Вычисление спектра циклов графа

10 апреля 2017 г.

1 Ведение

Интуиция подсказывает, что большой обхват и малое число коротких циклов положительно сказываются на эффективности итеративного декодирования.

Существуют и другие критерии для поиска хороших МППЧ кодов, например, ACE. В любом случае, все сводится к анализу струтуры циклов графа. Поиск кодов, обладающих хорошей структурой – вычислительно сложная задача. Наша задача - упростить вычисления.

Известные подходы ....

Новый подход имеет простое описание и низкую вычислительную сложность

2 Алгоритм подсчета спектров

Эта задача порождена проблемой анализа и оптимизации МППЧ кодов. Пусть B – (двоичная) базовая матрица кода. Для нее определен двудольный граф Таннера $T = \{V, E\}$ с множеством вершин $V = V_s \cup V_c$, где V_s и V_c – множества символьных и проверочных вершин, соответственно. Единицам матрицы B соответствуют ребра графа T.

Пример 2.1. *Рассмотрим базовую матрицу B*

$$B(D) = \begin{pmatrix} 1 & 1 & 0 & 1 \\ 1 & 1 & 1 & 0 \\ 1 & 0 & 1 & 1 \end{pmatrix}, \tag{1}$$

для задающей квазициклический МППЧ код полиномиальной проверочной матрицы

$$H(D) = \begin{pmatrix} D^{w_{11}} & D^{w_{12}} & 0 & D^{w_{14}} \\ D^{w_{21}} & D^{w_{22}} & D^{w_{23}} & 0 \\ D^{w_{31}} & 0 & D^{w_{33}} & D^{w_{34}} \end{pmatrix} \tag{2}$$

Для удобства переназначим веса переходов

$$H(D) = \begin{pmatrix} D^{w_1} & D^{w_4} & 0 & D^{w_8} \\ D^{w_2} & D^{w_5} & D^{w_6} & 0 \\ D^{w_3} & 0 & D^{w_7} & D^{w_9} \end{pmatrix} \tag{3}$$

Соответствующая матрица инцидентности графа Таннера

$$T(D) = \begin{pmatrix} 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 \\ D^{w_1} & 0 & 0 & D^{w_4} & 0 & 0 & 0 & D^{w_8} & 0 \\ 0 & D^{w_2} & 0 & 0 & D^{w_5} & D^{w_6} & 0 & 0 & 0 \\ 0 & 0 & D^{w_3} & 0 & 0 & 0 & D^{w_7} & 0 & D^{w_9} \end{pmatrix}$$

Сам граф показан на рис. 2. Кружками и квадратами показаны символьные и проверочные узлы.

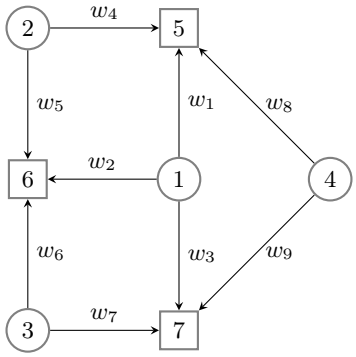

Рис. 1: Граф Таннера для кода из примера 2.1

Задача состоит в подсчете числа циклов заданной длины в расширенном графе с заданной степенью расширения M. Длина цикла равна числу переходов в пути, начинающемся и заканчивающемся в одном и том же узле и таком, что сумма весов переходов по модулю M равна нулю. Веса суммируются с учетом знаков, зависящих от направления перехода (см. рис. 2).

Такой тип циклов часто нызывают замкнутым обходом, однако при подсчете рассматриваемых объектов необходимо учесть следующие дополнительные ограничения.

- Запрещено двигаться обратно по последнему пройденному ребру. Например, путь 6→1→6 веса 0 запрещен.
- Циклические сдвиги путей и инверсии путей должны учитываться как один цикл. Например, путь $p = 6 \to 1 \to 7 \to 3 \to 6$ образует цикл при

условии $w_2 - w_6 + w_7 - w_3 = 0$. При этом пути $6 \to 3 \to 7 \to 1 \to 6$ и $1 \to 7 \to 3 \to 6 \to 1$ тоже циклы, но они уже не вносят вклад в число циклов длины 4, если цикл p учтен.

Хотелось бы применить стандартные методы, используемые при анализе систем на основе конечных автоматов. Данный граф не является конечным автоматом, поскольку перемещение из состояния в состояние зависит от предыдущего состояния. Например, на рис. 2 после состояния 4 возможно только 5, если предыдущим было 7 и, наоборот, только 7, если предыдущим было 5.

Чтобы свести задачу к анализу конечных автоматов, введем новое множество состояний $U = \{e, \xi\}$, где e задает ребро исходного графа, а ξ – направление перехода. Сокращенно будем записывать пары в виде $+e$ и $-e$.

Из графа на рис. 2 получится граф с 18 состояниями $\{\pm 1, \pm 2, ..., \pm 9\}$. Заметим, однако, что после отрицательного ребра следуют только положительные и после положительного отрицательные. Это позволит записать матрицу переходов компактно в виде двух матриц, матрицы положительных и матрицы отрицательных переходов. Например, следующими состояниями (ребрами графа Таннера) после положительного перехода $+1$ возможны отрицательные $-4, -8$. После отрицательного перехода -1 возможны положительные $2, 3$.

В нашем примере две матрицы переходов имеют вид

$$A_- = \begin{pmatrix} 0 & 0 & 0 & D^{-w_4} & 0 & 0 & 0 & D^{-w_8} & 0 \\ 0 & 0 & 0 & 0 & D^{-w_5} & D^{-w_6} & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & D^{-w_7} & 0 & D^{-w_9} \\ D^{-w_1} & 0 & 0 & 0 & 0 & 0 & 0 & D^{-w_8} & 0 \\ 0 & D^{-w_2} & 0 & 0 & 0 & D^{-w_6} & 0 & 0 & 0 \\ 0 & D^{-w_2} & 0 & 0 & D^{-w_5} & 0 & 0 & 0 & 0 \\ 0 & 0 & D^{-w_3} & 0 & 0 & 0 & 0 & 0 & D^{-w_9} \\ D^{-w_1} & 0 & 0 & D^{-w_4} & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & D^{-w_3} & 0 & 0 & 0 & D^{-w_7} & 0 & 0 \end{pmatrix}$$

$$A_- = \begin{pmatrix} 0 & D^{w_2} & D^{w_3} & 0 & 0 & 0 & 0 & 0 & 0 \\ D^{w_1} & 0 & D^{w_3} & 0 & 0 & 0 & 0 & 0 & 0 \\ D^{w_1} & D^{w_2} & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & D^{w_5} & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & D^{w_4} & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & D^{w_7} & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & D^{w_6} & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & D^{w_9} \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & D^{w_8} & 0 \end{pmatrix}$$

Заметим, что все циклы имеют четную длину и состоят из пар переходов (по стрелке, против стрелки). Можно построить матрицу переходов за два

шага, как произведение матриц A_+, A_-

$$A = A_+A_- = \begin{pmatrix} 0 & 0 & 0 & 0 & \omega_{45} & 0 & 0 & 0 & \omega_{89} \\ 0 & 0 & 0 & \omega_{54} & 0 & 0 & \omega_{67} & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & \omega_{76} & 0 & \omega_{98} & 0 \\ 0 & \omega_{12} & \omega_{13} & 0 & 0 & 0 & 0 & 0 & \omega_{89} \\ \omega_{21} & 0 & \omega_{23} & 0 & 0 & 0 & \omega_{67} & 0 & 0 \\ \omega_{21} & 0 & \omega_{23} & \omega_{54} & 0 & 0 & 0 & 0 & 0 \\ \omega_{31} & \omega_{32} & 0 & 0 & 0 & 0 & 0 & \omega_{98} & 0 \\ 0 & \omega_{12} & \omega_{13} & 0 & \omega_{45} & 0 & 0 & 0 & 0 \\ \omega_{31} & \omega_{32} & 0 & 0 & 0 & \omega_{76} & 0 & 0 & 0 \end{pmatrix}$$

где $\omega_{ij} = D^{-w_i+w_j}$.

В общем случае матрица переходов за 2 шага будет иметь размер, равный числу единиц в базовой матрице B.

Для подсчета производящей функции числа путей длины $2L$, начинающихся с ребра 1 в положительном направлении нужно начальный вектор $a_0 = (1, 0, ..., 0)$ умножить на A^L. Для нашего примера при

$$\boldsymbol{a}_2 = \boldsymbol{a}_0 A = \begin{pmatrix} 0 & 0 & 0 & 0 & \omega_{45} & 0 & 0 & 0 & \omega_{89} \end{pmatrix} \tag{4}$$

$$\boldsymbol{a}_4 = \boldsymbol{a}_2 A = \begin{pmatrix} \omega_{4521,8931} & \omega_{8932} & \omega_{4523} & 0 & 0 & \omega_{8976} & \omega_{4567} & 0 & 0 \end{pmatrix} \tag{5}$$

где первая компонента является сокращенной записью полинома

$$\omega_{4521,8931} = D^{-w_4+w_5-w_2+w_1} + D^{-w_8+w_9-w_3+w_1}$$

Кроме того необходимо отметить:

- Соблюдение условия неповторения ребра на стыке цикла также соблюдается.
- Хотя мы пишем $+, -$ в выражениях типа $D^{-w_4+w_5-w_2+w_1}$, w_i не коммутируют между собой так как неупорядоченный набор ребер не задает однозначно цикл (2.2), таким образом до подстановки конкретных значений w_i не могут быть сложены.

Пример 2.2. *Обозначим пути* $c_1 = 1 \to -4 \to 5 \to -2$, $c_2 = 1 \to -4 \to 6 \to -3$, $c_3 = 2 \to -5 \to 6 \to -3$. *Обратные к этим путям по направлению соответственно* c_{-1}, c_{-2}, c_{-3}. *Тогда набор ребер как объединением* $\{c_1, c_{-2}, c_{-3}\}$ *может обозначать два пути –* $c_1c_{-3}c_{-2}$ *и* $c_1c_{-2}c_{-3}$, *которые не могут быть получены друг из друга с помощью циклического сдвига и инверсии.*

Из (9) видим, что существуют 2 потенциальных цикла длины 4. Цикл образуется в том случае, когда сумма в показателе степени равна нулю. Например, если $-w_4 + w_5 - w_2 + w_1 = 0$, а $-w_8 + w_9 - w_3 + w_1 \neq 0$ то

$$a_{41}(D) = 1 + D^{-w_8+w_9-w_3+w_1},$$

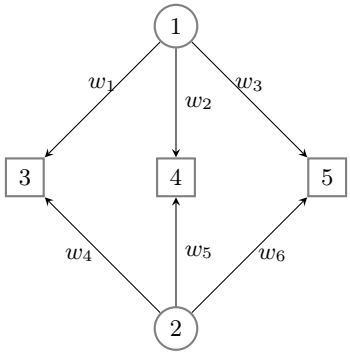

Рис. 2: Граф Таннера для кода из примеров 2.2, 2.3

и $a_{41}(0) = 1$. Пусть $a^j_{2L,i}$ – обозначет коэффициент при D^j в i-ой компонентете a_{2L}.

(здесь и далее считаем, что $D^0 = 1$, несмотря на то что D может быть 0. Таким образом свободный член полинома равен количеству циклов веса 0 без учета эквивалентных циклов) В общем случае все циклы длины $2L$, проходящие через ребро $+1$ учтены в $a_{2L,1}(0)$, однако некоторые из них могут быть учтены многократно (2.3). Кроме того необходимо аналогично учесть циклы, проходящие через ребро $+2$, $+3$ и так далее. Такое суммарное рассмотрение неизбежно будет учитывать циклы повторно, однако количество повторений не обязательно равно длине цикла и зависит от его состава.

Пример 2.3. *Пути* $1 \to -4 \to 5 \to -2 \to 1 \to -4 \to 6 \to -3$ *и* $1 \to -4 \to 6 \to -3 \to 1 \to -4 \to 5 \to -2$, *начинающийся с* 1 *эквивалентны, так как являются циклическим сдвигом друг друга, но будут учтены дважды.*

Дальнейшие рассуждения можно проводить двумя способами:

- Рассматривать w_i в качестве символьных переменных. Можно выписать все возможные комбинации весов, приводящие к потенциальным циклам длины $2L$ и затем вычислять спектры циклов, подставляя разные наборы разметок в полученные уравнения. Число циклов равно числу нулей в уравнениях.
- При заданных (фиксированных) весах ребер подсчитать спектр длин циклов в заданном диапазоне.

2.1 Веса как символьные переменные

В случае рассмотрения w_i как символьных переменных необходимо избавиться от эквивалентных относительно сдвига и инверсии комбинаций. Для этого каждый путь приводится к минимальному лексикографическому виду после чего убираются дубликаты.

Заметим, что цикл в обратном направлении к рассматриваемому циклу не может быть эквивалентет относительно сдвига исходному. Обозначим $\bar{p}$ путь в обратном направлении к пути $p = e_1, e_2, ..., e_n$. Предположим p и $\bar{p}$ эквивалентны относительно сдвига – тогда найдется индекс i, такой что $\bar{e}_i, \bar{e}_{i-1}, ..., \bar{e}_1 = e_1, e_2, ..., e_i$. Если i нечетно, тогда $\bar{e}_{(i+1)/2} = e_{(i+1)/2}$ – противоречие. Если i четно, то $e_{i/2+1} = \bar{e}_{i/2}$, что противоречит ограничению что путь не может идти обратно по последнему пройденному ребру.

Таким образом каждому циклу соответствует ровно два пути в разных направлениях. Непосредственно инвертируя каждый путь несложно оставить ровно один из каждой пары.

По причине того, что число различных путей растет экспоненциально при построении всех возможных символьных комбинаций, лучше воспользоваться подходом meet-in-the-middle. Это позволяет для нахождения всех возможных циклов длины $2L$ рассматривать только пути длины L, в то время как описанный алгоритм рассмотривает пути длины $2L$.

2.2 Фиксированные веса

В случае когда веса изначально зафиксированы можно считать, что веса коммутируют, поэтому достаточно считать не более M членов в каждом полиноме при умножении на матрицу A, по одному значению для каждой возможной суммы весов. После чего устранить дубликаты из результата с помощью обращения Мебиуса.

Рассмотрим для цикла p сколько раз он был учтен. Порядком цикла назовем максимальное r, такое что p можно представить как

$$p = \underbrace{ss\ldots s}_{r \text{ раз}}$$

Периодом цикла p назовем длину $|s| = \frac{|p|}{r}$. Таким образом цикл p периода l имеет l различных циклических сдвигов – следовательно будет учтен $2l$ раз, с учетом последовательностей в обратном порядке.

Последовательно для каждого i зафиксируем

$$a_0 = (\underbrace{0, 0, ..., 0}_{i \text{ раз}}, 1, 0, ..., 0)$$

и вычислим

$$a_{2L} = a_0 A^L$$

проводя все вычисления в кольце многочленов по модулю D^M.

Запомним количество циклов (всех возможных весов) проходящих через +i, содержащееся в многочлене $a_{2L,i}$ как $b_{2L,i}$.

Суммируя по всем возможным a_0 получаем

$$b_{2L} = \sum_i b_{2L,i}$$

- многочлен, с коэффициентами при D^w соответсвенно равными количеству циклов длины $2L$ и веса w, где каждый цикл учтен дважды столько, сколько он имеет различных циклических сдвигов (в обоих направлениях).

Далее временно забудем об ограничении инверсии, так как для устранения путей эквивалентных относительно разворота достаточно разделить результирующий спектр на два.

Обозначим за $g(l)$ – многочлен, коэффициент при D^w которого равен числу циклов длины и периода l веса w.

Определение 2.1. *Введем операцию $T_d(p(D))$ где $p(D)$ – многочлен, а d – натуральное число как:*

$$T_d(c_0 + c_1 D + c_2 D^2 + ... + c_n D^n) = c_0 + c_1 D^d + c_2 D^{2d} + ... + c_n D^{nd} \pmod{D^M}$$

Тогда при фиксированной длине L имеем равенство:

$$\sum_{d|L} d \cdot T_{\frac{L}{d}}(g(d)) = b_L$$

Равенство справедливо, так как каждый цикл длины L периода d веса W состоит из повторенного $\frac{L}{d}$ раз цикла длины и периода d веса w, такого что $w \cdot \frac{L}{d} = W \pmod{M}$ и однозначно им задается. Каждый такой цикл имеет d различных циклических сдвигов, поэтому учтен в b_L d раз. Таким образом суммирование ведется по всем возможных длинам периодов, после чего благодаря $T_{\frac{L}{d}}$ вес каждого цикла из $g(d)$ домножается на число повторений цикла для достижения длины L и каждый из циклов учитывается с коэффициентов d так как входит в b_L в виде d различных линейных последовательностей ребер.

В равенствах вида

$$f(n) = \sum_{d|n} g(d)$$

$g(d)$ может быть выражено с помощью формулы обращения Мебиуса:

$$g(n) = \sum_{d|n} \mu(d) f(n/d)$$

где

$$n = p_1^{e_1} \cdot p_2^{e_2} \cdot \ldots p_r^{e_r}$$

- разложение n на простые множители

$$\mu(n) = \begin{cases} 1 & \text{при } n = 1, \\ 0 & \text{если } \exists i \quad e_i > 1, \\ (-1)^r & \text{если } e_1 = e_2 = \ldots = e_r = 1 \end{cases}$$

В рассматриваемом случае можно ввести и использовать следующее обобщение формулы Мебиуса:

Теорема 2.1. *Если*

$$f(n) = \sum_{d|n} T_{\frac{n}{d}}(g(d))$$

где T_k удовлетворяет свойствам:

$$T_k(c \cdot p(D)) = c \cdot T_k(p(D)) \tag{6}$$
$$T_k(p(D) + q(D)) = T_k(p(D)) + T_k(q(D)) \tag{7}$$
$$T_{k_1}(T_{k_2}(p(D))) = T_{k_1 \cdot k_2}(p(D)) \tag{8}$$
$$T_1(p(D)) = p(D) \tag{9}$$

Тогда $g(d)$ может быть выражено:

$$g(n) = \sum_{d|n} \mu(d) T_d(f(n/d))$$

Доказательство.

Лемма 2.2.

$$\sum_{d|n} \mu(d) = 0 \quad при\ n > 1$$

Доказательство.

$$n = p_1^{e_1} \cdot p_2^{e_2} \cdot \ldots p_r^{e_r}$$

$$n^* = p_1 \cdot p_2 \ldots p_r$$

$$\sum_{d|n} \mu(d) = \sum_{d|n^*} \mu(d)$$

так как любое $\mu(d)$ где $p_i^2|d$ равно 0.

$$\sum_{d|n^*} \mu(d) = 1 - \binom{r}{1} + \binom{r}{2} - \ldots + (-1)^k \cdot \binom{r}{k} + \ldots = (1-1)^r = 0$$

так как существует $\binom{r}{k}$ делителей n^* состоящих из k простых множителей, каждый из которых внесет вклад $\mu(d) = (-1)^k$. □

$$\sum_{d|n} \mu(d) \cdot T_d(f(n/d)) =$$
$$\sum_{d|n} \mu(d) \cdot T_d(\sum_{d'|\frac{n}{d}} T_{\frac{n}{dd'}}(g(d'))) =$$
$$\sum_{d|n} \mu(d) \cdot \sum_{d'|\frac{n}{d}} T_d(T_{\frac{n}{dd'}}(g(d'))) =$$

$$\sum_{d|n} \mu(d) \cdot \sum_{d'|\frac{n}{d}} T_{\frac{n}{d'}}(g(d')) =$$

$$\sum_{d|n} \sum_{d'|\frac{n}{d}} \mu(d) \cdot T_{\frac{n}{d'}}(g(d')) =$$

$$\sum_{d'|n} \sum_{d|\frac{n}{d'}} \mu(d) \cdot T_{\frac{n}{d'}}(g(d')) =$$

$$\sum_{d'|n} T_{\frac{n}{d'}}(g(d')) \cdot \sum_{d|\frac{n}{d'}} \mu(d) =$$

$$T_{\frac{n}{n}}(g(n)) = T_1(g(n)) = g(n)$$

□

Нетрудно заметить что операция $T_d(p(D))$ из определения 2.1 удовлетворяет ограничениям из теоремы 2.1. Действительно, первые два свойства следуют из линейности кольца многочленов по модулю. Третье и четвертое свойства очевидно следуют из определения операции 2.1.

Таким образом, можно воспользоваться теоремой 2.1 для разрешения выражения 2.2 относительно $g(L)$. В качестве $f(L)$ выступает b_L, а в качестве $g(d)$ необходимо взять $d \cdot g(d)$, внеся множитель d внутрь операции T_d, согласно свойству один:

$$L \cdot g(L) = \sum_{d|L} \mu(d) T_d(b_{L/d})$$

$$g(L) = \frac{\sum_{d|L} \mu(d) T_d(b_{L/d})}{L}$$

Напомним, что $g(L)$ содержит многочлен, коэффициент при D^w которого равен числу циклов длины и периода L веса w. Обозначим результирующую величину количества циклов длины L веса 0 с учетом эквивалентности относительно сдвига за C_L. Для подсчета величины C_L с помощью $g(L)$ необходимо произвести суммирование по всем возможным длинам периодов d в циклах длины L и весам w таким, что период повторенный $\frac{L}{d}$ раз приведет к циклу нулевого веса:

$$C_L = \sum_{d|L} \sum_{\frac{w \cdot L}{d} = 0 \pmod M} g^{(w)}(0)$$

где $g^{(w)}$ – w-ая производная g, используемая для получения коэффициента при D^w.

Наконец для устранения дубликатов относительно инверсии достаточно разделить C_L на два.

2.3 Алгоритм

$$a_0^i = (\underbrace{0, 0, ..., 0}_{i \text{ раз}}, 1, 0, ..., 0) \tag{10}$$

$$a_{2L}^i = a_0^i \cdot A^L \tag{11}$$

$$b_{2L} = \sum_i a_{2L}^i \tag{12}$$

$$g(L) = \frac{\sum_{d|L} \mu(d) T_d(b_{L/d})}{L} \tag{13}$$

$$C_L = \sum_{d|L} \sum_{\frac{w \cdot L}{d} = 0 \pmod M} g^{(w)}(0) \tag{14}$$

2.4 Оценка сложности

Пусть исходная базовая матрица имела размер $b \times c$, а коэффициент расширения равен M. Для простоты будем рассматривать (J, K)-регулярный МППЧ код с весом столбцов и строк J и K соответственно.

Тогда число ребер в графе Таннера обозначим за $E = b \cdot K = c \cdot J$. Ограничение на максимальную длину в спектре обозначим S.

Таким образом размер матрицы A составляет $E \times E$, каждый из элементво которой представляет собой многочлен степени не больше M. И для получения всех интересующих A^L, посредством перемножения матриц, достаточно затратить $O(S \cdot E^3 \cdot M)$ операций, так как перемножение и сложение многочленов по модулю D^M требует $O(M)$ времени в отличии от $O(1)$ для обычных чисел.

При каждом фиксированном стартовом ребре i и длине циклов $2L$, многочлен $a_{2L}^i = a_0^i \cdot A^L$ **??** может быть получен за время $E^2 \cdot M$, посредством перемножения вектора a_0 длины E на матрицу A^L размера $E \times E$, опять же по причине того что перемножение и сложение многочленов по модулую требует $O(M)$ операций. Таким образом суммарно получение всех необходимых a_{2L} займет время $O(E \cdot S \cdot E^2 \cdot M) = O(S \cdot E^3 \cdot M)$.

Сложение всех необходимых a_{2L} для получения b_{2L} (**??**) может быть осуществленно за суммарный размер полиномов a, а именно $O(E \cdot S \cdot M)$.

Все необходимые значения функции Мебиуса $\mu(d)$ могут быть подсчитаны за время $O(S \cdot \ln S)$ с помощью решета Эратосфена. При оценке времени суммарно затраченного на подсчет $g(L)$ заметим, что суммирование ведется по всем делителям чисел L от 1 до S. Как изветно суммарное количество делителей чисел от 1 до n имеет порядок $O(n \cdot \ln n)$. Действительно, число d является делителем для $\left\lfloor \frac{n}{d} \right\rfloor$ чисел: $d, 2d, 3d, \ldots, \left\lfloor \frac{n}{d} \right\rfloor \cdot d$. Получаем сумму гармонического ряда $\sum_d \left\lfloor \frac{n}{d} \right\rfloor = O(n \cdot \ln n)$. Таким образом для подсчета всех $g(L)$ необходимо затратить $O(M \cdot S \cdot \ln S)$, так как операция T_d и сложение многочленов степени M может быть осуществлено за $O(M)$.

Пользуясь оценкой суммы гармонического ряда для **??** с учетом суммирования по весам получаем время необходимое для подсчета C_L – $O(M \cdot S \cdot$

$\ln S$), так как теперь складываются коэффициенты многочлена, а не целые многочлены.

Итого по всем шагам алгоритма получаем:

$$O(S \cdot E^3 \cdot M) + O(S \cdot E^3 \cdot M) + O(E \cdot S \cdot M) + O(M \cdot S \cdot \ln S) + O(M \cdot S \cdot \ln S) =$$

$$O(S \cdot E^3 \cdot M) + O(M \cdot S \cdot \ln S) = O(M \cdot S \cdot max(\ln S, E^3))$$

Очевидно для всех разумных входных данных член E^3 мажорирует $\ln S$ таким образом итоговая сложность:

$$O(M \cdot S \cdot E^3)$$

3 Численный анализ влияния спектров циклов на вероятность ошибки БП-декодирования

- описание ансамблей кодов
- описание эксперимента
- несколько (4-6) графиков из Матлаба или TikZ
- выводы по графикам