Обнаружение и исправление ошибок

Материал из Википедии — свободной энциклопедии

Обнаружение ошибок в технике связи — действие, направленное на контроль целостности данных при записи/воспроизведении информации или при её передаче по линиям связи. **Исправление ошибок (коррекция ошибок)** — процедура восстановления информации после чтения её из устройства хранения или канала связи.

Для обнаружения ошибок используют **коды обнаружения ошибок**, для исправления — **корректирующие коды (коды, исправляющие ошибки, коды с коррекцией ошибок, помехоустойчивые коды)**.

Содержание

Способы борьбы с ошибками

В процессе хранения данных и передачи информации по сетям связи неизбежно возникают ошибки. Контроль целостности данных и исправление ошибок — важные

задачи на многих уровнях работы с информацией (в частности, физическом, канальном, транспортном уровнях модели OSI).

В системах связи возможны несколько стратегий борьбы с ошибками:

- обнаружение ошибок в блоках данных и *автоматический запрос повторной передачи* повреждённых блоков — этот подход применяется в основном на канальном и транспортном уровнях;
- обнаружение ошибок в блоках данных и отбрасывание повреждённых блоков — такой подход иногда применяется в системах потокового мультимедиа, где важна задержка передачи и нет времени на повторную передачу;
- *исправление ошибок* (*forward error correction*) применяется на физическом уровне.

Коды обнаружения и исправления ошибок

Корректирующие коды — коды, служащие для обнаружения или исправления ошибок, возникающих при передаче информации под влиянием помех, а также при её хранении.

Для этого при записи (передаче) в полезные данные добавляют специальным образом структурированную *избыточную* информацию (контрольное число), а при чтении (приёме) её используют для того, чтобы обнаружить или исправить ошибки. Естественно, что число ошибок, которое можно исправить, ограничено и зависит от конкретного применяемого кода.

С *кодами, исправляющими ошибки*, тесно связаны *коды обнаружения ошибок*. В отличие от первых, последние могут только установить факт наличия ошибки в переданных данных, но не исправить её.

В действительности, используемые коды обнаружения ошибок принадлежат к тем же классам кодов, что и коды, исправляющие ошибки. Фактически, любой код, исправляющий ошибки, может быть также использован для обнаружения ошибок (при этом он будет способен обнаружить большее число ошибок, чем был способен исправить).

По способу работы с данными коды, исправляющие ошибки делятся на *блоковые*, делящие информацию на фрагменты постоянной длины и обрабатывающие каждый из них в отдельности, и *свёрточные*, работающие с данными как с непрерывным потоком.

Блоковые коды

Пусть кодируемая информация делится на фрагменты длиной k бит, которые преобразуются в *кодовые слова* длиной n бит. Тогда соответствующий блоковый код обычно обозначают $(n,\ k)$. При этом число $R = \frac{k}{n}$ называется *скоростью кода*.

Если исходные k бит код оставляет неизменными, и добавляет $n - k$ *проверочных*, такой код называется *систематическим*, иначе *несистематическим*.

Задать блоковый код можно по-разному, в том числе таблицей, где каждой совокупности из k информационных бит сопоставляется n бит кодового слова. Однако, хороший код должен удовлетворять, как минимум, следующим критериям:

- способность исправлять как можно большее число ошибок,
- как можно меньшая избыточность,
- простота кодирования и декодирования.

Нетрудно видеть, что приведённые требования противоречат друг другу. Именно поэтому существует большое количество кодов, каждый из которых пригоден для своего круга задач.

Практически все используемые коды являются линейными. Это связано с тем, что нелинейные коды значительно сложнее исследовать, и для них трудно обеспечить приемлемую лёгкость кодирования и декодирования.

Линейные коды общего вида

Линейный блоковый код — такой код, что множество его *кодовых слов* образует *k*-мерное линейное подпространство (назовём его *C*) в *n*-мерном линейном пространстве, изоморфное пространству *k*-битных векторов.

Это значит, что операция кодирования соответствует умножению исходного *k*-битного вектора на невырожденную матрицу *G*, называемую *порождающей матрицей*.

Пусть $C^{\perp}$ — ортогональное подпространство по отношению к *C*, а *H* — матрица, задающая базис этого подпространства. Тогда для любого вектора $\vec{v} \in C$ справедливо:

$$\vec{v} H^T = \vec{0}.$$

Минимальное расстояние и корректирующая способность

Расстоянием Хемминга (*метрикой Хемминга*) между двумя кодовыми словами $\vec{u}$ и $\vec{v}$ называется количество отличных бит на соответствующих позициях, $d_H(\vec{u},\ \vec{v}) = \sum_s |u^{(s)} - v^{(s)}|$, что равно числу «единиц» в векторе $\vec{u} \oplus \vec{v}$.

Минимальное расстояние Хемминга $d_{\min} = \min_{u \neq v} d_H(\vec{u},\ \vec{v})$ является важной характеристикой линейного блокового кода. Она показывает насколько «далеко» расположены коды друг от друга. Она определяет другую, не менее важную характеристику — *корректирующую способность*:

$t = \left\lfloor \frac{d_{\min} - 1}{2} \right\rfloor$, округляем «вниз», так чтобы $2t < d_{\min}$.

Корректирующая способность определяет, сколько ошибок передачи кода (типа $1 \leftrightarrow 0$) можно *гарантированно* исправить. То есть вокруг каждого кода *A* имеем *t*-окрестность A_t, которая состоит из всех возможных вариантов передачи кода *A* с числом ошибок ($1 \leftrightarrow 0$) не более *t*. Никакие две окрестности двух любых кодов не пересекаются друг с другом, так как расстояние между кодами (то есть центрами этих окрестностей) всегда больше двух их *радиусов* $d_H(A,\ B) \geqslant d_{\min} > 2t$.

Таким образом получив искажённый код из A_t декодер принимает решение, что был исходный код A, исправляя тем самым не более t ошибок.

Поясним на примере. Предположим, что есть два кодовых слова A и B, расстояние Хемминга между ними равно 3. Если было передано слово A, и канал внёс ошибку в одном бите, она может быть исправлена, так как даже в этом случае принятое слово ближе к кодовому слову A, чем к любому другому, и в частности к B. Но если каналом были внесены ошибки в двух битах (в которых A отличалось от B) то результат ошибочной передачи A окажется ближе к B, чем A, и декодер примет решение что передавалось слово B.

Коды Хемминга

Коды Хемминга — простейшие линейные коды с минимальным расстоянием 3, то есть способные исправить одну ошибку. Код Хемминга может быть представлен в таком виде, что *синдром*

$\vec{s} = \vec{r} H^T$, где $\vec{r}$ — принятый вектор, будет равен номеру позиции, в которой произошла ошибка. Это свойство позволяет сделать декодирование очень простым.

Общий метод декодирования линейных кодов

Любой код (в том числе нелинейный) можно декодировать с помощью обычной таблицы, где каждому значению принятого слова $\vec{r}_i$соответствует наиболее вероятное переданное слово $\vec{u}_i$. Однако, данный метод требует применения огромных таблиц уже для кодовых слов сравнительно небольшой длины.

Для линейных кодов этот метод можно существенно упростить. При этом для каждого принятого вектора $\vec{r}_i$вычисляется *синдром* $\vec{s}_i = \vec{r}_i H^T$. Поскольку $\vec{r}_i = \vec{v}_i + \vec{e}_i$, где $\vec{v}_i$ — кодовое слово, а $\vec{e}_i$ — вектор ошибки, то $\vec{s}_i = \vec{e}_i H^T$. Затем с помощью таблицы по синдрому определяется вектор ошибки, с помощью которого определяется переданное кодовое слово. При этом таблица получается гораздо меньше, чем при использовании предыдущего метода.

Линейные циклические коды

Несмотря на то, что декодирование линейных кодов уже значительно проще декодирования большинства нелинейных, для большинства кодов этот процесс всё ещё достаточно сложен. Циклические коды, кроме более простого декодирования, обладают и другими важными свойствами.

Циклическим кодом является линейный код, обладающий следующим свойством: если $\vec{v}$ является кодовым словом, то его циклическая перестановка также является кодовым словом.

Слова циклического кода удобно представлять в виде многочленов. Например, кодовое слово $\vec{v} = (v_0,\ v_1,\ \ldots,\ v_{n-1})$представляется в виде полинома $v(x) = v_0 + v_1 x + \ldots + v_{n-1} x^{n-1}$. При этом циклический сдвиг кодового слова эквивалентен умножению многочлена на x по модулю $x^n - 1$.

В дальнейшем, если не указано иное, мы будем считать, что циклический код является *двоичным*, то есть $v_0, v_1, \cdots$ могут принимать значения *0* или *1*.

Порождающий (генераторный) полином

Можно показать, что все кодовые слова конкретного циклического кода кратны определённому *порождающему полиному* $g(x)$. Порождающий полином является делителем $x^n - 1$.

С помощью порождающего полинома осуществляется кодирование циклическим кодом. В частности:

- *несистематическое кодирование* осуществляется путём умножения кодируемого вектора на $g(x)$: $v(x) = u(x)g(x)$;
- *систематическое кодирование* осуществляется путём «дописывания» к кодируемому слову остатка от деления $x^{n-k}u(x)$ на $g(x)$, то есть $v(x) = x^{n-k}u(x) + [x^{n-k}u(x) \bmod g(x)]$.

Коды CRC

Коды CRC (cyclic redundancy check — циклическая избыточная проверка) являются *систематическими* кодами, предназначенными не для исправления ошибок, а для их обнаружения. Они используют способ систематического кодирования, изложенный выше: «контрольная сумма» вычисляется путем деления $x^{n-k}u(x)$ на $g(x)$. Ввиду того, что исправление ошибок не требуется, проверка правильности передачи может производиться точно так же.

Таким образом, вид полинома $g(x)$ задаёт конкретный код CRC. Примеры наиболее популярных полиномов:

название кода	**степень полином**	
CRC-12	12	$x^{12} + x^{11} + x^3 + x^2 + x + 1$
CRC-16	16	$x^{16} + x^{15} + x^2 + 1$
CRC-CCITT	16	$x^{16} + x^{12} + x^5 + 1$
CRC-32	32	$x^{32} + x^{26} + x^{23} + x^{22} + x^{16} + x^{12} + x^{11} + x^{10} + x^8 + x^7 + x^5 + x^4 + x^2 + x + 1$

Коды БЧХ

Коды Боуза — Чоудхури — Хоквингема (БЧХ) являются подклассом циклических кодов. Их отличительное свойство — возможность построения кода БЧХ с минимальным расстоянием не меньше заданного. Это важно, потому что, вообще говоря, определение минимального расстояния кода есть очень сложная задача.

Математически полинома $g(x)$ на множители в поле Галуа.

Коды коррекции ошибок Рида — Соломона

Коды Рида — Соломона — недвоичные циклические коды, позволяющие исправлять ошибки в блоках данных. Элементами кодового вектора являются не биты, а группы

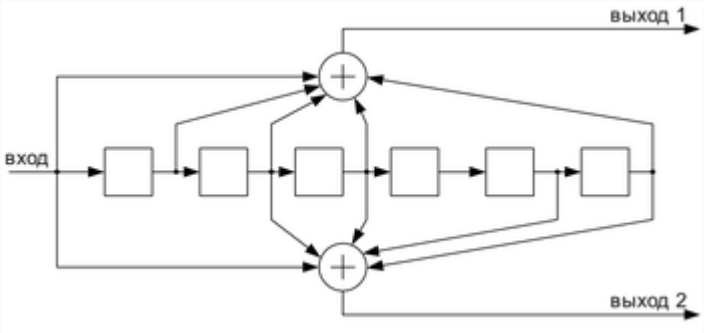
битов (блоки). Очень распространены коды Рида-Соломона, работающие с байтами (октетами).

Математически коды Рида — Соломона являются кодами БЧХ.

Преимущества и недостатки блоковых кодов

Хотя блоковые коды, как правило, хорошо справляются с редкими, но большими *пачками ошибок*, их эффективность при частых, но небольших ошибках (например, в канале с АБГШ), менее высока.

Свёрточные коды



Свёрточный кодер ($k = 7,\ R = 1/2$)

Свёрточные коды, в отличие от блоковых, не делят информацию на фрагменты и работают с ней как со сплошным потоком данных.

Свёрточные коды, как правило, порождаются дискретной линейной инвариантной во времени системой. Поэтому, в отличие от большинства блоковых кодов, свёрточное кодирование — очень простая операция, чего нельзя сказать о декодировании.

Кодирование свёрточным кодом производится с помощью регистра сдвига, отводы от которого суммируются по модулю два. Таких сумм может быть две (чаще всего) или больше.

Декодирование свёрточных кодов, как правило, производится по алгоритму Витерби, который пытается восстановить переданную последовательность согласно критерию максимального правдоподобия.

Преимущества и недостатки свёрточных кодов

Свёрточные коды эффективно работают в канале с белым шумом, но плохо справляются с пакетами ошибок. Более того, если декодер ошибается, на его выходе всегда возникает пакет ошибок.

Каскадное кодирование. Итеративное декодирование

Преимущества разных способов кодирования можно объединить, применив *каскадное кодирование*. При этом информация сначала кодируется одним кодом, а затем другим, в результате получается *код-произведение*.

Например, популярной является следующая конструкция: данные кодируются кодом Рида-Соломона, затем *перемежаются* (при этом символы, расположенные близко, помещаются далеко друг от друга) и кодируются свёрточным кодом. На приёмнике сначала декодируется свёрточный код, затем осуществляется обратное перемежение (при этом *пачки ошибок* на выходе свёрточного декодера попадают в разные кодовые слова кода Рида — Соломона), и затем осуществляется декодирование кода Рида — Соломона.

Некоторые коды-произведения специально сконструированы для *итеративного декодирования*, при котором декодирование осуществляется в несколько проходов, каждый из которых использует информацию от предыдущего. Это позволяет добиться большой эффективности, однако, декодирование требует больших ресурсов. К таким кодам относят *турбо-коды* и LDPC-коды (коды Галлагера).

Оценка эффективности кодов

Эффективность кодов определяется количеством ошибок, которые тот может исправить, количеством избыточной информации, добавление которой требуется, а также сложностью реализации кодирования и декодирования (как аппаратной, так и в виде программы для ЭВМ).

Граница Хемминга и совершенные коды

Пусть имеется двоичный блоковый (n,k) код с корректирующей способностью t. Тогда справедливо неравенство (называемое *границей Хемминга*):

$$\sum_{i=0}^{t} \binom{n}{i} \leqslant 2^{n-k}.$$

Коды, удовлетворяющие этой границе с равенством, называются *совершенными*. К совершенным кодам относятся, например, коды Хемминга. Часто применяемые на практике коды с большой корректирующей способностью (такие, как коды Рида — Соломона) не являются совершенными.

Энергетический выигрыш

При передаче информации по каналу связи вероятность ошибки зависит от отношения сигнал/шум на входе демодулятора, таким образом при постоянном уровне шума решающее значение имеет мощность передатчика. В системах спутниковой и мобильной, а также других типов связи остро стоит вопрос экономии энергии. Кроме того, в определённых системах связи (например, телефонной) неограниченно повышать мощность сигнала не дают технические ограничения.

Поскольку помехоустойчивое кодирование позволяет исправлять ошибки, при его применении мощность передатчика можно снизить, оставляя скорость передачи информации неизменной. Энергетический выигрыш определяется как разница отношений с/ш при наличии и отсутствии кодирования.

Применение кодов, исправляющих ошибки

Коды, исправляющие ошибки, применяются:

- в системах цифровой связи, в том числе: спутниковой, радиорелейной, сотовой, передаче данных по телефонным каналам.
- в системах хранения информации, в том числе магнитных и оптических.

Коды, обнаруживающие ошибки, применяются в сетевых протоколах различных уровней.

Автоматический запрос повторной передачи

Системы с автоматическим запросом повторной передачи (ARQ — Automatic Repeat reQuest) основаны на технологии обнаружения ошибок. Распространены следующие методы автоматического запроса:

Запрос ARQ с остановками (stop-and-wait ARQ)

Идея этого метода заключается в том, что передатчик ожидает от приемника подтверждения успешного приема предыдущего блока данных перед тем как начать передачу следующего. В случае, если блок данных был принят с ошибкой, приемник передает отрицательное подтверждение (negative acknowledgement, NAK), и передатчик повторяет передачу блока. Данный метод подходит для полудуплексного канала связи. Его недостатком является низкая скорость из-за высоких накладных расходов на ожидание.

Непрерывный запрос ARQ с возвратом (continuous ARQ with pullback)

Для этого метода необходим полнодуплексный канал. Передача данных от передатчика к приемнику производится одновременно. В случае ошибки передача возобновляется, начиная с ошибочного блока (то есть, передается ошибочный блок и все последующие).

Непрерывный запрос ARQ с выборочным повторением (continuous ARQ with selective repeat)

При этом подходе осуществляется передача только ошибочно принятых блоков данных.

См. также

- Цифровая связь
- Линейный код
- Циклический код
- Код Боуза — Чоудхури — Хоквингема
- Код Рида — Соломона
- LDPC
- Свёрточный код
- Турбо-код

Литература

- Мак-Вильямс Ф. Дж., Слоэн Н. Дж. А. Теория кодов, исправляющих ошибки. М.: Радио и связь, 1979.
- Блейхут Р. Теория и практика кодов, контролирующих ошибки. М.: Мир, 1986.

- Морелос-Сарагоса Р. Искусство помехоустойчивого кодирования. Методы, алгоритмы, применение. М.: Техносфера, 2005. — ISBN 5-94836-035-0

Ссылки

Имеется викиучебник по теме:
Обнаружение и исправление ошибок

- Помехоустойчивое кодирование (11 ноября 2001). — реферат по проблеме кодирования сообщений с исправлением ошибок. Проверено 25 декабря 2006.