac{n^2+n}{2})}{q(\frac{n^2+n}{2} + n)}$$

Выбор параметра p играет важную роль для определения скорости, поэтому необходимо сделать его как можно меньше.

Предлагается следующий подход: пусть задано некоторое $q > 8$, тогда возьмем начальное соединение такого типа $\{(0)(1\ 3)(2\ 4\ 6)...\}$ и покрываем все множество $\overline{0 \dots q-1}$. Чередуем блоки с четными и нечетными числами увеличивая длину блока до тех пор, пока не дойдем до q. Последние два блока могут быть неполными. Посчитаем количество блоков, которые необходимо для покрытия множества. Данный набор удовлетворяет условиям **A, B, C**.

Рассмотрим последовательность a_i $i \geq 0$ самых больших чисел в каждом блоке: $0, 3, 6, 11, 16, \dots$. Тогда последовательность $a_{i+1} - a_i$ имеет вид: $3, 3, 5, 5, 7, \dots$. Найдем общий вид этих чисел в четных и нечетных подпоследовательностях. В четном блоке имеет вид $a_{2N} = 2(\frac{3+(2N+1)}{2}N) = 2N^2 + 4N$. В нечетном блоке $a_{2N+1} = a_{2N} + (3 + 2N) = 2N^2 + 6N + 3$. Последовательность a_i - возрастающая. Следовательно, $a_{i-1} < q \leq a_i$ и нам необходимо $i + 1$ набор. Отметим, что если последний неполный блок имеет 1 элемент, то предпоследний блок тоже будет неполным. Тогда вместо q возьмем $q + 2$ при этом колличество блоком неувеличится, а в последнем блоке будет 2 элемента.

На данном этапе мы имеем только одно соединение базисных графов с дополнительными наборами. Построим второе соединение. Мы имеем $i+1$ блоков $\{(0)(1\ 3)(2\ 4\ 6)\dots\}$. Добавим еще i блоков следующим образом: первый новый блок имеет вид: $(0\ 3\ 6\dots)$ возьмем первый элемент из первого блока, второй элемент из второго блока и так далее, пока это возможно. Получаем, что в первом блоке нет элемента из последнего блока предыдущего соединения. Второй новый блок $(1\ 4\dots)$ т.е. первый элемент второго блока, второй элемент третьего блока и так далее. Таким образом заполняем все $i+1$ блоков. В последнем будет один элемент, который был первым в последнем блоке в прошлом соединении. Получаем второе начально соединение типа $\{(0\ 3\ 6\dots)(1\ 4\dots)(2\dots)\dots\}$. Второе соединение также удовлетворяет **A, B, C**.

Теперь объединяем первое и второе начальное соединения, при этом, отождествляя первый единичный блок из первого и последний единич-

ный блок из второго. Таким образом получим соединение с $p = 2i + 1$ блоком типа $\{(1\ 3)(2\ 4\ 6)\dots(0\ 3\ 6\dots)(1\ 4\dots)(2\dots)\dots(0\ j)\}$, где j единственный элемент последнего блока во втором соединении.

Такое построение обусловлено тем, чтобы полученное соединение удовлетворяло условиям **A, B, C**. Очевидно, что в каждом блоке нет соседних элементов, поэтому **B** выполнено, если $q \neq 2N^2 + 4N$ и $q \neq 2N^2+6N+3$. Это необходимо, что первый и последний элемент не попали в один блок. Так как при построении второго начального соединения мы объединяли в блоки элементы из разных блоков в первом соединении, то в совокупности получим соединение удовлетворяющее условию **C**. Рассмотрим соединение полученное из последнего удалением блока $(0\ j)$. Предположим, существует три блока, которые не удовлетворяют условию **A**. Т.е. существует элементы с номерами $i, j, k \in \overline{0\dots q-1}$ и блоки: $(\dots i \dots j \dots)$, $(\dots i \dots k \dots)$ и $(\dots j \dots k \dots)$. Так как первый и второй блоки имеют общий элемент i, они изначально принадлежали разными начальным соединениям. Пусть первый блок принадлежал первому соединению, второй блок второму. Третий блок имеет элемент j, следовательно, он не принадлежит первому соединению. Но третий блок также содержит элемент k, следовательно, второму соединению он также не принадлежит. Следовательно, получаем противоречие и полученное соединение удовлетворяет **A**. Таким образом, если полное соединение не удовлетворяет **A**, то «неправильными» блоками являются $(0\ j)$, последний блок первого начального соединения и первый блок второго начально соединения. Но так как первый блок из второго начально соединения не содержит элементов из последнего блока первого начально разбиения, получаем противоречие. Следовательно, полное разбиение удовлетворяет **A** и полученный граф имеет обхват 8.

Перейдем к подсчету скорости. Пусть $i = 2N$ четное и $a_{2N-1} < q < a_{2N}$. Тогда:

$$2(N-1)^2 + 6(N-1) + 3 < q \leq 2N^2 + 4N$$

$$2N^2 + 2N - 1 < q < 2N^2 + 4N$$

Так как $N, q \in \mathbb{N}$, получаем, что $\lfloor\sqrt{3+2q} - 1\rfloor \geq N \geq \lceil\sqrt{4+2q} - 2\rceil$. Соответственно $p = 2i+1 = 4N+1 \leq 4\lfloor\sqrt{3+2q}-1\rfloor+1 = 4\lfloor\sqrt{3+2q}\rfloor-3$. С другой стороны $p > 4\lceil\sqrt{4+2q}\rceil - 5$ В результате мы имеем:

$$v = 1 - \frac{q(1+2n) + p(\frac{n^2+n}{2})}{q(\frac{n^2+n}{2} + n)} \geq 1 - \frac{q(1+2n) + (4\lfloor\sqrt{3+2q}\rfloor - 3)(\frac{n^2+n}{2})}{q(\frac{n^2+n}{2} + n)}$$

Видно, что при увеличении q скорость стремится к отношению скорости на одном базисном графе:

$$\lim_{q->\infty} v = 1 - \frac{2+4n}{n^2+3n}$$

Обозначим через p_3 и p_4 доли символьных вершин, степени 3 и 4 соответственно, в получившемся графе. Тогда, в такой конструкции максимальная доля символьных вершин имеющих степень 4 равна

$$p_4^m = \frac{\frac{n^2+n}{2}}{\frac{n^2+n}{2}+n} = \frac{n^2+n}{n^2+3n} = 1 - \frac{2n}{n^2+3n}$$

Для построения графа с произвольной долей p_4 вершин степени 4, необходимо последовательно заполнять второе начальное соединение пока доля вершин степени не достигнет заданного значения, с той лишь оговоркой, что блок $(1, j)$ необходим, чтобы в конструкции не было висячих вершин. Поэтому будет 1 блок вершин степени четыре необходим. $v_4^0 = \frac{\frac{n^2+n}{2}}{q(\frac{n^2+n}{2}+n)} = \frac{n^2+n}{qn^2+3qn} = \frac{1}{q} - \frac{2n}{qn^2+3qn}$ Следовательно:

$$\frac{1}{q} - \frac{2n}{qn^2+3qn} = v_4^0 \le v_4 \le v_4^m = 1 - \frac{2n}{n^2+3n}$$

Можно заметить, что с увеличением q и n растет интервал выбора скорости и отрезок выбора доля вершин степени четыре. Таким образом можно сформулировать следующую теорему:

Теорема 4. *Для любой скорости кода v и доли символьных вершин степени p_4 существуют такие n, q и двудольный граф задающий проверочную матрицу LDPC кода, у которой все символьные вершины имеют степень 3 или 4, причем доля вершин степени 4 приблизительно равна p_4, а скорость полученного кода $v_1 > v$.*

Список литературы

[1] Shannon C.E. A Mathematical Theory of Communication // Bell System Technical Journal. — 1948. — Т. 27. — С. 379-423, 623–656.

[2] Gallager, R. G. Low Density Parity Check Codes. — Cambridge: M.I.T. Press, 1963. — P. 90.

[3] http://www.inference.phy.cam.ac.uk/mackay/otherECC.html

[4] Gallager R.G. Low-Density Parity-Check Codes. Cambridge, MA, MIT Press, 1963.

[5] Морелос-Сарагоса Р. Искусство помехоустойчивого кодирования. Методы, алгоритмы, применение. — М.: Техносфера, 2006. — 320с.

About methods of constructing LDPC-codes with preset characteristics

Ananiev K.Y.

The work presents algoritms for building test matrices for LDPC, which are codes based on a Tanner graph with a girth of 8. Other parameters of the graph, apart from the girth, include the division of degrees of character vertices: the ratio of portion of vertices with degrees 3 and 4 to their total number, as well as the speed of the code generated. The code is built for random speed and random division in linear time depending on the number of elements of the matrix.

Keywords: LDPC-codes, bipartite graph, division of degrees of vertices.