ТЕОРИЯ КОДИРОВАНИЯ

Крук К.Ю.

Белорусский государственный университет информатики и радиоэлектроники
г. Минск, Республика Беларусь

Русин Виталий Геннадьевич-зам.декана ФКСИС

Код Хэмминга позволяет закодировать какое-либо информационное сообщение определённым образом и после передачи (например по сети) определить появилась ли какая-то ошибка в этом сообщении (к примеру из-за помех) и, при возможности, восстановить это сообщение.

Код Хэмминга — это самоконтролирующийся и самокорректирующийся код, построенный для обеспечения надежности передачи данных в двоичной системе счисления. Он позволяет обнаруживать и исправлять одиночные ошибки (ошибки в одном бите слова) и находить двойные ошибки. Таким образом, код Хэмминга тесно связан с информатикой и является важным инструментом для обеспечения целостности и надежности передачи данных.

Код Хэмминга.

Всякий раз, когда данные передаются или сохраняются, возможно, что они могут быть повреждены. Это может принимать форму переворота битов , когда двоичная 1 становится 0 или наоборот. Коды, исправляющие ошибки, направлены на обнаружение ошибок в некоторых данных. Это делается путем добавления к данным битов четности или избыточной информации. Если добавлено достаточное количество данных четности, это обеспечивает прямое исправление ошибок, при котором ошибки могут автоматически исправляться при обратном чтении. Код Хэмминга использует механизм блочной четности. Данные делятся на блоки, и к блоку добавляется четность. Код Хэмминга может исправлять однобитовые ошибки и обнаруживать наличие двухбитных ошибок в блоке данных. Количество данных четности, добавляемых в код Хэмминга, определяется формулой:

$$2^p \geq d + p + 1 \quad (1)$$

где p — количество битов четности, а d — количество битов данных.

Наибольший интерес представляют двоичные блочные корректирующие коды. При использовании таких кодов информация передаётся в виде блоков одинаковой длины, и каждый блок кодируется и декодируется независимо от другого. Почти во всех блочных кодах символы можно разделить на информационные и проверочные или контрольные. Таким образом, все слова разделяются на разрешённые (для которых соотношение информационных и проверочных символов возможно) и запрещённые.

Граница Хэмминга устанавливает максимально возможное число разрешённых кодовых комбинаций:

$$2^k \geq 2^n / \sum_{i=0}^{\frac{d-1}{2}} C_n^i \quad (2)$$

Все вышеперечисленные оценки дают представление о верхней границе d при фиксированных n и k или оценку снизу числа проверочных символов.

Построение кодов Хэмминга основано на принципе проверки на чётность числа единичных символов: к последовательности добавляется такой элемент, чтобы число единичных символов в получившейся последовательности было чётным:

$$r_1 = i_1 \oplus i_2 \oplus \cdots \oplus i_k \quad (3)$$

$$S = i_1 \oplus i_2 \oplus \cdots \oplus i_n \oplus r_1 \quad (4)$$

Если S=0 — то ошибки нет, если S=1 — то однократная ошибка. Такой код называется (k+1, k) или (n, n-1). Первое число — количество элементов последовательности, второе — количество информационных символов. Для каждого числа проверочных символов r=3, 4, 5... существует классический код Хэмминга с маркировкой:

$$(n,\ k) = (2^r - 1,\ 2^r - 1 - r\ ), \tag{5}$$

то есть - (7,4), (15,11), (31,26).

Для примера рассмотрим классический код Хемминга *(7, 4)*. Сгруппируем проверочные символы следующим образом:

$$r_1 = i_1 \oplus i_2 \oplus i_3 \tag{6}$$

$$r_2 = i_2 \oplus i_3 \oplus i_4 \tag{7}$$

$$r_3 = i_1 \oplus i_2 \oplus i_4 \tag{8}$$

Получение кодового слова выглядит следующим образом:

$$\begin{pmatrix} i_1 & i_2 & i_3 & i_4 \end{pmatrix} \begin{pmatrix} 1 & 0 & 0 & 0 & 1 & 0 & 1 \\ 0 & 1 & 0 & 0 & 1 & 1 & 1 \\ 0 & 0 & 1 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 1 & 1 \end{pmatrix} = \begin{pmatrix} i_1 & i_2 & i_3 & i_4 & r_1 & r_2 & r_3 \end{pmatrix}. \tag{9}$$

На вход декодера поступает кодовое слово $V = (i_1', i_2', i_3', i_4', r_1', r_2', i_3')$,где штрихом помечены символы, которые могут исказиться в результате действия помехи. В декодере в режиме исправления ошибок строится последовательность синдромов:

$$S_1 = r_1 \oplus i_1 \oplus i_2 \oplus i_3 \tag{10}$$

$$S_2 = r_2 \oplus i_2 \oplus i_3 \oplus i_4 \tag{11}$$

$$S_3 = r_3 \oplus i_1 \oplus i_2 \oplus i_4 \tag{12}$$

$S = (S_1,\ S_2,\ S_3)$называется синдромом последовательности. Получение синдрома выглядит следующим образом:

$$\begin{pmatrix} i_1 & i_2 & i_3 & i_4 & r_1 & r_2 & r_3 \end{pmatrix} \begin{pmatrix} 1 & 0 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 0 \\ 0 & 1 & 1 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} = \begin{pmatrix} S_1 & S_2 & S_3 \end{pmatrix}. \tag{13}$$

Стандартный код Хэмминга может обнаружить и исправить только однобитовую ошибку. Если два бита ошибочны, возможно, что эти две ошибки будут выглядеть как однобитовая ошибка. Чтобы учесть это, можно добавить дополнительный общий бит четности для надежного обнаружения ошибок в двух битах. Это известно как исправление одиночных ошибок/обнаружение двойных ошибок.

Пример кодирования и нахождение ошибок.

ЗАДАНИЕ №1. Пользуясь кодом Хэмминга найти ошибку в сообщении.

1) 1111 1011 0010 1100 1101 1100 110

РЕШЕНИЕ. Сообщение состоит из 27 символов, из них 22 информационные, а 5 – контрольные. Это разряды b1 = 1, b2 = 1, b4 = 1, b8 = 1, b16=0.

Вычислим число J для обнаружения ошибки:

Введем для удобства следующие множества:

V1 = 1, 3, 5, 7, 9, 11, 13, 15, 17, 19, 21, 23, 25, 27… - все числа у которых первый разряд равен 1

V2 = 2, 3, 6, 7, 10, 11, 14, 15, 18, 19, 22, 23, 26, 27… - все числа, у которых второй разряд равен 1

V3 = 4, 5, 6, 7, 12, 13, 14, 15, 20, 21, 22, 23 … - все числа, у которых третий разряд равен 1

V4 = 8, 9, 10, 11, 12, 13, 14, 15, 24, 25, 26, 27 … - все числа, у которых четвертый разряд равен 1,

V5 = 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27 … - все числа, у которых пятый разряд равен 1.

Разряды числа J определяются следующим образом:

j1 = b1 + b3+b5+b7+b9+b11+b13+b15+b17+b19+b21+b23+b25+b27 = 1

j2 = b2+ b3+b6+b7+b10+b11+b14+ b15+ b18+ b19+ b22+ b23+ b26+ b27= 0

j3 = b4+ b5+b6+b7 +b12+b13+ b14+ b15+ b20 +b21+b22+b23 = 0

j4 = b9+b10+b11+b12+b13+b14+b15+b24+b25+b26+b27 = 0,

j5 = b16+ b17+b18+b19+b20+b21+b22+b23+b24+b25+b26+b27 = 1

то есть число J=100012 = 1710.

Таким образом, ошибка произошла в семнадцатом разряде переданного числа, следует 1 заменить на 0. Получим 1111 1011 0010 1100 0101 1100 110

Теперь удалим контрольные разряды. Получим 1101 0010 1100 1011 1001 10 - переданное число.

ЗАДАНИЕ №2. Закодировать данное слово кодом Хэмминга.

1) 1001 0001 1101 1110 0000 000 РЕШЕНИЕ.

Для кодирования данного сообщения длиной m = 23 потребуется k = 5 дополнительных разряда, т.е. на выходе получим сообщение длиной n = 28 (количество дополнительных разрядов подбирали из соотношения $2k \geq n+1$, n – число полученных разрядов, k – число дополнительных разрядов).

Пусть закодированное сообщение имеет вид b28 b27 b26 b25 b24 b23 b22 b21 b20 b19 b18 b17 b16 b15 b14 b13 b12 b11 b10 b9 b8 b7 b6 b5 b4 b3 b2 b1, причем разряды b1, b2, b4, b8, b16 будут контрольными, а остальные информационными.

Помещаем в информационные разряды разряды исходного числа по порядку, т.е.

b3 =1, b5 = 0, b6 = 0, b7=1,

b9=0, b10 =0, b11=0, b12=1,

b13= 1, b14=1, b15=0, b17=1,

b18=1, b19=1, b20=1, b21=0,

b22=0, b23=0, b24=0, b25=0,

b26=0, b27=0, b28=0.

Теперь найдем значения контрольных разрядов.

Введем для удобства следующие множества:

V1 = 1, 3, 5, 7, 9, 11, 13, 15, 17, 19, 21, 23, 25, 27… - все числа у которых первый разряд равен 1

V2 = 2, 3, 6, 7, 10, 11, 14, 15, 18, 19, 22, 23, 26, 27… - все числа, у которых второй разряд равен 1

V3 = 4, 5, 6, 7, 12, 13, 14, 15, 20, 21, 22, 23, 28… - все числа, у которых третий разряд равен 1

V4 = 8, 9, 10, 11, 12, 13, 14, 15, 24, 25, 26, 27, 28… - все числа, у которых четвертый разряд равен 1,

V5 = 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28 … - все числа, у которых пятый разряд равен 1.

Далее под + будем понимать сложение по модулю 2.

Тогда b1 = b3+b5+b7+b9+b11+b13+b15+b17+b19+b21+b23+b25+b27 = 1 (все разряды из V1, кроме первого)

b2 = b3+b6+b7+b10+b11+b14+ b15+ b18+ b19+ b22+ b23+ b26+ b27 = 1 (все разряды из V2, кроме первого)

b4 = b5+b6+b7 +b12+b13+ b14+ b15+ b20 +b21+b22+b23+b28 = 1 (все разряды из V3, кроме первого)

b8 = b9+b10+b11+b12+b13+b14+b15+b24+b25+b26+b27+b28 = 1 (все разряды из V4, кроме первого),

b16 = b17+b18+b19+b20+b21+b22+b23+b24+b25+b26+b27+b28 = 0 (все разряды из V5, кроме первого).

Таким образом, получили код 1111 0011 0001 1100 1111 0000 0000.

Заключение

Код Хэмминга используется в ситуациях, когда согласованность важнее эффективности передачи. Эффективность его передачи возрастает по мере увеличения размера блока. В Хэмминге(7, 4) эффективная скорость передачи данных составляет всего 0,571.

Код Хэмминга относительно прост в использовании и может быть реализован аппаратно. Это также означает, что вычисления выполняются быстро. Эти свойства делают его идеальным для использования в компьютерной памяти с коррекцией ошибок, поскольку в оперативной памяти компьютера может возникнуть ошибка или переворот битов из-за радиации или космических лучей, попадающих в ячейку памяти.

Код Хэмминга — это широко используемый метод исправления ошибок, который используется в различных приложениях, включая телекоммуникации, компьютерные сети и системы хранения данных.

код Хэмминга способен исправлять однобитовую ошибку, что делает его идеальным для использования в приложениях, где ошибки могут возникнуть из-за внешних факторов, таких как электромагнитные помехи.

код Хэмминга может исправить только ограниченное количество множественных ошибок. В приложениях, где вероятно возникновение нескольких ошибок, могут потребоваться более совершенные методы исправления ошибок.

Одной из областей, где код Хэмминга используется для передачи данных, является спутниковая и космическая связь. Из-за больших расстояний, длительного времени передачи и требований к точным данным предпочтительнее использовать более медленный, но более точный код Хэмминга и жертвовать общей скоростью передачи.

Список использованных источников:

TechTarget[Электронный ресурс]. – Режим доступа: https://www.geeksforgeeks.org/hamming-code-in-computer-network – Дата доступа: 13.04.2024.

Hamming code[Электронный ресурс]. – Режим доступа: https://en.wikipedia.org/wiki/Hamming_code – Дата доступа: 13.04.2024.

Авдошин С.М., Набебин А.А. Дискретная математика. Модулярная алгебра, криптография, кодирование. — М.: ДМК Пресс, 2017. -352 с.

Блейхут Р. Теория и практика кодов, контролирующих ошибки. Пер. с англ. М.: Мир, 1986, 576 с.