

ЛАБОРАТОРНАЯ РАБОТА № 2

Исследование спектральных характеристик сверточных кодов методом производящих функций

2.1. Производящие функции и их свойства

Рассмотрим дискретное множество $X = \{x_1, \dots, x_N\}$, содержащее N элементов. Пусть на этом множестве определена некоторая функция $f(x)$, принимающая значения из дискретного множества $Z = \{z\}$, представляющего собой подмножество множества целых чисел. Обозначим через N_z количество элементов x множества X , для которых $f(x) = z$. Полином вида

$$T(D) = \sum_{z \in Z} N_z D^z \quad (2.1)$$

называется производящей функцией для $f(x)$.

Пример 1. Пусть $Z = \{1, \dots, 6\}$ – множество исходов бросания игральной кости. Порождающая функция числа очков при одном бросании

$$T_1(D) = D + D^2 + \dots + D^6 = \frac{D^7 - D}{D - 1}.$$

Пример 2. Рассмотрим игру в орлянку, при которой выпадение орла уменьшает капитал одного из игроков на единицу, выпадение решетки – увеличивает капитал на единицу. В этом случае $Z = \{-1, 1\}$ и производящая функция исходов одного бросания равна

$$T_2(D) = D^{-1} + D.$$

Главное свойство производящих функций, благодаря которому их используют для анализа сумм функций, состоит в том, что производящая функция суммы двух функций равна произведению производящих функций слагаемых. Действительно, если $T_U(D) = \sum_{u \in U} N_u D^u$ и $T_V(D) = \sum_{v \in V} N_v D^v$ – две производящие функции, то их произведением будет

$$T_U(D)T_V(D) = \sum_{u \in U} \sum_{v \in V} N_u N_v D^{u+v} = \sum_z \left(\sum_{(u,v): u+v=z} N_u N_v \right) D^z = T_Z(D),$$

где в правой части мы получили производящую функцию суммы $z = u + v$.

Пример 3. Монету бросают 10 раз. Каково число комбинаций исходов, при котором выигрыш игрока составит 6 единиц? Для ответа на этот вопрос запишем производящую функцию суммы исходов 10 опытов в виде

$$T_3(D) = T_2(D)^{10} = D^{-10} (1 + C_{10}^1 D^2 + C_{10}^2 D^4 + \dots + C_{10}^9 D^{18} + D^{20}).$$

Поскольку коэффициент при D^6 равен $C_{10}^8 = 120$, именно в таком числе исходов из общего числа 1024 всевозможных исходов игрок получит доход в 6 единиц.

2.2. Применение производящих функций для анализа путей в направленных графах

На рис 2.1 приведен простой пример направленного графа. Числа, сопоставленные ребрам графа, интерпретируются как длины ребер. Задача анализа графа состоит в подсчете числа путей длины 0, 1, 2, ... из узла 0 в узел 3. Обозначим через N_i число путей суммарной длины i . Непосредственный перебор по множеству путей показывает, что $N_0 = N_1 = N_2 = N_3 = 0$, $N_4 = 1, \dots$. Для этого простого примера нетрудно вывести общую

формулу для N_i , но понятно, что для более сложного графа задача окажется трудной. Мы используем этот простой пример для иллюстрации метода производящих функций.

Обозначим через $g_s(D)$ производящую функцию длин путей из узла 0 в узел с номером s . Задача состоит в нахождении $g_3(D)$. Рассмотрим узел 1. В этот узел ведет один путь длины 1 из узла 0 и множество путей, проходящих по ребру длины 3 из узла 2. Поскольку суммированию длин соответствует перемножение производящих функций, производящая функция для путей из 0 равна D , а для путей из 2 она равна $D^3 g_2(D)$. В итоге имеем

$$g_1(D) = D + D^3 g_2(D).$$

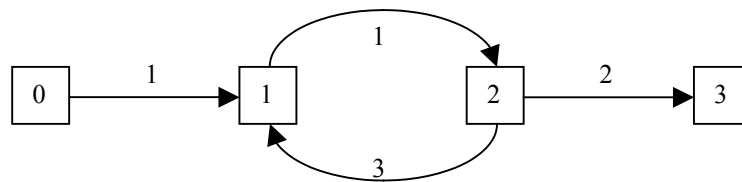


Рис. 2.1. Пример направленного графа

Аналогично для узлов 2 и 3 имеем

$$g_2(D) = Dg_1(D);$$

$$g_3(D) = D^2 g_2(D).$$

Решая систему из трех уравнений, находим

$$g_3(D) = \frac{D^4}{1 - D^4}.$$

Чтобы найти коэффициенты производящей функции в явном виде, воспользуемся формулой

$$\frac{1}{1-x} = 1 + x + x^2 + \dots \quad (|x| < 1).$$

Окончательный результат имеет вид

$$g_3(D) = D^4 + D^8 + D^{12} + \dots$$

Отсюда следует, что, как можно было предвидеть, в нашем графе есть один путь из 0 в 3 длины 4, один путь длины 8 и т.д.

2.3. Вычисление производящей функции спектра сверточного кода

Решение задачи будет рассмотрено на примере кода (5,7) со скоростью 1/2. На рис.2.2 показана схема кодера и на рис. 2.3 – соответствующий кодеру конечный автомат. При подсчете спектра расстояний сверточного кода для нас не важна конкретная последовательность, сопоставленная конкретному ребру графа, важен только ее вес. Поэтому граф автомата приведен к виду, показанному на рис. 2.4, где ребрам приписаны производящие функции весов соответствующих последовательностей. Помимо этого,

исключен переход из состояния 0 в состояние 0, поскольку нашей целью является анализ множества кодовых последовательностей первого неправильного поддеревья, т.е. последовательностей, отличающихся от нулевой последовательности в первом подблоке. Как и прежде, обозначим через $g_s(D)$ производящую функцию веса путей из узла 0 в узел с номером s . Задача теперь состоит в нахождении $g_0(D)$.

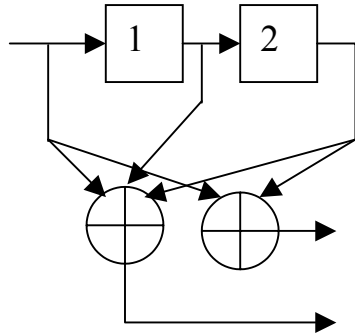


Рис. 2.2. Кодер кода (5,7)

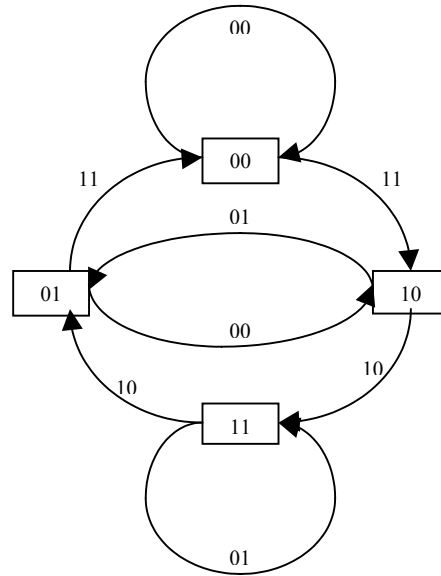


Рис. 2.3. Диаграмма состояний кодера кода (5,7)

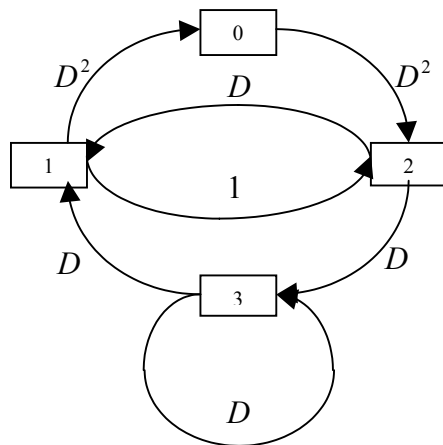


Рис. 2.4. Преобразованная диаграмма состояний кодера кода (5,7)

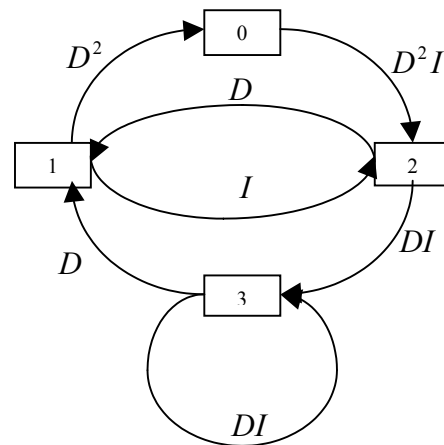


Рис. 2.5. Диаграмма состояний кодера кода (5,7) для подсчета расширенной производящей функции

Аналогично рассмотренному выше примеру получаем систему уравнений

$$\begin{cases} g_0(D) = D^2 g_1(D) \\ g_1(D) = Dg_2(D) + Dg_3(D) \\ g_2(D) = D^2 + g_1(D) \\ g_3(D) = Dg_2(D) + Dg_3(D) \end{cases} .$$

Решением системы является производящая функция

$$g_0(D) = \frac{D^5}{1-2D} = D^5 + 2D^6 + 4D^7 + \dots .$$

Итак, наши вычисления показывают, что среди кодовых слов первого неправильного поддерева имеется 1 слово веса 5, 2 слова веса 6, 4 слова веса 7 и т.д.

Как уже говорилось, при расчете помехоустойчивости сверточных кодов важно не только расстояние между кодовыми словами, но и количество искаженных информационных символов, обусловленных принятием ошибочного решения в пользу того или иного кодового слова. Таким образом, каждый путь в графе мы можем характеризовать двумя параметрами: вес Хэмминга соответствующего кодового слова и вес Хэмминга соответствующей информационной последовательности.

Метод производящих функций, который был выше описан для случая одного дискретного множества чисел, легко обобщается для совместного анализа двух и большего числа множеств. В частности, если обозначить через $t(d, i)$ число кодовых слов, порождаемых информационными последовательностями веса i и имеющих вес Хэмминга равный d , то производящая функция числа таких путей определяется как следующая функция двух формальных переменных D и I

$$T(D, I) = \sum_{d=1}^{\infty} \sum_{i=1}^{\infty} t(d, i) D^d I^i . \quad (2.2)$$

Эту производящую функцию называют расширенной производящей функцией сверточного кода. Найдем $T(D, I)$ для кода (7,5). Для этого каждому ребру конечного автомата, описывающего работу кодера, сопоставим вычисленную для этого ребра совместную производящую функцию веса кодового блока и веса соответствующей информационной последовательности. Размеченная таким образом диаграмма состояний приведена на рис.2.5.

Пусть $g_s(D, I)$ обозначает расширенную производящую функцию для путей, ведущих из состояния 0 в состояние s . Искомая производящая функция равна $T(D, I) = g_0(D, I)$. По диаграмме состояний кодера составляем систему уравнений для нахождения $g_0(D, I)$

$$\begin{cases} g_0(D, I) = D^2 g_1(D, I) \\ g_1(D, I) = Dg_2(D, I) + Dg_3(D, I) \\ g_2(D, I) = D^2 I + Ig_1(D, I) \\ g_3(D, I) = DIg_2(D, I) + DIg_3(D, I) \end{cases} .$$

Решив систему, получим

$$T(D, I) = g_0(D, I) = \frac{D^5 I}{1-2DI} = D^5 + 2D^6 I^2 + 4D^7 I^3 + 8D^8 I^4 \dots . \quad (2.3)$$

Этот результат означает, что код содержит одно слово веса 1, порождаемое информационной последовательностью веса 1. Каждое из 2 слов веса 6 порождается информационной последовательностью веса 2 и т.д.

Для практических расчетов вероятности ошибок часто используется производящая функция

$$F(D) = \sum_{d=1}^{\infty} f_d D^d, \quad (2.4)$$

в которой f_d обозначает суммарное количество единиц в информационных последовательностях, порождающих кодовые слова веса d . Понятно, что

$$f_d = \sum_{i=1}^{\infty} it(d, i). \quad (2.5)$$

Отсюда следует формула, связывающая функции $T(D, I)$ и $F(D)$:

$$F(D) = \sum_{d=1}^{\infty} \sum_{i=1}^{\infty} it(d, i) D^d = \left. \frac{\partial}{\partial I} T(D, I) \right|_{I=1}. \quad (2.6)$$

Из (2.3) и (2.6) для кода с генераторами (7,5) получаем

$$F(D) = D^5 + 4D^6 + 12D^7 + 32D^8 + \dots$$

Понятно, что даже для кода с кодовым ограничением 2 применение метода производящих функций позволило существенно упростить задачу вычисления спектральных характеристик кода по сравнению с перебором по множеству кодовых слов кода. В то же время, для получения функций $F(D)$ и $T(D)$ мы решили в символьном виде систему из 4 линейных уравнений. Для кода с кодовым ограничением 3 число уравнений равно 8 и т.д. В общем случае число уравнений равно 2^v , где v - кодовое ограничение кода. Для практически интересных длин кодового ограничения задача относительно просто решается с помощью ЭВМ. В данной лабораторной работе один из методов поиска спектра кодов, реализованный в виде программы для IBM PC используется для проверки правильности вычислений, выполненных студентами.

2.4. Порядок выполнения работы

1. Для кодера, выбранного по таблице 1 в соответствии с номером варианта задания, построить диаграмму состояний конечного автомата и разметить ее порождающими функциями весов переходов.
2. Найти порождающие функции $T(D)$ и $F(D)$.
3. Проверить правильность вычислений с помощью тестирующей программы.
4. Оформить отчет.

2.5. Контрольные вопросы

1. Монету бросают до первого выпадения орла. Постройте производящую функцию числа бросаний.
2. Полезен ли спектр весов кодовых слов при анализе характеристик нелинейных кодов? Возможно ли применение описанного в разделе 2.3 метода нахождения производящих функций для анализа катастрофических кодов?
3. Оцените скорость роста сложности вычисления производящей функции спектра кода с ростом длины кодового ограничения.
4. Выпишите в матричном виде уравнение для вектора $\mathbf{g}(D) = (g_0(D), \dots, g_{2^v-1}(D))$ и запишите в общем виде его решение. Свяжите условие существования решения с условием катастрофичности.